

A FRAMEWORK AND METHODOLOGY FOR
LINKING INDIVIDUAL AND ORGANIZATIONAL LEARNING:
APPLICATIONS IN TQM AND PRODUCT DEVELOPMENT

by

Daniel H. Kim

B.S., Electrical Engineering, Massachusetts Institute of Technology, 1986

Submitted to the Sloan School of Management
in Partial Fulfillment of the Requirements for the Degree of
DOCTOR OF PHILOSOPHY

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

May 1993

© Daniel H. Kim 1993

All rights reserved

The author hereby grants to MIT permission to reproduce and to distribute
publicly copies of this thesis document in whole or in part.

Signature of Author _____

Sloan School of Management
May, 1993

Certified by _____

Charles H. Fine
Associate Professor of Management
Thesis Supervisor

Accepted by _____

James B. Orlin
Chairman, Ph.D. Committee
Department of Management

ARCHIVES

MASSACHUSETTS INSTITUTE
OF TECHNOLOGY

JUN 04 1993

A FRAMEWORK AND METHODOLOGY FOR LINKING
INDIVIDUAL AND ORGANIZATIONAL LEARNING:
APPLICATIONS IN TQM AND PRODUCT DEVELOPMENT

by
Daniel H. Kim

Submitted to the Department of Management in partial fulfillment
of the requirements for the degree of Doctor of Philosophy

ABSTRACT

The scope and magnitude of change that is occurring in business today is unprecedented in industrial history. The current paradigm shift from mass production to lean production—from the machine age view of the past to the systems view of the future—is one example of the profound forces that are reshaping corporations. The vast magnitude of this change has led to a deep interest in the concept of organizational learning.

This dissertation explores key issues in organizational learning by providing a framework for integrating individual and organizational learning. Specifically, it identifies mental models as the *transfer* mechanism through which individual learning becomes operationalized and advances organizational learning. Mental models are described as a critical mechanism for 1) enhancing individual learning by making the individual's learning explicit for that person, and 2) doing so in such a way that the learning can be more easily transferred and diffused throughout the organization as shared mental models.

By formulating a typology of methods for representing mental models, this dissertation shows the limits of current maps, and illustrates the effectiveness of causal loop diagrams and systems archetypes in the mapping of complex dynamic systems. System archetypes are described as dynamic scripts that can be used to map stories and schemas of significant dynamic complexity.

The methodology presented in this dissertation offers a detailed process through which individual learning is translated into organizational learning. First, individual learning is mapped into explicit mental model representations. From those representations, a more integrated diagram is developed that captures data and the experiences of other members of the organization for the purpose of making the mental models explicit. Systems archetypes are then used to abstract (or decompose) the large, integrated diagram into more understandable and memorable scripts. According to the model of organizational learning proposed in this dissertation, out of those shared mental models will come new organizational actions which will produce new environmental response, from which the learning cycle will continue.

The experience of applying the methodology is represented in two case settings: Total Quality Management implementation false starts and product development. In the case studies of TQM false-start implementations, the methodology is applied to developing a shared understanding of the organizational issues preventing successful TQM efforts. The product development management case study describes the full cycle of working with product development managers as a team to translate individual learning into organizational learning and action. Specifically, it identifies two leverage points for managing product development efforts: the importance of managing the dynamic implications of the Tragedy-of-the-Commons archetype, and the systemic reasons why a heavyweight program manager with wide authority over the entire program may be requisite for a successful product development project.

In so doing, this dissertation will attempt to make contributions to three areas: organizational learning theory, system dynamics and operations management.

Thesis Committee:

Charles H. Fine, Associate Professor of Management (chair)

Edgar H. Schein, Professor of Management

Peter M. Senge, Principal Researcher

*To my mother and father,
with love*

Acknowledgments

I dedicate this dissertation to my mother and father for their many years of hard work and innumerable sacrifices that laid the foundation on which I am able to build my future.

Pursuing a Ph.D. is an honor and a privilege for which I am grateful (although the process of *finishing* it is another matter). Like most worthwhile things in life, the completion of this dissertation would not have been possible without the help and involvement of numerous people. Though I won't be able to mention every name, I deeply appreciate all who have touched my life through these past several years, both within and outside of M.I.T.

First and foremost I thank Colleen, my wife, for her unwavering support throughout the whole process. It is hard not to succeed when I am blessed with such love. Even so, for added good measure, she provided me with her excellent editing skills in pulling together my final drafts.

To my dissertation committee, I am especially grateful for all the support and encouragement they have given me. In addition, I want to thank each member for his unique contribution to helping me through:

To Ed Schein, for his enthusiastic response to my ideas presented in Chapter 2 which led me to revise the direction of my dissertation to pursue those ideas in greater depth and for all that I have learned from him about process consultation and the clinical perspective that proved invaluable in my work at Cars, Inc.

To Charlie Fine, for sticking with me as we struggled together with my ever elusive dissertation topic and for taking an experimental step by supervising an interdisciplinary dissertation that will be of interest to operations management, system dynamics, and organizational studies.

To Peter Senge, for blazing the trail and galvanizing interest in systems thinking through his book, *The Fifth Discipline*, that led to the creation of the Organizational Learning Center which has supported my work in every way. As a mentor, colleague, and friend, I have learned much from him over the years.

My thanks to John Sterman whose brilliant teaching and genuine passion for the field of system dynamics is the reason that I came back to M.I.T. for a Ph.D. I also extend an intellectual debt of gratitude to Jay Forrester for pioneering this important and exciting field of system dynamics.

I thank all the team members at Cars, Inc. whose openness and commitment to the project was a key to its success. My thanks also to all the managers who participated in the TQM study at Test, Inc. and Wafers, Inc. for allowing us to study and learn from their mistakes.

My thanks to everyone at the Organizational Learning Center and the System Dynamics Group, especially to Angela Lipinski, for sustaining the spirit of friendship and collegiality that always make it a pleasant place to be. In particular, I thank Anjali Sastry, my fellow Ph.D. student and officemate, for accommodating my “camping out” in our office as I approached my crunch time, Gary Burchill for his partnership and support on the TQM projects, and Don Seville for his excellent assistance in the product development project.

To everyone at Pegasus Communications—Colleen, Michelle, Kellie, Deirdre, Anne, and all the interns—my thanks for their moral support and for allowing me to focus on my dissertation with complete confidence that all was well in their hands.

And finally a very special thank you to Eileen Mullen without whose professional and reliable help I could not have even hoped to finish within my timetable. Her writing/editing support and attention to details was of tremendous help. I especially appreciated the diligence with which she handled every request.

This dissertation marks the closing of yet another chapter in my life (seven chapters to be exact!) as I look forward to the beginning of a new one.

Daniel H. Kim
May, 1993

**A FRAMEWORK AND METHODOLOGY FOR LINKING
INDIVIDUAL AND ORGANIZATIONAL LEARNING:
APPLICATIONS IN TQM AND PRODUCT DEVELOPMENT**

Abstract	3
Acknowledgements	5
Table of Contents	7
Table of Figures	15
Chapter 1 Introduction	
1.1 Current Business Environment	20
1.1.1 Lean Production	20
1.1.2 The Essence of Lean Production	21
1.1.3 A Second Revolution	23
1.2 Emerging Interest in Organizational Learning	24
1.2.1 Paradigms	24
1.2.2 Problem-Solution Model of Management	25
1.2.3 Pre-determined Solutions	26
1.2.4 Problem Articulation (or Mess Formulation).....	27
1.2.5 Organizational Learning	29
1.3 Contributions of this dissertation	30
1.4 Summary of Chapters.....	31
Chapter 1 References	35
Chapter 2 A Framework for Integrating Individual Learning and Organizational Learning	
2.1 Introduction.....	37
2.1.1 Current Models of Organizational Learning	38
2.1.2 The Link between Individual and Organizational Learning	41
2.2 Individual Learning	43
2.2.1 A Working Definition of Learning	44
2.2.1.1 Levels of Learning: Operational and Conceptual.....	44
2.2.1.2 Experiential Learning Model	47
2.2.2 The OADI-IMM Model of Individual Learning	50
2.2.2.1 Individual Mental Models.....	52

2.2.2.2 Frameworks and Routines	54
2.3 Organizational Learning	57
2.3.1 Levels of Complexity	58
2.3.2 The Individual-Organization Learning Dilemma	59
2.3.3 Organizations as Behavioral Systems	61
2.3.4 Organizations as Interpretation Systems	64
2.3.5 The Missing Link: from Individual to Organizational Learning	66
2.4 An Integrated Model of Organizational Learning	67
2.4.1 The Role of Individuals in Organizational Learning	68
2.4.1.1 Groups and Organizational Learning	70
2.4.2 The Transfer Mechanism: Shared Mental Models	71
2.4.2.1 Weltanschauung and Organizational Routines	73
2.4.3 Incomplete Learning Cycles	75
2.4.3.1 Situational Learning	76
2.4.3.2 Fragmented Learning	76
2.4.3.3 Opportunistic Learning	77
2.5 Summary	77
Chapter 2 References	79
Chapter 3 Representing Mental Models	
3.0 Introduction	83
3.1 Literature On Mental Models	86
3.1.1 Physical Domain of Knowledge	88
3.1.1.1 Physical Representations	88
3.1.1.2 Analogies	90
3.1.1.3 Autonomous Objects	91
3.1.2 Conceptual Domain of Knowledge	92
3.1.2.1 Scripts	93
3.1.2.2 Story Model	95
3.1.2.3 Schemas	96
3.1.2.4 Influence Diagrams	98
3.1.2.5 Total Quality Management (TQM) Maps	100
3.1.2.6 Action Maps	104
3.1.2.7 System Dynamics	105
3.2 A Typology Of Mental Model Representations	110
3.2.1 Criteria for Representing Mental Models Based on the Literature	110

3.2.2 Domains of Usefulness: Dynamic vs. Detail Complexity	112
3.2.2.1 Characteristics of Dynamic Complexity	113
3.2.2.2 Organizational Complexity	114
3.2.2.3 Time Delays	115
3.2.2.4 Fuzzy variables	116
3.2.3 Mapping Dynamic Complexity	116
3.2.4 A Typology	118
3.3 Shared Mental Models	120
3.3.1 Accessibility	120
3.3.2 Systems Archetype Descriptions	123
3.3.2.1 Limits-to-Success Archetype	123
3.3.2.2 Diets and Weight Loss	124
3.3.2.3 Service Capacity Limit	125
3.3.2.4 Using the Archetype	127
3.3.3 Shifting-the-Burden	127
3.3.3.1 The Structure	128
3.3.3.2 Central vs. Local	129
3.3.3.3 Using the Archetype	131
3.3.4 Tragedy-of-the-Commons	132
3.3.4.1 Brazil's Inflation Game	134
3.3.4.2 Common "Commons"	135
3.3.4.3 Managing the Commons	136
3.3.5 Drifting-Goals	136
3.3.5.1 The Structure	137
3.3.5.2 Budget Deficits, Drifting Goals	138
3.3.5.3 Using the Archetype	139
3.3.6 Fixes-that-Fail	139
3.3.6.1 The Structure	140
3.3.6.2 Expediting Customer Orders	140
3.3.6.3 Using the Archetype	142
3.3.7 Summary Table of Systems Archetypes as Dynamic Scripts	142
3.4 Summary	146
Chapter 3 References	146

Chapter 4 A Methodology for Linking

Individual Mental Models to Shared Mental Models

4.0 Introduction.....	149
4.1 Research Process and Approach	150
4.1.1 Alternative Ways of Knowing.....	152
4.1.2 The Scientific Method	153
4.1.3 Ladder of Inference and the Reflexive Loop.....	156
4.1.4 Grounded Theory Building	159
4.2 Methodology for Mapping Individual Mental Models	162
4.2.1 Step 1: Select System and Gather Data.....	164
4.2.1.1 Field Theory and Force-field Analysis	165
4.2.1.2 Case Settings.....	169
4.2.2 Step 2: Coding Variables	169
4.2.2.1 Open Coding	169
4.2.2.2 Semantics as a way of "Cleaning" Verbal Data	170
4.2.2.3 Case Setting	172
4.2.3 Step 3: Describe Variable Relationships.....	173
4.2.4 Step 4: Map Individual Story Causal Loops	173
4.2.4.1 An Extended Example	174
4.2.5 Step 5: Identify Potential Archetypes	182
4.2.6 Step 6: Refine Fit of Archetype and Data.....	183
4.2.7 Summary of Steps 1-6	185
4.3 Methodology for Transfer of Individual Mental Models to Shared Mental Models	186
4.3.1 Step 7: Build Integrated Map.....	187
4.3.2 Step 8: Make Sense of Map through Decomposition	188
4.3.2.1 Model Decomposition.....	189
4.3.2.2 Causal Loop Diagram Map Decomposition	190
4.3.3 Step 9: Share Decomposition with Others	196
4.3.4 Step 10: Confirm and Test the Decomposition	196
4.3.5 Summary of Steps 7-10	197
4.4 Discussion	198
4.4.1 Validity	198
4.4.2 Role of System Dynamics Computer Models	203
4.4.3 A Final Note on Validity	206
Chapter 4 References	208

Chapter 5 Mapping Mental Models: TQM Implementation False-Starts

5.0 Introduction.....	211
5.1 Total Quality Management	213
5.1.1 A Brief History of Modern TQM	215
5.1.2 TQM Implementation in America	216
5.1.3 Emerging Problems with TQM Implementations	217
5.1.4 The Need for a Systemic Approach to TQM.....	218
5.2 Design of TQM Implementation False-Starts Study	221
5.2.1 A Systemic View of TQM Implementation	221
5.3 Case 1: Testing Equipment Manufacturer—Test, Inc.	224
5.3.1 Test, Inc.: Background	225
5.3.2 Test, Inc.: Chronology	226
5.3.3 Test, Inc.: Systems Maps.....	228
5.3.3.1 Directive without Commitment.	228
5.3.3.2 Customers Demand Quality.	229
5.3.3.3 Cost of Poor Quality.	230
5.3.3.4 Urgent vs. Important Work.....	231
5.3.3.5 Roll-out without Follow-up.	233
5.3.3.6 Training Capacity.	234
5.3.4 Test, Inc.: Integrated Map	236
5.3.5 Test, Inc.: Systems Archetypes	237
5.4 Case 2: Semiconductor Manufacturer—Wafers, Inc.	241
5.4.1 Wafers, Inc.: Background	242
5.4.2 Wafers, Inc: Chronology	243
5.4.3 Wafers, Inc.: Integrated Map	246
5.4.4 Wafers, Inc.: Systems Archetypes	248
5.5 Discussion	251
5.5.1 Common Failure Modes	251
5.5.1.1 Constancy of Purpose	252
5.5.1.2 Training Capacity	254
5.5.1.3 Old vs. New.....	255
5.5.1.4 Productivity Dilemma	256
5.5.2 Next Steps	257
5.5.3 Final Note	258
Chapter 5 References	259

**Chapter 6 Mental Models and Organizational Learning:
Product Development Management**

6.0 Introduction.....	261
6.1 Background on Product Development Management	264
6.1.1 Importance of Product Development	264
6.1.2 Product Development and Organizational Learning	267
6.1.3 Review of Studies on Product Development	268
6.1.3.1 Best Practices	268
6.1.3.2 Total Quality Management	270
6.1.3.3 Marketing	270
6.1.3.4 Organizational Behavior.....	271
6.1.3.5 Operations Management.....	273
6.2 Product Development Management Challenge.....	274
6.2.1 Focusing on Product Engineering and Process Engineering.....	275
6.2.2 Focusing on Overlap/Coordination	277
6.2.3 Product Development and the OADI-SMM Learning Cycle.....	278
6.3 Consumer Cars manufacturer—Cars, Inc.....	280
6.3.1 Company Background.....	280
6.3.2 Project Chronology	280
6.4 Mapping Shared Mental Models	282
6.4.1 Focusing on an Issue: Parts Behind Schedule	283
6.4.1.1 Supplier Timing	284
6.4.1.2 Build Timing.....	285
6.4.1.3 Development Process.....	286
6.4.1.4 Delay Program	286
6.4.1.5 Engineering Resources	287
6.4.1.6 Cancel Late Parts	288
6.4.1.7 Information Delay	289
6.4.1.8 Summary.....	291
6.4.2 Clarifying a Point through Mapping.	292
6.4.3 The Completed Systems Map.....	294
6.4.4 Leverage Points	297
6.4.5 Outcomes.....	298
6.5 From Shared Mental Models to Organizational Action	300
6.5.1 Tragedy of the “Power Supply” Commons Story	301
6.6 Research Findings	307

6.6.1 A “Commons Management” View of Product Development	308
6.6.1.1 Tragedy-of-the-Commons Revisited	310
6.6.1.2 Economic Externalities	311
6.6.1.3 Strategic vs. Operational Autonomy	313
6.6.1.4 Managing the Product Development “Commons”	313
6.6.2 Structural Impediments to Intensive Communication	315
6.6.2.1 Shifting-the-Burden and Functional Gridlock.....	316
6.6.2.2 Teams: Definition vs. incentive structure.	318
6.6.2.3. Implications	319
6.7 Building Shared Meaning	320
6.7.1 The Role of Computer Simulators and Learning Labs	320
6.7.1.1 Overview of Computer Model	321
6.7.1.2 Learning Labs.....	323
6.7.2 My Weltanschauung.....	324
6.7.3 A Final Note.....	325
Chapter 6 References	326
Chapter 7 Discussion and Summary	
7.0 Contributions	331
7.1 Contributions to Operations Management.....	332
7.1.1 Total Quality Implementation Setting	332
7.1.2 Product Development Management Setting.....	333
7.2 Contribution to System Dynamics:.....	335
7.2.1 Rigor, Relevance, and Usability	335
7.2.2 Levels of Use of Systems Archetypes.....	336
7.2.3 Model Decomposition and Identification of New Archetypes	338
7.3 Contributions to Organizational Learning	339
7.4 Future directions.....	341
7.5 Reflections	344
7.5.1 Managers’ New Roles: Researcher and Theory-Builder	344
7.5.2 Collaborative Research/Practitioner Model	345
Chapter 1 References	346
Full Reference List.....	349

Appendices.....357

Appendix A.....357

Appendix B359

Appendix C361

Appendix D.....363

Appendix E403

Table of Figures

No.	Title	Page
Chapter 1		
Figure 1.1	Fundamental Shifts in Competitive Emphasis	24
Figure 1.2	Problem-Solution Model	25
Figure 1.3	Problem Articulation Model	28
Chapter 2		
Figure 2.1	An Overview of Organizational Learning Models	40
Figure 2.2	Levels of Learning	45
Figure 2.3	The Lewinian Experiential Learning Model	49
Figure 2.4	The O-A-D-I Cycle of Individual Learning	50
Figure 2.5	A Simple Model of Individual Learning: OADI-IMM Cycle	52
Figure 2.6	March and Olsen's Model of Organizational Learning	62
Figure 2.7	Relationships among Organizational Scanning, Interpretation, and Learning	64
Figure 2.8	Types of Organizational Interpretation Systems	65
Figure 2.9	An Integrated Model of Organizational Learning: OADI-SMM Cycle	69
Figure 2.10	Incomplete Learning Cycles	75
Chapter 3		
Figure 3.1	Individual and Shared Mental Models	84
Figure 3.2	Explicit Mental Models Representations	86
Figure 3.3	Comparison of Naive and Physical Representations	89
Figure 3.4	Heat Exchanger Model	91
Figure 3.5	Sample Story Model for Juror Decision-making	96
Figure 3.6	Graphics vs. Language-based Representations	98
Figure 3.7	Partial Influence Diagram of Radon Risks	99
Figure 3.8	Graphics-Language System Classification of TQM Tools	101
Figure 3.9	Cause-Effect Diagram	103
Figure 3.10	A Relations Diagram	103
Figure 3.11	Action Map of a Counselor's Practice	105
Figure 3.12	Reinforcing Loop	107

Figure 3.13	Balancing Loop	109
Figure 3.14	Dynamic Complexity of “Messses”	115
Figure 3.15	Classification of Models	117
Figure 3.16	A Typology of Mental Model Representational Systems	119
Figure 3.17	Limits-to-Success Template	123
Figure 3.18	The Dieting Bind	125
Figure 3.19	Limits to Passenger Growth	126
Figure 3.20	Shifting-the-Burden Template	128
Figure 3.21	Central Support vs. Branch Capability	130
Figure 3.22	Table of Shifting-the-Burden Examples	131
Figure 3.23	Tragedy-of-the-Commons Template	132
Figure 3.24	Brazil’s Inflation	134
Figure 3.25	Drifting-Goals Archetype	137
Figure 3.26	Budget Deficit	138
Figure 3.27	Fixes-that-Fail Archetype	140
Figure 3.28	Expediting Customer Orders	141
Figure 3.29a-c	Systems Archetypes as Usable Theories	143-5

Chapter 4

Figure 4.1	Linking Individual Learning to Organizational Learning through Shared Mental Models	150
Figure 4.2	Model of the Scientific Method	154
Figure 4.3	The OADI Learning Cycle and the Scientific Method	155
Figure 4.4	The Ladder of Inference	157
Figure 4.5	From Individual Learning to Individual Mental Models	163
Figure 4.6	Force Field Diagram	168
Figure 4.7	Causes, Ccnstraints, and Consequences	168
Figure 4.8	Ladder of Abstraction	171
Figure 4.9	Partial List of Directed Arc Fragments	175
Figure 4.10a-h	Extended Example: Directive without Commitment	176-82
Figure 4.11	“Training Capacity” Story	184
Figure 4.12	“Training Capacity” Limits to Success	185
Figure 4.13	From Individual Mental Models to Shared Mental Models	187
Figure 4.14	Behavior Modes of Interest	191
Figure 4.15	“Training Capacity” Limits to Success—Revisited	192

Figure 4.16	Preserving Directed Arc Polarity	193
Figure 4.17	Collapsing Links Example	194
Figure 4.18	Simplified "Training Capacity" Story	195
Figure 4.19	Assessing the Gaps in Organizational Learning	201
Figure 4.20	Systems Archetypes and Computer Models	205
Figure 4.21	Operator vs. Observer View of Models and Validity	206
 Chapter 5		
Figure 5.1	Mapping Experience into Explicit Mental Models	212
Figure 5.2	TQM Implementation Model	222
Figure 5.3	Reinforcing TQM Implementation Loops	223
Figure 5.4	Multiple Possible Outcomes	224
Figure 5.5	Directive without Commitment	228
Figure 5.6	Customers Demand Quality	229
Figure 5.7	Cost of Poor Quality	231
Figure 5.8	Urgent vs. Important Work	232
Figure 5.9	Roll-out without Follow-up	233
Figure 5.10	Training Capacity	235
Figure 5.11	Main Reinforcing Implementation Loop	236
Figure 5.12	Test, Inc. Integrated Map	238
Figure 5.13	Wafers, Inc. Integrated Loop Diagram	247
Figure 5.14	Three Major Systemic Issues in TQM Implementation	252
 Chapter 6		
Figure 6.1	Double-Loop Organizational Learning	262
Figure 6.2	Average Project Schedule by Stages	276
Figure 6.3	Overlapping with and without Intensive Communication	277
Figure 6.4	Parts Behind Schedule	284
Figure 6.5	Compress Supplier Timing	285
Figure 6.6	Compress Build Timing	286
Figure 6.7	Compress Development Timing	286
Figure 6.8	Delay Program	287
Figure 6.9	Add Engineering Resources	288
Figure 6.10	Cancel Late Parts	289
Figure 6.11	Information Delay	290
Figure 6.12	"Parts Behind Schedule" Causal Loop Diagram	296

Figure 6.13	Tragedy of the Power Supply	303
Figure 6.14	Tragedy of the Suppliers	307
Figure 6.15	NVH-Chassis Reinforcing Cycle of Interlocking Problems	317
Figure 6.16	NVH-Chassis Interlocked in an "Us vs. Them" Structure	317
Figure 6.17	New Product Development Model Overview	322

Chapter 7

Figure 7.1	Classification Scheme for Determining Dominant Model Structure	339
Figure 7.2	Organization Theory Building Cycle	342

Chapter 1

Introduction

"One wonders sometimes if science will not grind to a stop in an assemblage of walled-in hermits, each mumbling to himself words in a private language that only he can understand."

—Kenneth E. Boulding

It is not an exaggeration to say that the pace and scope of change that is occurring today is unprecedented in industrial history. Many of the changes of the last decade have been of global importance. The dissolution of the Soviet Union and the end of the Cold War, the ensuing struggle to set up independent democracies, and the creation of a European Common Market in 1992 are just some of the major headlines of this past decade.

There have been many visible signs of major change in American business as well. The encroachment of the Japanese automakers into the U.S. domestic market, the near extinction of the U.S. motorcycle industry, and the decline and rebirth of our steel industry are just a few examples. More recent changes in corporate structure have resulted in an unprecedented number of CEO's ejected by their boards at such giants as IBM, General Motors, and American Express. These power-plays signal more than just disgruntled stockholders wanting to take a more active role; it is an admission that business-as-usual, conducted in the style of the old guard, is no longer acceptable. It is a symptom of a deeper

problem that lies at the heart of most organizations' problems—the structure of the organizations themselves.

1.1 CURRENT BUSINESS ENVIRONMENT

In *Made in America*, (Dertouzos, Lester, & Solow, 1989) an MIT commission assessed the current health of American business and concluded that American companies were uncompetitive along a number of fronts. In particular, Americans were lagging far behind the Japanese in manufacturing processes. The land of the “cheap imitators” had become the land of the “superior imitators.” Even though the US was still a leader in inventing new products, American businesses as a whole were unsuccessful at translating them into profitable businesses.

1.1.1 Lean Production

The profound revolution that has been taking place in manufacturing over the last several decades—from mass production to “lean” production—serves as a useful example to understand the magnitude of the shift that is occurring in business today. In 1991, Toyota, the world's third largest auto maker, had cash reserves large enough to buy both Ford and Chrysler at current stock prices with nearly \$5 billion to spare. That's sobering news for American car manufacturers, who face a slumping market, eroding profits, and increased foreign competition. Japanese carmakers now command over 25% of all US car sales—almost double the amount a decade ago. But automakers are not alone: during the same time period, Japanese firms have decimated the motorcycle industry both in the US and Britain (with the notable exception of Harley Davidson), dominated consumer electronics, and made major inroads in many other industries.

Many companies have responded to Japan's competitive onslaught by pursuing Total Quality Control (TQC), Just-in-Time (JIT), Quality Function

Deployment (QFD), and Statistical Process Control (SPC) in an effort to imitate Japan's success. But after many years of hard work, improvement rates are plateauing even as companies increase their efforts. Many have hit their organization's limit, given their current management system and organizational infrastructure.

The harsh reality now emerging is that many quality programs have distracted companies from a far more fundamental realization—that JIT, SPC and TQC are all pieces of a whole new way of doing business that is radically different from traditional Western methods. In *The Machine that Changed the World*, Womack, Jones, & Roos (1990) document with clarity and forcefulness the true nature of Japan's success, which they credit to a fundamental shift from mass production to what the authors call a "lean" production system.

The book, the culmination of a five-year, five million dollar study of the future of the automobile, involved researchers from around the globe and auto producers in 17 different countries. Not all of the best lean producers studied were Japanese—Ford, for example, has made much progress toward becoming lean. Nor were all of the worst manufacturers mass producers. But what the best manufacturers had in common was a systemic understanding of how to integrate separate parts of the organization into a smoothly functioning whole.

1.1.2 The Essence of Lean Production

The lean production method, pioneered by Eiji Toyoda, requires less human effort, manufacturing space, investment in tools, and product development time than mass production. According to Womack, et al., (1990), "Lean production welds the activities of everyone from top management to line workers to suppliers into a tightly integrated whole that can respond almost instantly to

marketing demands from consumers. It can also double production and quality, while keeping costs down¹.”

By almost every measure, the lean producer outperforms the classic mass producer. According to Womack, Jones and Roos, the mass assembler takes 31 hours to assemble a car with 130 defects per 100 cars and carries an average of 2 weeks' inventories of parts. The lean assembler takes 16 hours, produces 45 defects per 100 cars, and carries an average of 2 *hours* of inventories.

The lean assembly plant has two key organizational features, according to the book: “It transfers the maximum number of tasks and responsibilities to those workers actually adding value to the car on the line, and it has in place a system for detecting defects that quickly traces every problem, once discovered, to its ultimate cause” (p. 99). In the typical lean producer factory, “indirect” workers such as machine repairers or inventory runners are non-existent. Almost every person on the floor is working directly on the car. Narrow aisles between work areas facilitate face-to-face communication between workers, an important part of the collaborative learning that is essential to lean production.

Perhaps the most striking difference between the mass and lean producer occurs at the end of the assembly line: the massive areas full of cars needing further repairs before shipment is non-existent in the lean factory. Almost every car is driven directly from the line to boats or trucks for shipment.

The lean producer's advantage is not limited to the factory floor. In product design, the lean producer takes approximately 485 people on average to design a new car versus 900 for a mass producer. Average development time is 46.2 months for the Japanese, 60.4 months for Americans and 57.3 months for Europeans. As these numbers show, lean producers are able to offer a wider

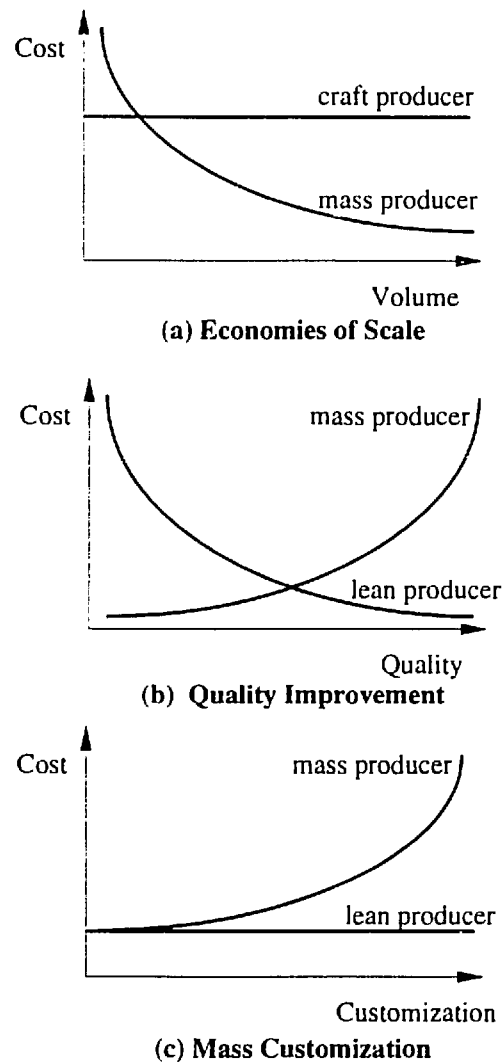
¹This is a direct quote from the back flap of the book jacket to *The Machine that Changed the World*.

variety of products and replace them more frequently than mass production competitors. Or they can use their cost savings to invest in new technologies and increase product attractiveness. “The choice, and the advantage,” note the authors, “will always lie with the lean producer” (Womack et al., p. 127).

1.1.3 A Second Revolution

The shift from mass production to lean production constitutes the second manufacturing revolution. Most craft producers did not realize in the early 1900s that their very survival was threatened by mass production. The competitive landscape in the automotive industry changed dramatically when the mass producers overwhelmed the craft producers with economies of scale. Their performance disproved the belief that unit costs stayed constant independent of volume (see Figure 1.1). As the lean producers emerged, the competitive emphasis shifted to quality improvement, where the old dogma of high quality, high cost had to be abandoned. The maturation of the lean producer is challenging yet another axiom—that increasing customization leads to increasing costs—as we enter the age of mass customization.

As we near the turn of another century, mass producers face a similar challenge. “Manufacturing companies in all industries will be affected by the spread of lean production,” claim Womack, et al. “They will be forced to [alter the way they operate] or face [extinction] by lean production competitors....Its adoption, as it inevitably spreads beyond the auto industry, will change almost every industry...and the fate of companies and nations as they respond to its impact” (Womack et al., p. 12).



Fundamental Shifts in Competitive Emphasis

Figure 1.1

1.2 EMERGING INTEREST IN ORGANIZATIONAL LEARNING

1.2.1 Paradigms

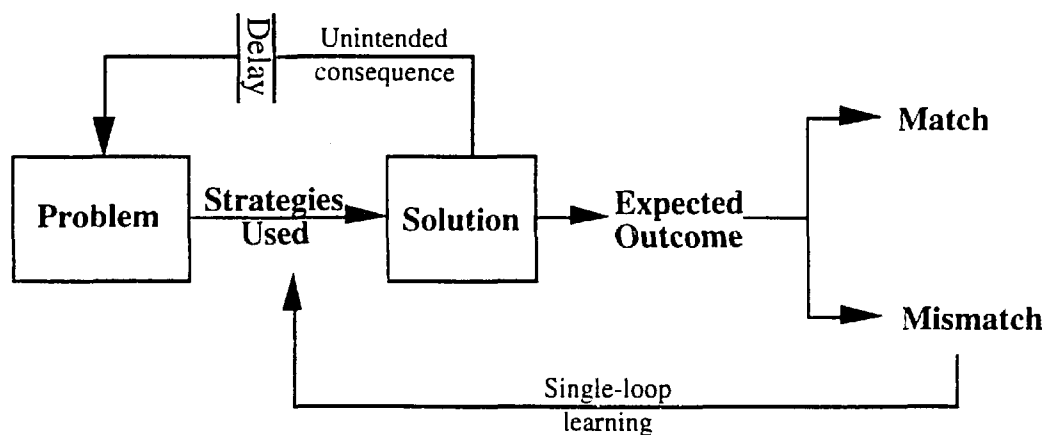
The profound changes that are occurring in business today, summarized in the shift from mass production to lean production, can be categorized as a paradigm shift. To respond to that shift, what is needed may not be a change of action, but a change in perception. How we think, act, and value are all associated with our particular view of reality. In order to create a new "reality," we must discover how our current world view affects the way we perceive and respond to

problems. The leverage lies in going to a more fundamental level—to look beyond current problems and re-examine the paradigm that gave rise to them. As Perrow (1986, p. 137) writes, there seems to be

a general trend in the social sciences and humanities that is likely to be significant over the next decade or two. This trend can be labeled “deconstructionism.” For a couple of centuries we have been “constructing” a world that we view as organized on rational principles, where what happened was intended to happen, where interactions are discrete and quite atomistic, and where progress is continuous....All this is now being questioned, and we are thus beginning to “deconstruct” this construction.

1.2.2 Problem-Solution Model of Management

The predominant model of management can be described as a “problem-solution” model (Argyris & Schön, 1978; Cowan, 1991; Thornton, 1992) individuals encounter problems (a mismatch between what is desired and what is obtained) and take corrective action to produce what they want (see Figure 1.2). This is what Argyris & Schön refer to as single-loop learning. In this single-



Problem-Solution Model

Figure 1.2

loop mode, people attack each instance of the problem individually, “fix” it, move on if the outcome matches expectations, or go back and modify their strategies if there is a mismatch. For example, a marketing manager’s response to sluggish sales is to launch a new ad campaign to stir up sales. If, after a couple

of weeks, sales increase as expected, she moves on to the next product in trouble. If sales fall short of expectations, she may try a variation of the old campaign or try a whole new one.

From a systems thinking perspective, however, solutions often feed back to create *other* problems, or even a repeat of the same problem (Senge, 1990b; Sterman, 1989). By the time this happens, it can appear to be a brand new problem because people have either forgotten about the previous round of solutions, or the same person is no longer in that position (the average tenure in a position is 18 months or less in some companies). Studies have shown that managers systematically misperceive feedback and cannot transfer lessons learned from one setting to a similar setting (Bakken, 1993; Diehl, 1992; Sterman, 1989).

Bakken (1993), for example, conducted a series of experiments using two isomorphic cases with different “cover” stories. The experiment involved the use of a system dynamics computer simulation model that had been converted into an interactive decision making game (see also Peterson & Tok How, 1993). In one setting the simulator was presented as a real estate development game where the individual was to make decisions to maximize profits by investing in new developments or buying existing properties. Although players learned to improve their scores in repeated trials in that setting, they did not transfer the learnings to the next setting which was structurally identical but with a different cover story of it being the supertanker business.

1.2.3 Pre-determined Solutions

At its worst, the problem-solution paradigm leads people to see problems in terms of pre-determined solutions. This phenomenon is similar to the notion of competency traps where companies continue to use an inferior technology

because that was the first one they had used, and they had become skilled at using it (Levitt & March, 1988; Reinhart, 1985). Statements such as, “The problem is we need a better information system,” or “The problem is we need the latest flexible manufacturing system,” are examples of solution statements at work. The danger of this habit is that once problems are framed in terms of solution statements, there is a tendency to exclude exploring other possibilities—including the possibility that the original statement of the problem may be wrong. That is, people never engage in double-loop learning.

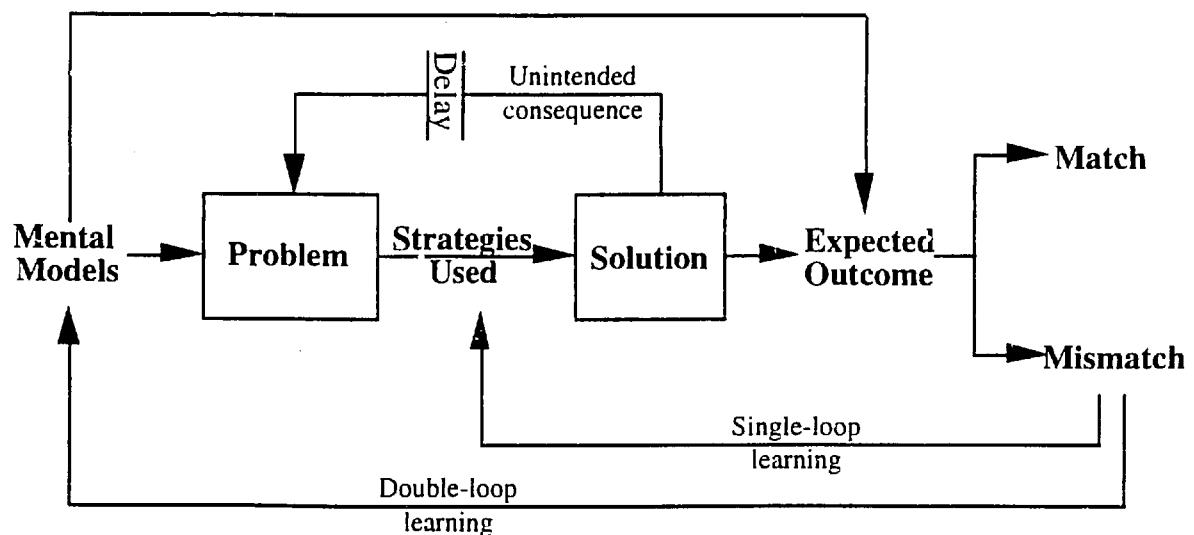
Even when someone’s or some department’s favorite solution is not used, the problem statement itself is often not challenged. A problem can be defined as *a formal statement of a set of assumptions* about the world. Those assumptions, however, are often not made explicit. By conversing and making decisions at the level of tacit assumptions, people can get very good at defending their point of view at the expense of learning. This can lead to what Chris Argyris calls “skilled incompetence.” Rather than looking at the real data and real issues—which may prompt a re-articulation of the problem—people become very skilled at dancing around the issues.

1.2.4 Problem Articulation (or Mess Formulation)

Problems that are “solved” often do not remain solved for long because nothing has been done to fundamentally alter the deeper forces that gave rise to the problems in the first place (the unintended consequence loop at top of Figure 1.3). Or, in Argyris and Schön’s terms, the underlying values and assumptions which govern the stated goals have not been surfaced and changed, a process they refer to as double-loop learning.

To re-examine the way we think about problems and solutions, we need to understand more fundamentally what a problem is. In reality, there are no

problems “out there” in the world—nature just *is*. Whether we see an event or situation as a problem depends on our view of the world. For example, if oil prices double, is that a problem? Our response would be a resounding “Yes!,” since our economy is heavily dependent on petroleum products. If we lived in an OPEC nation, however, we would not see it as a problem at all. If we lived in an undeveloped country with no dependence on oil, we probably would not even be interested in the issue.



Problem Articulation Model

Figure 1.3

Problems do not exist independently of the person who sees them. Out of the pool of life we “construct” problems in our minds (or in our organizations) by the way we view reality. Kofman (1992) suggests that deconstructing a problem and finding a way to re-articulate it can provide much more leverage than trying to just double our efforts to solve the problem as it is currently stated. One of the clear challenges is to explore more explicitly how we articulate problems. *Why* do we consider something a problem? The “why” is what leads us to surface the deeper set of assumptions that may give insight into reformulating an entirely different problem.

Kuhn (1962, p. 62) relates a vivid example of how paradigms can blind us from seeing what actually “is” because our perception becomes so heavily influenced by what we “expect” to see:

In a psychological experiment that deserves to be far better known outside the trade, Bruner and Postman asked experimental subjects to identify on short and controlled exposure a series of playing cards. Many of the cards were normal, but some were made anomalous, e.g., a red six of spades and a black four of hearts...

Even on the shortest exposures many subjects identified most of the cards, and after a small increase all the subjects identified them all. For the normal cards these identifications were usually correct, but the anomalous cards were almost always identified, without apparent hesitation or puzzlement, as normal. The black four of hearts might, for example, be identified as the four of spades or hearts. Without any awareness of trouble, it was immediately fitted to one of the conceptual categories prepared by prior experience...Even at forty times the average exposure required to recognize normal cards for what they were, more than 10 per cent of the anomalous cards were not correctly identified.

“Knowing” that hearts are red and spades are black can preclude us from seeing what is really there. In what ways do our mental models blind us from seeing the world as it is, and from noticing signs of real distress in the system?

1.2.5 Organizational Learning

The vast magnitude of change occurring in organizations today has led to a deep interest in the concept of organizational learning. A study by Shell International Petroleum Company claimed the average age of a corporation to be less than the average lifetime of a human being, and stressed that the key to an organization’s longevity is organizational learning (de Geus, 1988, p. 70).

It has been claimed that organizational learning is the root from which all competitive advantage stems (Stata, 1989). The level of advantage depends on the speed and quality of learning, whether behavior change is accompanied by cognitive change, and whether continual education is emphasized over sporadic training. By emphasizing the importance of trying to *understand* a problem, not simply *solve* it, the framework and methodology in this dissertation attempts to help transform problem-solving organizations into *learning organizations*. A

learning organization is one that consciously manages its learning process to be consistent with its strategies and objectives through an inquiry-driven orientation of all its members. That is, learning organizations *actively and explicitly* encourage the learning process to ensure that areas of strategic importance are not neglected and emphasizes the importance of trying to *understand* a problem, not simply *solve* it.

Building a learning organization means continually developing the capacity to create one's vision of the future (Senge, 1990b). Designing and implementing effective policies to create desired results requires an understanding of an organization and its environment as a unified system so that one can focus on a small set of high-leverage points to produce changes that self-reinforce and endure. To acquire such an understanding requires an ongoing management education process to develop a new style of thinking with the right blend of analysis and synthesis.

Although the topic of organizational learning has gained a lot of attention in recent years, there is little agreement on what we mean by organizational learning and even less on what it means to create a learning organization. This lack of consensus is representative of the current state of knowledge about the whole concept of organizational learning. Without an explicit understanding of the mechanism through which individual learning is transferred to the organization, there is little opportunity to actively manage the learning process of an organization.

1.3 CONTRIBUTIONS OF THIS DISSERTATION

This dissertation makes contributions in three areas:

- 1) At the *theoretical* level, it provides a framework and a methodology for integrating organizational learning and individual learning. The key

- transfer mechanism in this framework is “mental models”—constructs and routines that shape how we think and act.
- 2) The contribution for *system dynamics* is two-fold. It is an attempt to increase accessibility and broaden the system dynamics audience through further development of the front-end tools of systems archetypes (as opposed to the higher-end, computer modeling tools). Secondly, I link the concept of system archetypes and causal loop diagrams to the organizational literature relating to mental models, juxtaposed with scripts, stories, and schemas, clarifying the particular strengths that system dynamics mapping has relative to these other types when mapping dynamic complexity.
 - 3) This dissertation contributes to *operations management* by providing insights into TQM and product development management in Chapters 5 and 6, respectively. Specifically, the studies of TQM implementation false starts begin to provide a base of understanding of the systemic reasons why TQM implementations fail. In product development, we found theoretical support for why heavyweight program managers are needed to manage large product development programs—because they are fundamentally “commons” management issues.

1.4 SUMMARY OF CHAPTERS

Much has been written about the topic of organizational learning, but the discussions usually address either individual learning *or* organizational learning, without explicitly addressing how individual learning actually translates into organizational learning. The tendency is to jump from individual learning to organizational learning, with the implication that organizational learning retains all of the attributes of individual learning, but on a larger scale.

This dissertation provides a framework and methodology for *linking* individual learning to organizational learning. We develop a simple model of individual learning which is then integrated into a model of organizational learning in which two types of learning are differentiated—*conceptual* and *operational*. To do this, we address the role of individual learning and memory, differentiate between different levels of learning, take into account different organizational types, and specify the explicit mapping of *mental models* as the transfer mechanism between individual and organizational learning.

The purpose of **Chapter 2** is to build a theory about the process through which individual learning advances organizational learning. Chapter 2 lays out a framework for understanding the link between individual and organizational learning, first by building a model of individual learning based on the experiential model of Piaget, Dewey, and Lewin. This model, which is grounded in real experience, is a continuous cycle of *observation, assessment, design, and implementation*. For the purposes of organizational learning, however, the model is inadequate because it does not specifically address the process of making the learning explicit for the individual, and therefore not easily transferable to the organization.

To address the need for a transfer mechanism from individual learning to organizational learning, we identify mental models as a critical mechanism for 1) enhancing individual learning by making the individual's learning explicit for that person, and 2) doing so in a way in which the learning can be more easily transferred and diffused throughout the organization as *shared mental models*. According to our proposed model of organizational learning, out of those shared mental models will come new organizational actions which will produce new environmental responses, from which the learning cycle continues. Thus, Chapter 2 lays out framework of individual learning, and proposes that

organizational learning is dependent on individual learning and built through the mechanism of mental models.

In **Chapter 3**, we address more explicitly the challenge of operationalizing individual mental models in a way that will enable the individual and the organization to have access to them. Therefore, we begin by exploring various representational systems of mental models. We survey the literature in the physical domain and the conceptual domain. We make a distinction between *detail complexity* and *dynamic complexity* and assert that most of the representational systems are good at mapping detail complexity. That is, they are designed to address either very static relationships or relatively simple systems but are not well-suited to represent complex dynamic systems. However, most of the significant problems in organizations lie in the realm of dynamic complexity.

We then explain the features of dynamic complexity and propose a methodology that is well-suited to address dynamic complexity; namely, the field of *system dynamics* and its set of tools. In particular, system dynamics has a set of representational systems that is very powerful for making individual mental models explicit.

However, when we begin to address the issue of building shared mental models, *accessibility* becomes a key issue. We cannot expect an individual to carry an explicit mathematical model in their heads. *Systems archetypes*, however, offer a simple representational schema that is particularly well-suited for communicating complex dynamic phenomena in a compact and succinct form. Systems archetypes can be seen as *dynamic scripts* that are well-suited for capturing dynamic complexity. Five systems archetypes are presented and described, along with illustrations of their suitability for mapping dynamically complex issues.

After having proposed the model and framework, **Chapter 4** describes the proposed *methodology* for how we map individual mental models, and how individual mental models become shared mental models. This dissertation proposes a particular methodology that addresses dynamic complexity via system dynamics representational systems. Chapter 4 presents a detailed process through which individual learning is translated into an explicit mental model representation through *grounded theory* building. From that, a more integrated diagram is developed that captures data and experience of other members of the organization for the purpose of making the mental models explicit. Then we use the systems archetypes to abstract or *decompose* the large and relatively complicated picture back into more understandable and memorable (rememberable) chunks.

The methodology contains at least three main phases: 1) to make individual learning explicit in the form of mental models; 2) to put it into a form in which it can be readily shared; and 3) to test how well it is shared. We present ways in which the level of sharedness can be tested, as well as the validity of the mental model representations themselves.

The next two chapters present the experience of applying the methodology in two different case settings. In **Chapter 5**, we present the results of TQM (Total Quality Management) implementation false-starts in two different companies. We begin with a brief history of total quality management, discuss TQM implementation in America, the emerging problems with TQM implementations that have been encountered, and the need for a systemic approach to TQM. We then demonstrate the process of mapping from individual learning to individual mental models, integrating them into a diagram, and decomposing the large diagram into systems archetypes. This study, however, stops short of establishing the sharedness of the mental models or testing their validity because

that was not its intent. Instead, the purpose was to use the mapping methods to help make systemic sense of the companies' experiences.

In **Chapter 6**, we take the methodology further by working actively in partnership with a real line management team around product development. In this setting we describe the full cycle of working together as a team to translate individual learning to individual mental models to an integrated map. The systems archetypes were used to share insights with others in the program which began the process of building a common understanding of some critical management issues. In addition, we discuss briefly another way in which the mental models were more widely shared through the development and implementation of managerial practice fields or *learning labs*. These learning labs integrate the mapping methodology and the research approach described in Chapter 4 with system dynamics computer tools, such as management flight simulators, to accelerate the process of building shared mental models.

Finally, **Chapter 7** contains a brief summary and reflections on the role of systems archetypes as a representational system for mapping individual mental models and for building *shared* mental models. Implications for further work on the framework and methodology are discussed as part of a larger research effort at the MIT Organizational Learning Center.

Chapter 1 References

- Argyris, Chris , & Schön, Donald (1978). *Organizational Learning: A Theory of Action Perspective*. Reading, MA: Addison-Wesley.
- Bakken, Bent (1989). Learning in Dynamic Simulation Games: Using Performance as a Measure. In P. Milling & E. Zahn (Eds.), *Computer -Based Management of Complex Systems: Proceedings of the 1989 International Conference of the System Dynamics Society*. Berlin: Springer-Verlag.
- Bakken, Bent (1993) *Learning and Transfer of Understanding in Dynamic Decision Environments*. Unpublished Dissertation, MIT Sloan School of Management.
- Cowan, D.A. (1991). Effect of Decision Making Styles and Contextual Experience on Executive Descriptions of Organizational Problem Formulation. *Journal of Management Studies*, 28(September), 463-83.
- de Geus, Arie (1988). Planning as Learning. *Harvard Business Review*, March-April, 70-74.
- Dertouzos, Michael L., Lester, Richard K., & Solow, Robert M. (1989). *Made in America: Regaining the Productive Edge*. Cambridge, MA: MIT Press.

- Diehl, Ernst (1989). *A Study of Human Control in Stock-Adjustment Tasks*. Unpublished manuscript, MIT Sloan School of Management.
- Diehl, Ernst (1992) *Effects of Feedback Structure on Dynamic Decision Making*. Unpublished dissertation, MIT Sloan School of Management.
- Kofman, Fred (1992). Lecture Slides. In Cambridge, MA: MIT Sloan School of Management.
- Kuhn, Thomas (1962). *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press.
- Levitt, Barbara, & March, James G. (1988). Organizational Learning. *Annual Review of Sociology*, 14, 319-340.
- Perrow, Charles (1986). *Complex Organizations: A Critical Essay*. New York: Random House.
- Peterson, John T., & Tok How, Lim (1993) *A System Dynamics Analysis of Total Quality Management Implementation False Starts*. Unpublished Master's Thesis, MIT Sloan School of Management.
- Reinhart, C. (1985). The Competency Trap. *Training*, 22(May), 160.
- Senge, Peter M. (1990b). *The Fifth Discipline*. New York: Doubleday.
- Stata, Ray (1989). Organizational Learning—The Key to Management Innovation. *Sloan Management Review*, 12(1), 63-74.
- Sterman, John D. (1989). Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment. *Management Science*, 35,(3), 321-339.
- Thornton, P.B. (1992). Teamwork: Focus, frame, facilitate. *Managerial Review*, 81(November), 46-7.
- Womack, James P., Jones, Daniel T., & Roos, Daniel (1990). *The Machine that Changed the World*. New York: Rawson Associates.

Chapter 2[†]

A Framework for Integrating Individual Learning and Organizational Learning

"Organizational learning may be the only sustainable competitive advantage."

—Ray Stata

2.1 INTRODUCTION

All organizations learn¹, whether they consciously choose to or not—it is a fundamental requirement for their sustained existence. Some firms are deliberate in their efforts to advance organizational learning, developing capabilities that are consistent with their objectives, while others make no focused effort and, therefore, acquire habits that are counter-productive towards achieving stated goals. Nonetheless, all organizations do learn.

Organizations are comprised of individuals and must ultimately learn via their individual members. Hence, theories of individual learning are crucial for understanding organizational learning. Although psychologists have studied the area of individual learning for decades, they are still far from fully understanding the workings of the human mind. Our

[†]An adaptation of this chapter is forthcoming in *Sloan Management Review*, Fall 1993, "Creating Learning Organizations: Understanding the Link between Individual and Organizational Learning."

¹Stating that an organization "learns" begs the question "What is an organization that it may learn?" This issue will be addressed later in the chapter, but for the time being, we can think of it as an organizational metaphor based on its familiar usage with individuals.

understanding of organizational learning, likewise, is still in an embryonic stage (Huber, 1991; Sims, Gioia, & Associates, 1986). In addition, theories of organizational learning often do not explicitly address the role of the individual (Duncan, 1974; Daft & Weick, 1984; Levitt & March, 1988). Within the context of organizations, one cannot talk about individual learning in a vacuum. Nor can we talk about organizational learning without being clear about the role of the individual, lest we become guilty of anthropomorphizing organizations.

The purpose of this chapter is to build a theory about the *process* through which individual learning advances organizational learning. To do this, we must address the role of *individual learning and memory*, differentiate between different *levels of learning*, take into account different *organizational types*, and specify the *transfer mechanism* between individual and organizational learning. Each of these elements are developed in greater detail in later sections of this chapter. In a broader context, the purpose of this chapter is to contribute towards building a theory of what a *learning organization* is by developing a framework that focuses on the crucial link between individual learning and organizational learning. Once we have a clear understanding of this transfer process, we can actively manage the learning process to be consistent with an organization's goals, vision, and values.

2.1.1 Current Models of Organizational Learning

Learning occurs at different levels, and each level involves very different processes and capabilities (see Figure 2.1). For example, Fiol & Lyles (1985) distinguish between lower-level learning (routine, which occurs through repetition in a well-understood context and at all levels in an organization)

and higher-level learning (non-routine, which occurs through heuristics and insights in an ambiguous context and mostly at upper levels of an organization). Others have made similar distinctions such as single-loop vs. double-loop learning (Argyris & Schön, 1978), behavioral level and strategy level learning (Duncan, 1974, Dutton & Jackson, 1987), habit-forming vs. discovery learning (Hedberg, Nystrom, & Starbuck, 1976), and reactive vs. proactive learning (Miles & Randolph, 1980). Each of these theories implies a preference to one type of learning either for specific groups of people (higher-level *vs.* lower-level) or for the entire organization (double-loop *vs.* single-loop). The view that different types of learning can be pigeon-holed into specific parts of an organization or that one type of learning is wholly preferable over another is too simplistic. The organizational learning process is much too diffuse and dynamic to be categorized and located so precisely.

These views of individual and organizational learning, however, have several gaps which leave them incomplete for our discussion of organizational learning. Daft & Weick's (1984) model of organizations as interpretation systems does not explicitly deal with individual actors at all. March and Olsen's (1975) model largely ignores the interaction between individual learning and learning at the organizational level. Roth (1992, p. 7) cites four shortcomings in March and Olsen's model: "inattention to stimuli interpretation and sharing of meaning among individuals; lack of consideration for structural elements impeding learning; focus on the organization as a learning environment of individuals; and emphasis on environmental response to organizational actions." In March & Olsen's model, individual learning is primarily driven by environmental responses, and organizational learning occurs when the whole cycle is completed. Their

Author	Label	Meaning
Argyris & Schön (1978)	<i>Learning</i> Single-loop Double-loop	Lower-level cognition Higher-level cognition
Cangelosi & Dill (1965)	<i>Learning</i> Interaction between individual & group adaptation	Behavioral development Cognitive development
Chakravarthy (1982)	<i>Adaptation</i>	Cognitive development
Cyert & March (1963)	<i>Learning</i> Adaptation of goals, attention rules and search rules	Behavioral development
Daft & Weick (1984)	<i>Learning</i> Action after interpretation	Behavioral development
Durcan (1974)	<i>Learning</i> Behavioral level Strategy level	Behavioral development Cognitive development
Duncan & Weiss (1978)	<i>Learning</i> Action-outcome relationships	Cognitive development
Hedberg (1981)	<i>Learning</i> Habit-forming Discovery	Behavioral development Cognitive development
Jelinek (1979)	<i>Learning</i> OST-belief sharing	Cognitive development
March & Olsen (1975)	<i>Learning</i> Rational adaptation Interpretation	Cognitive development
Meyer (1982)	<i>Adaptation</i> Deviation-reducing Deviation-amplifying	Lower-level cognition Higher-level cognition
Miles (1982)	<i>Learning</i> Diversification outcomes Planning formalization	Behavioral development Cognitive development
Miles & Randolph (1980)	<i>Learning</i> Reactive learning Proactive learning	Behavioral development Cognitive development
Miller & Friesen (1982)	<i>Adaptation</i> Actions	Behavioral development
Shrivastava & Mitroff (1982)	<i>Learning (Systems)</i> Evolutionary Designed	Behavioral development Cognitive development

An Overview of Organizational Learning Models

Source: Fiol & Lyles (1985)

Figure 2.1

model implies that all organizational learning must be driven in some measure by what is happening in the environment, and does not explain what learning occurs *within* a firm, independent of the outside environment.

Argyris and Schön (1978) focus their study of learning at the individual level. Their theory of learning is defined in terms of an individual's ability to change strategies to correct deviations from desired results (single-loop learning), and ability to change the cognitive maps that drive strategy formulation (double-loop learning). They then extend their model of individual learning to organizations, speak of an organization as if it were simply a large individual.

Others let organizational learning stand for the actions of a group of individuals, such as a top management group (Miles, Snow, Meyer, & Coleman, 1978). As with March & Olsen and Argyris & Schön, there is no explicit transfer process identified through which individual learning is being retained by the organization. Hence, if individuals should leave, the organization is likely to suffer a tremendous loss in its learning capacity.

2.1.2 The Link between Individual and Organizational Learning

I would argue that at the heart of organizational learning is the transfer process through which individual learning becomes embedded in an organization's memory and structure. This transfer process, which is essential to organizational learning, has received very little attention and is not well understood (Hedberg, 1981; Huber, 1991) although a promising interaction between organization theory and psychology has begun (Cohen, 1991). Schein (1993) explores links between psychology and individual learning in an organizational setting, adding emotional conditioning and

learned anxiety to his working definition of learning—a concept that affects both operational and conceptual learning.

In this chapter, in order to better understand organizational learning, we will look at several theories of individual learning, present a working definition of learning, as well as models of experiential learning that differentiate between what I call operational and conceptual learning.² Using a simple model of individual learning, we will discuss two types of mental models, frameworks and routines, and address the individual-organizational learning dilemma by integrating these into a model for organizational learning. This combined framework highlights the crucial link between organizational and individual learning, i.e., the transfer mechanism of mapping *mental models* that allows an organization to absorb individual learning.

Three potential incomplete learning cycles that have not been previously identified are introduced and explained. The OADI-SMM model of organizational learning in Figure 2.10 represents an integration of learning theories of individuals, current models of organizational learning, and the role of mental models on learning, both in individuals and in organizations. Although this model builds upon the current theories of organizational learning, specifically March and Olsen (1975) and Argyris and Schön (1978), it focuses more directly on the *transfer process* between individual learning and organizational learning via mental models.

2P. M. Senge makes a similar distinction between instrumental learning, adjustments in behavior to cope with changing circumstances, versus generative learning, changes in dominant ways of thinking in "Organizational Learning: New Challenges for System Dynamics," D-memo 4023, System Dynamics Group, MIT Sloan School of Management, Cambridge, MA. (Senge, 1989)

2.2 INDIVIDUAL LEARNING

The importance of individual learning for organizational learning is at once obvious and subtle—obvious because all organizations are composed of individuals; subtle because organizations can learn independent of any *specific* individual. The topic of learning at the individual level has been heavily researched by psychologists, linguists, educators, and others. There have been discoveries about cognitive limitations (Simon, 1957) as well as the seemingly infinite capacity of the human mind to learn new things (Restak, 1988). Piaget's focus on cognitive-development processes of children and Lewin's work on action research and laboratory training have provided much insight into how we learn as individuals and in group settings (Kolb, 1984). Some of these theories are based on stimulus-response behaviorism; some focus on cognitive capabilities, and others on psychodynamic theory. Numerous other theories about the ways humans learn have been proposed, debated and tested, such as Pavlov's classical conditioning, Skinner's operant conditioning, Tolman's sign learning, Gestalt theory, and Freud's psychodynamics (Hilgard & Bower, 1966).

Research in the area of human cognition has also revealed some interesting findings about our mental capabilities. Miller (1956) for example, identified the limits of short term memory to handling roughly seven "chunks" of information, a limit which he called the span of absolute judgment. Simon (1957) advanced the notion of bounded rationality and the act of satisficing in the absence of perfect information. Tversky and Kahneman (1974) revealed the consistent biases and irrelevant cues that influence people's decision-making as well as the dramatic differences produced by the framing of decisions (Tversky & Kahneman, 1987)

Despite all the research done to date, we still know relatively little about the human mind and the learning process. It seems that the more knowledge we gain, the more we realize how little we really do know. Models on human thought are under constant revision. Before proceeding further, however, we need to have a common definition of the word "learning" upon which to build.

2.2.1 A Working Definition of Learning

Johnson-Laird (1983) remarked that a rationally designed language would try to avoid the use of ambiguous words, and if used, would use them infrequently. Yet the words we use most often are precisely the ones with many different meanings and are, hence, ambiguous. Jaques (1989) listed 24 words, such as, manager, plan, work, results, etc., and stated that not one word is unequivocally defined in the whole field of organization development. Each word has "so many meanings that they have value only as vague slogans."

The problem is that our language is not rationally designed, but evolves and emerges through daily use. Certain terms become over-used because new terms are not spontaneously created to represent a subtle difference in meaning that is required. Instead, we use and re-use the same words with many qualifiers to convey the specific meaning desired. In the process, the qualifiers (themselves inadequate) get dropped and the original word itself is left to convey the many different meanings. Such is the case with the word "learning," a term whose meaning varies widely by context.

2.2.1.1 Levels of Learning: Operational and Conceptual

The dictionary definition of learning simply states "the acquiring of knowledge or skill." Learning by this definition encompasses two meanings:

the acquisition of skill or *know-how* which implies the physical ability to produce some action and the acquisition of *know-why* which implies an ability to articulate a conceptual understanding of an experience. Argyris and Schön's (1978) theory of action stresses the importance of action as a measure of what is actually learned. In their framework, learning has taken place only when new knowledge has been translated into different behavior that is replicable.

	Lower-level	Higher-level
<i>Characteristics</i>	<ul style="list-style-type: none"> • Occurs through repetition • Routine • Control over immediate task, rules and structures • Well-understood context • Occurs at all levels in organization 	<ul style="list-style-type: none"> • Occurs through use of heuristics and insights • Non-routine • Development of differentiated structures, rules, etc. to deal with lack of control • Ambiguous context • Occurs mostly in upper levels
<i>Consequence</i>	<ul style="list-style-type: none"> • Behavioral outcomes 	<ul style="list-style-type: none"> • Insights, heuristics, and collective consciousness
<i>Examples</i>	<ul style="list-style-type: none"> • Institutionalized formal rules • Adjustments in management systems • Problem-solving skills 	<ul style="list-style-type: none"> • New missions and new definitions of direction • Agenda setting • Problem-defining skills • Development of new myths, stories, and culture

Levels of Learning
Source: Fiol & Lyles (1985)

Figure 2.2

Fiol & Lyles (1985) divide learning into two levels (see Figure 2.2). Lower-level learning includes learning that may be a repetition of past behaviors, or adjustments in what an organization does. They link this with Argyris and Schön's (1978) single-loop learning. Higher-level learning involves the development of complex rules and associates regarding new actions, which they equate with double-loop learning.

For Piaget, the key to learning lies in the mutual interaction of the process of *accommodation* (adapting our mental concepts based on experience in the world) and the process of *assimilation* (integrating our experience into existing mental concepts) (Piaget, 1970). Or, as Kolb (1984, p. 38) states:

Learning is the process whereby knowledge is created through the transformation of experience. This definition emphasizes several critical aspects of the learning process as viewed from the experiential perspective. First is the emphasis on the process of adaptation and learning as opposed to content or outcomes. Second is that knowledge is a transformation process, being continuously created and recreated, not an independent entity to be acquired or transmitted. Third, learning transforms experience in both its objective and subjective forms. Finally, to understand learning, we must understand the nature of knowledge, and vice versa.

In each of the above cases, the link between action and thought is defined as integral to the definition of learning. For our purposes, then, our definition of knowledge includes both the know-how and know-why of what is learned. We retain both parts of the definition because we are interested not only in the patterned responses that one picks up in mastering a skill (know-how), but the causal understanding (know-why) that accompanies it as well. Both are essential in order for learning to have any significant impact on the learner's ability to take effective action.

For example, a carpenter who has mastered the skills of woodwork without understanding the concept of building coherent structures like tables and houses can not utilize those skills very effectively. Similarly, a carpenter who possesses immense know-why about architecture and design but has no

complementary skills to produce any designs cannot put that know-why to effective use. We can think of *operational* learning leading to new ways of *doing* things and *conceptual* learning leading to new ways of *thinking* about things. The carpenter example illustrates this distinction between conceptual (know-why) learning and operational (know-how) learning which we will discuss again in the context of organizational learning.

Schein (1993) distinguishes three different types of learning, 1) knowledge acquisition and insight, 2) habit and skill learning, and 3) emotional conditioning and learned anxiety. The first two correspond with conceptual and operational learning, respectively. The third is an important distinction that affects both types of learning. Schein describes Pavlov's discovery in his research with dogs who were put in a green room, where they heard a bell ring and were given electric shocks. What the dog learned was to avoid green rooms and to cower or run whenever it heard a bell. Even after the shocks in the room were turned off, the dog would not go in, and so could not learn that anything had changed.

What that finding teaches us about learning is that avoidance behavior is learned through punishment, it does not encourage trial and error learning, and that people who are punished across a wide range of behavior are likely to limit themselves to very narrow safe ranges or become paralyzed for fear of making mistakes.

2.2.1.2 *Experiential Learning Model*

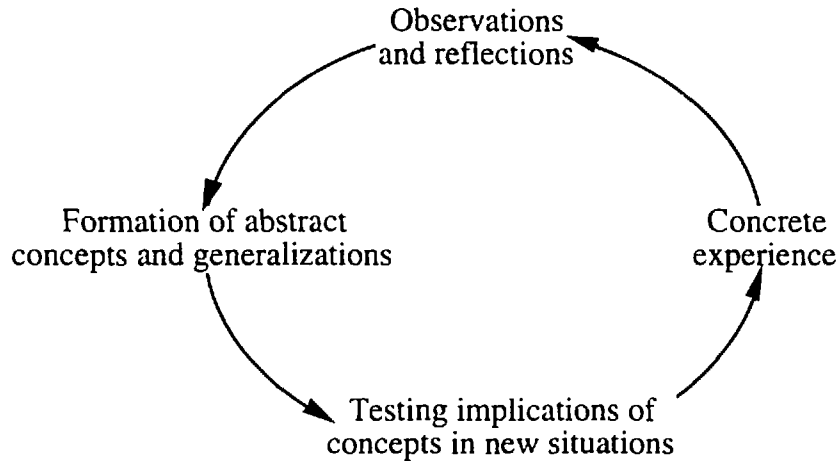
The work of Piaget on the nature of intelligence and how it develops is part of a triad of work that is collectively referred to as experiential learning (Kolb, 1984). Dewey's (1926) philosophical perspective of pragmatism in education and Lewin's (1951) work in translating phenomena to concepts and field

theory are the two other cornerstones in this school of thought. Experiential learning, as a body of learning theory, is the most consistent with addressing the two levels of learning in our definition above. As Kolb (1984, p. 20) writes:

This perspective on learning is called "experiential" for two reasons. The first is to tie it clearly to its intellectual origins in the work of Dewey, Lewin, and Piaget. The second reason is to emphasize the central role that experience plays in the learning process. This differentiates experiential learning theory from rationalist and other cognitive theories of learning that tend to give primary emphasis to acquisition, manipulation, and recall of abstract symbols, and from behavioral learning theories that deny any role for consciousness and subjective experience in the learning process.

In Kolb's view, learning should not be conceived in terms of outcomes (as behavioral theories tend to be) but rather as a continuous process grounded in experience. An outcome orientation would be analogous to what Freire (1974) referred to as the "banking" concept of education where teachers were the depositors and the students were the depositories of "things." Learning as a process, on the other hand, would be directed at stimulating inquiry and developing skills in the process of knowledge-getting (Bruner, 1966). In a way, the outcome view stresses the "product" of learning that is available for storing away while the process view focuses on the "acquisition" of new knowledge. As we shall see, both aspects need to be explicitly included in our model of individual learning.

Figure 2.3 contains a learning cycle that is representative of the experiential theory of learning. A person continually cycles through a process of moving from having concrete experiences, to making observations and reflections on that experience, to forming abstract concepts and generalizations based on those reflections, to testing those ideas in a new situation which leads to another concrete experience. This basic cycle has



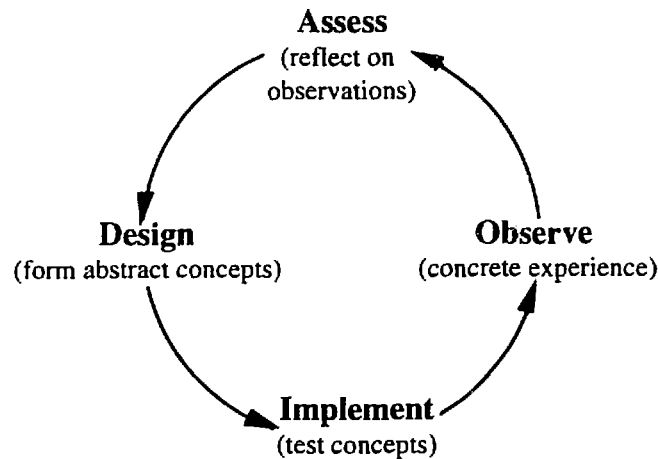
The Lewinian Experiential Learning Model

Figure 2.3

appeared in many forms in a variety of settings using different terms. In the Total Quality Management (TQM) literature, it shows up as the Deming cycle of Plan-Do-Check-Act or PDCA (Ishikawa, 1985) while Deming (1992) refers to it as the Shewhart cycle of Plan-Do-Study-Act. In Organizational Development (OD), Schein (1987c) calls his version the ORJI cycle (Observation, emotional Reaction, Judgment, and Intervention). In Action Science, Argyris & Schön (1978) refer to a Discovery-Invention-Production-Generalization cycle of learning.

At the risk of added confusion, I have chosen to base my model of individual learning on Kofman's (1992) version of the learning cycle as shown in Figure 2.4. The Observe-Assess-Design-Implement (OADI) cycle preserves the salient features of all the other versions mentioned above, but I believe the choice of terms have clearer connections to activities conducted in an organizational context. In the OADI cycle, a person experiences a concrete event and actively *observes* what is happening. She *assesses* (consciously or subconsciously) her experience by reflecting on her observations and then *designs* or constructs an abstract concept that seems to be an appropriate

response to the assessment. She tests the design by *implementing* in the concrete world which leads to a new concrete experience, commencing another cycle.



The O-A-D-I Cycle of Individual Learning

Figure 2.4

2.2.2 The OADI-IMM Model of Individual Learning

For the purposes stated in the beginning, the OADI cycle as drawn in Figure 2.4 is incomplete because it does not explicitly address the role of memory, which plays a critical role in linking individual to organizational learning. Integrating the role of memory will also require making an explicit distinction between the two aspects of learning described above, namely, conceptual and operational, and how the product of each type of learning is stored differently.

Memory can be modeled as a separate stage since a similar distinction is made between human learning and memory in the field of psychological research (Postman, 1976), implying that they are separate but related domains of inquiry. The two are distinct because learning has more to do with *acquisition* while memory has more to do with *retention* of whatever was acquired. In reality, however, separating the two processes is difficult because

they are tightly interconnected—what we already have in our memory affects what we learn and what we learn affects our memory. The combination of these two processes is fundamental to individual learning because learning without memory would require relearning everything as if it were the first time, while memory without learning would be like a storage drum with no inflow whose usefulness decays over time.³

The concept of memory is commonly understood to be analogous to a storage device where everything we perceive and experience is filed away. Powers (1973, p. 205) defines memory as “the storage and retrieval of information carried by neural signals” and draws an analogy to computer memory. Miller, et al. (1960) see memory as “plans for retrieval,” without which storage would be meaningless. In an ACT (Adaptive Control of Thought) production system, memory is divided into three types: working, declarative, and production (Anderson, 1983). Declarative memory contains permanent records that have been stored for retrieval. Working memory is basically declarative memory that has been retrieved for active use. Production memory is engaged when an action is being executed.

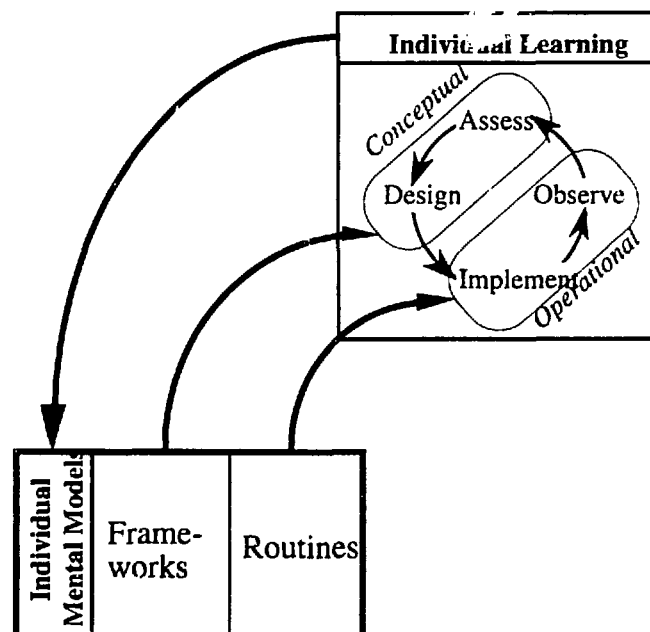
In each of the views of memory, the storage and retrieval process are most prominent and the active role that memory plays in the learning process itself is notably missing. Hence, we need to differentiate between “stored” memory such as baseball trivia and products of rote memorization and “active” structures that affect our thinking process and the actions we

³Restak (Restak, 1988) documents a pathological case of a patient whose ability to store anything new into his long-term memory was destroyed. He would see his wife and say how glad he was to see her for the first time that day even though he had just seen her moments before. When he turned his head and could no longer see her, no memory of that recent event was stored. Hence, when he turned his head back and faced her again, it was as if he were seeing her for the first time.

take. We will refer to these active structures as an individual's *mental models*.

2.2.2.1 Individual Mental Models

A model of individual learning (OADI-IMM) is proposed in Figure 2.5 in which "individual mental models" has been added to the OADI model to represent a particular type of memory. Forrester (1971) postulates that all of our decisions are made on the basis of mental models, and yet those models are fuzzy, incomplete, and imprecise. Norman (1983) contrasts mental models (what people really have in their heads that guides their actions) with conceptual models (tools meant to help us understand the actual reality). His findings suggest that there is often no direct and simple relationship between the two and that people's mental models are apt to be deficient in many ways. Senge (1990b, p. 174) describes mental models as "deeply held internal images



A Simple Model of Individual Learning: OADI-IMM Cycle

Figure 2.5

of how the world works” which have a powerful influence on what we *do* because they affect what we *see*. Troubles can arise when we take actions based on our mental models as if they *were* reality. In each case, the concept of mental models differs from the traditional notion of memory as static storage because they play an active role in what an individual sees and does.

Mental models represent a person’s view of the world, both explicit and implicit understandings. Mental models provide the context in which to view and interpret new material, and they determine how stored information is relevant to a given situation. They represent more than a collection of ideas, memories, and experiences—they are like the source code of a computer’s operating system, the manager and arbiter of acquiring, retaining, using, and deleting new information. But they are also much more than that because they are also like the programmer of that source code with the know-how to design a different source code as well as the know-why to choose one over the other.

The following excerpt is an example of mental models at work:

Without an ability to make implicit inferences, written and spoken discourse would be beyond anyone’s competence. In order to understand the following passage, it is necessary to make a variety of inferences:

The pilot put the plane into a stall just before landing on the strip. He just got it out of it in time. Wasn’t he lucky?

Every word in the first sentence, apart perhaps from the articles and prepositions, is ambiguous, and the appropriate meanings can be recovered only by making implicit inferences from linguistic context and general knowledge. (Johnson-Laird, 1983, p. 128)

In this example, the reader is able to draw the appropriate meanings not because the sentences match an identical set already in memory but because he or she has a mental model about the concept of flying an airplane which contains scripts and props (Bower, Black, & Turner, 1979; Winston, 1984) that provide the proper context for understanding the sentences. If the reader, for

some reason, invoked a different mental model, such as building a house, the words plane, stall, landing, and strip would mean very different things and the sentence would be incoherent.

Mental models not only help us make sense of what we see, they can also let us see only what makes sense to the mental model. Consider this example of mental models at work:

[H]ave you ever heard a statement such as, "Laura doesn't care about people," and wondered about its validity? Imagine that Laura is a superior or colleague who has some particular habits that others have noted. She rarely offers generous praise. She often stares off into space when people talk to her, and then asks, "What did you say?" She sometimes cuts people off when they speak. She never comes to office parties...From these particular behaviors, Laura's colleagues have concluded that she "doesn't care much about people." It's been common knowledge—except, of course, for Laura, who feels that she cares very much about people...Once Laura's colleagues accept as fact that she doesn't care about people, no one questions her behavior when she does things that are "noncaring," and no one notices when she does something that doesn't fit the [mental model]. The general view that she doesn't care leads people to treat her with greater indifference, which takes away any opportunity she might have had to exhibit more caring. (Senge, 1990b, pp. 192-193)

People's untested assumptions about Laura played an active role in creating the set of interactions which made their mental model of her a self-fulfilling prophesy. Whenever we take actions on the basis of stereotypes, we risk committing the same error as Laura's colleagues.

2.2.2.2 Frameworks and Routines

As mentioned earlier, there are two levels of learning—operational and conceptual—so we need to distinguish between two types of mental models. Operational learning represents learning at the procedural level, where one learns the steps that one must follow to complete a particular task. This know-how is captured as *routines* such as filling out entry forms, operating a piece of machinery, handling a switchboard, re-tooling a machine, etc. which capture the information about how to perform those tasks. Not only are routines accumulated and changed through operational learning, but they

affect operational learning process as well. Hence the two arrows going in both directions in the diagram represent this mutual influence.

Conceptual learning has to do with the thinking behind why things are done in the first place, sometimes challenging the very nature or existence of prevailing conditions, procedures, or conceptions and leading to new *frameworks* in the mental model. The new frameworks, in turn, can open up opportunities for discontinuous steps of improvement by reframing a problem in radically different ways.

To make the dynamics of the link between learning and mental models clearer, let's consider a simple example of driving a car home from work. Most of us, if we have lived in a place for a period of time and have worked in a place for a while, know the route home. We probably know several ways to get there, and have chosen one or two of them as the way we drive home most often. This is our routine, and we don't even have to think about it. The first time we made the trip, we attended to the details more carefully, perhaps consulted a map, looked closely at signs, and charted an unfamiliar path.

According to the model, we tested our chart, and observed how well it worked the first few times. If all went well, we might have fallen into a routine from the very first trip. We might have had to make adjustments if the streets were one-way (unmarked on our maps), or if, during rush hour, traffic was very heavy on the route we first took. After trying a few variations, we settled on our favorite way to get home, all things being equal. We began to think of it as "the best way to go home." That's the know-how, when things require little thought, because we know how. We sometimes think of it as being on "automatic pilot." *Acquiring* a routine involves much

more thought and attention to detail, but once it has been acquired, we're all set, unless things change.

The know-why, or conceptual part of the learning has more to do with how we think as individuals, and includes our values and personal preferences. One person might think the best way home is the "quickest" way home, while another might define "best" as the most scenic, or the least stressful. Yet another might hate the speed and volume of highways, contrasted with one who avoids stopping frequently at traffic lights. In the assessment phase, we consider all of the factors that are important to us and how our concrete experience matches those criteria. We then design new plans or renew our commitment to the original plan based on our experience. In either case, we form abstract concepts about the way home. We build a framework for "all things being equal," another for "in case of emergency," and another for "when I need to stop for errands in town."

Now, some of us may not spend too much time or consideration on these things, yet find we have a framework nonetheless. We discover these, often, when met with obstacles on the way home. Construction projects may block our normal route, and force us to take detours, either those planned by the highway department or one of our own design. All of a sudden, our routine has been challenged, blocked, and we experience distress, because our expectations have been thwarted. We feel we must go home this way, because this is the best way.

If we ask ourselves *why* we feel that strongly about it, we would discover the *thinking* behind our mental model about driving home this way, and could then reexamine the framework that we built. Clearly, there is always another way to go home. Our anxiety often stems from a belief that this is the only way. "Best," unchallenged, and entrenched in our routine, has grown to

“only.” And we probably don’t even recall when it happened, or why. We often take mental models for granted until we are confronted, questioned, or until they no longer provide expected results.

There are two processes: dynamics of solidification (cow paths) and dynamics of double loop learning (transformation). Frameworks are about perceptions that are created and designs that are discovered. Implement and observe are about routines. Although one can manage the flow of inputs, control the environment or manipulate the person in various ways, if that person’s view of the world remains unchanged, it is unlikely that any such actions will affect the type of learning that takes place. For our purposes then, *learning can be defined as increasing one’s capacity to take effective action.* Learning occurs when we know something new *and* we know how to translate it into action. We may or may not choose to implement the action, but we know *how* to. The same will hold true for organizational learning.

In the next section, we will examine ways in which organizational learning presents more complex and dynamic challenges than merely increasing the proportions of information and experience to match the numbers of people involved in the organization, and the reasons why this is so. We will also explore behavioral theories as well as theories of organizations as interpretation systems in order to develop an integrated model of organizational learning which provides for the necessary links and transfer mechanisms, and will illustrate the use of this model in an industrial setting.

2.3 ORGANIZATIONAL LEARNING

Organizational learning involves much more than a semantic shift from a singular to a plural reference of a generic learning process. The level of

complexity increases tremendously when we go from a single individual to a collection of diverse individuals. Although the meaning of the term “learning” remains essentially the same as in the individual case, the learning *process* is fundamentally different at the organizational level. A model of organizational learning has to resolve somehow the dilemma of imparting intelligence and learning capabilities to a non-human entity without anthropomorphizing it. In addition to capturing the learning process, a good model should also address questions such as what changes when an organization learns and what kinds of dysfunctions can inhibit their learning (March & Olsen, 1975).

2.3.1 Levels of Complexity

Moving from the individual human level to the organizational level adds at the least an order of magnitude increase in complexity in the unit of analysis. Issues such as motivation and reward, which are an integral part of human learning (Nuttin, 1976), become doubly complicated because the meaning of each depends, in part, on the type of organization as well as the individual.

Boulding (1956) outlined a hierarchy of complexity where the various empirical fields of “individuals” are arranged into nine distinct levels. The first level represents static structure or frameworks—the geography and anatomy of the universe. The ninth level represents a class of transcendental systems—“the inescapable unknowables.” In this hierarchy, the human level is at level seven and social organizations are at level eight. In describing this eighth level, Boulding (1956, p. 205) writes “the unit of such systems is not perhaps the person—the individual human as such—but the ‘role’—that part of the person which is concerned with the organization or situation in question...”

Thus, in the context of organizational learning, we are interested in the individual's role as a learner and not in the unique characteristics of that particular person as a human being. It matters less whether the person is male or female, conservative or liberal, introvert or extrovert. It matters more if he or she is a manager or an operator; technical or clerical.

2.3.2 The Individual-Organization Learning Dilemma

What do we mean by organizational learning? Various definitions have focused on different aspects of learning. In the early stages of an organization's existence, organizational learning is often synonymous with individual learning since it usually involves a very small group of people and the organization has minimal structure. As an organization grows, however, a distinction between the two levels of learning emerges, and somewhere in that process, a system for capturing learnings from its individual members evolves. The way in which an organization learns through individuals is a topic of growing interest, but little consensus. One of the main dilemmas shared by all who tackle this issue was posed by Argyris and Schön (1978, pp. 9-10):

There is something paradoxical here. Organizations are not merely collections of individuals, yet there are no organizations without such collections. Similarly, organizational learning is not merely individual learning, yet organizations learn only through the experience and actions of individuals. What, then, are we to make of organizational learning? What is an organization that it may learn?

Clearly, an organization learns through its individual members and, therefore is affected either directly or indirectly by individual learning. Argyris and Schön (1978) present a theory of action perspective where organizational learning takes place through individual actors whose actions are based on a set of shared models. They argue that most organizations have shared assumptions which protect the status quo, preclude challenging

people's troublesome or difficult qualities and characteristics, and provide silent assent to those attributions; hence, very little learning is possible. For example, when confronted with a leader of the organization's tendency to steamroll over any opposition to his or her ideas, people tend to accept it with resignation, as "the way X is," rather than to point out the occasions when the steamrolling occurs. Furthermore, we assume that the person is aware and doing it on purpose, or we assume that the person doesn't want to talk about it. We don't make our own mental models explicit. We don't test our assumptions with that person. Whenever we interact with this person, we "know" he or she will steamroll, so we act in ways that make it easy to steamroll.

There is little agreement on what constitutes "appropriate" learning; those actions or lessons that should be incorporated into an organization's memory. Standard operating procedures (SOP's)⁴, for example, are viewed as an important part of an organization's memory and a repository of its past learning (Argyris & Schön, 1978; Perrow, 1986; Winter, 1985). Hedberg, Nystrom, and Starbuck (1976), however, argue against SOP's because the use of old procedures that are inappropriate to radically different circumstances will preoccupy and delay the search for entirely new modes of behavior. Instead, they advocate minimal levels of consensus, contentment, affluence, faith, consistency, and rationality. Levitt and March (1988), on the other hand, caution against such fast, precise learning because it can lead to making mistakes faster and to specializing prematurely into inferior technologies.

⁴Throughout this chapter, I refer to organizational routines and standard operating procedures (SOP's) interchangeably. Both terms are meant to contain the general meaning of routines. (Cyert & March, 1963, Nelson & Winter, 1982)

In reality, both views are correct to a degree; the crux of the matter is knowing when organizational routines such as SOP's are appropriate and when they are not. As Winter (1985, p. 111) argues:

Routinized competence clearly does not dictate inattention to considerations that fall outside of the scope of the routines; in fact, it should make possible higher levels of attention to such considerations. But the wider the range of situations subsumed by the routines and the better the routinized performance, the fewer reminders there are that something besides routinized competence might on occasion be useful or even essential to survival.

This view is consistent with Perrow's (1986) praise for SOP's where he clearly saw a useful role for such organizational routines. They are viewed to be very useful in most cases as long as the appropriateness of their use is reviewed regularly and with each new application. But how does an organization decide when once-appropriate routines are no longer the correct actions to take? Can an organization anticipate such obsolescence of its SOP's or must it always learn by first making inappropriate decisions in the face of changing conditions? Are organizational SOP's different from individual routines? These are the types of issues which a model of organizational learning must address.

2.3.3 Organizations as Behavioral Systems

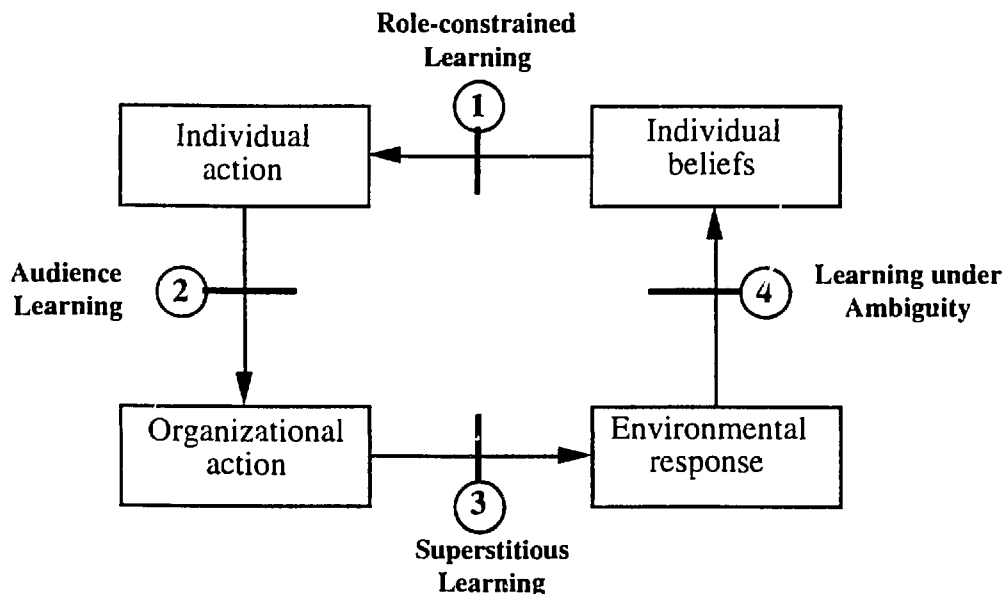
Simon (1981, p. 65) proposed the following provocative hypothesis:

A man, viewed as a behaving system, is quite simple. The apparent complexity of his behavior over time is largely a reflection of the complexity of the environment in which he finds himself.

This statement was accompanied by a couple of caveats. First, Simon limited the definition of man to the "thinking man" and analyzed cognition rather than behavior in general. His second caveat was to view an individual's memory to be more a part of the environment than the individual. That is, the memory is generated by the actions taken to adapt to the environment. This behavioral perspective can be extended to organizations. For example,

Cyert and March (1963) see the organization as an adaptively rational system which basically learns from experience. A firm changes its behavior in response to short-term feedback from the environment according to some fairly well-defined rules and adapts to longer-term feedback on the basis of more general rules. At some level in this hierarchy, they suggest, lie “learning rules.”

March and Olsen (1975) make a distinction between individual and organizational action in their four phase model of organizational learning (see Figure 2.6). In their model, individual actions are taken based on certain individual beliefs. These actions, in turn, lead to organizational action which produces some environmental response. The cycle is completed when the environmental response, in turn, affects individual beliefs. Tracing through this loop, we see that if the environmental response is static and unchanging, individual beliefs, actions and therefore organizational actions will also



March and Olsen's Model of Organizational Learning

Modified from March & Olsen (1975)

Figure 2.6

remain unchanged. If there are changes in the environment, however, individual beliefs about the nature of the environment *could* change, thus precipitating a different set of individual and organizational actions. This will, in turn, set off a whole new cycle of learning.

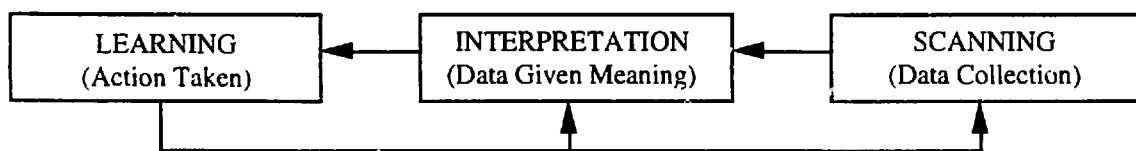
March and Olsen's model also addresses the issue of *incomplete learning cycles* where learning in the face of changing environmental conditions is impaired because one or more of the links is either weak or broken. They identify four cases where the learning cycle is incomplete and leads to dysfunctional learning. *Role-constrained* experiential learning can occur when individual learning has no effect on individual action because the circle is broken by the constraints of the individual's role. *Audience* experiential learning occurs when the individual affects organizational action in an ambiguous way. A third case is *superstitious* experiential learning where the link between organizational action and environmental response is severed. Thus, actions are taken, responses are observed, inferences are drawn, and learning takes place, but there is no real basis for the connections made between organizational action and environmental response.

The last incomplete cycle is one of experiential learning under *ambiguity*. Here, the individual affects organizational action which affects the environment, but the causal connections among the events are not clear. "Learning takes place and behavior changes; but a model of the process requires some ideas about the imputation of meaning and structure to events" (March & Olsen, 1975, p. 160). In other words, operational learning occurs, but whether conceptual learning occurs is dubious. Organizational learning, in order to be effective, must advance in a balanced process of complementary learnings in both know-how and know-why.

2.3.4 Organizations as Interpretation Systems

The behavioral view is consistent with the view of organizations as interpretation systems. Daft and Weick (1984) propose a model (see Figure 2.7) composed of three stages that represents the overall learning process of an organization, namely, scanning (data collection), interpretation (data given meaning), and learning (action taken). *Scanning* involves monitoring and obtaining data about the environment. *Interpretation* can be seen as a process of translating events and developing concepts consistent with prior understanding of the environment. *Learning* is defined as knowledge about the inter-relationship between the organization's actions and the environment as well as the actions that are taken on the basis of such knowledge.

This interpretation model is also consistent with Simon's hypothesis if we view the act of interpretation and learning to be analogous to the act of interacting with and adapting to the environment. The complexity of the actions taken (behaviors observed) is then largely determined by the complexity of the data collected from the environment. This implies that if the environment were utterly simple, the behavior of the organization would also be very simple, at least when viewed from outside the firm.



Relationships among Organizational Scanning, Interpretation, and Learning

Source: Daft & Weick (1984)

Figure 2.7

Daft and Weick (1984, p. 286) formally defined interpretation as “the process of translating events and developing shared understanding and

conceptual schemes among members of upper management.” Although they likened interpretation to learning a new skill by an individual, I will again separate know-why from skill and say that interpretation occurs more at the conceptual than the operational level. Their typology of four different interpretation types—undirected viewing, conditioned viewing, discovering, and enacting—is shown in Figure 2.8. The horizontal axis is a measure of an

ASSUMPTIONS ABOUT ENVIRONMENT	Unanalyzable	UNDIRECTED VIEWING Constrained interpretations. Nonroutine, informal data. Hunch, rumor, chance opportunities.	ENACTING Experimentation, testing, coercion, invent environ- ment. Learn by doing.
	Analyzable	CONDITIONED VIEWING Interprets within traditional boundaries. Passive detec- tion. Routines, formal data.	DISCOVERING Formal search. Questioning, surveys, data gathering. Active detection
		Passive	Active
ORGANIZATIONAL INTRUSIVENESS			

Types of Organizational Interpretation Systems

Source: Daft & Weick (1984)

Figure 2.8

organization’s willingness to look outside of their own boundaries. For example, a technology-focused company’s efforts may be inwardly directed (intensive research in core technologies) while a marketing-focused company’s efforts are outwardly focused (customer focus groups and market surveys). The two axes represent an organization’s assumptions about the world and its own role in it, the combination of which capture an organization’s world view or *Weltanschauung*. An organization’s *Weltanschauung*, defined as “a comprehensive philosophy of the world or of human life,” will determine how it interprets environmental responses,

whether it will act upon them, and what specific means it will employ if it chooses to act.

Ideally, to increase learning, organizations ought to move away from the undirected viewing type and move more towards the enacting type. Although there are some merits in the other two types—conditioned viewing and discovery—neither encourages the type of open experimentation and innovative thinking that the enacting type embraces. Managers in an enacting organization will behave more like researchers than the traditional managers because the only way to find out about an unanalyzable environment is through active experimentation and testing *in* the environment.

Non-adaptive organizations have short lives; most organizations appear to fall into this category. For example, fifty-year-old corporations represent only two percent of those created, and approximately 30% of those fifty-year-old corporations can be expected to disappear within ten years (Starbuck, 1983). In order to maximize a firm's chance for survival, it must be nimble in its ability to think *and* act. How it views the environment and its own role within it will have a (potentially) large impact on its ability to learn and adapt to changing conditions.

2.3.5 The Missing Link: from Individual to Organizational Learning

Analogies of the human learning process have been drawn with respect to organizations (Beer, 1972) and various theories of organizations have counterparts at the individual level. For example, adaptation theory (Miles, et al., 1978; Miller & Friesen, 1980; Tushman & Anderson, 1986) can be viewed as analogs of individual stimulus-response theories (Hilgard & Bower, 1966), and strategic choice models (Hrebiniak & Joyce, 1985) having similarities with

psychodynamic theories (Hilgard & Bower, 1966). If a distinction between organization and individual is not made explicit, however, a model of organizational learning will either obscure the actual learning process by ignoring the role of the individual (and anthropomorphize organizations) or become a simplistic extension of individual learning by glossing over organizational complexities.

As explained earlier, the integrated model of organizational learning developed in this chapter builds upon the work of March & Olsen (1975) and Argyris & Schön (1978), among others, but it goes beyond both models by explicitly integrating the transfer process from individual learning to organizational learning via mental models. In the next section, I will attempt to build an integrated model that will address some of the existing shortcomings.

2.4 AN INTEGRATED MODEL OF ORGANIZATIONAL LEARNING

Figure 2.9 shows an integrated model of organizational learning (OADI-SMM model) which encompasses all of the elements discussed in the previous sections into a cohesive framework for researching organizational learning. It builds on and extends the March and Olsen model (see Figure 2.6).⁵ The OADI-SMM integrates individual learning into the organizational learning process and addresses the issue of the transfer of learning through the exchange of mental models from individual to shared, and vice versa. Analogous to individual learning, *organizational learning is defined as increasing an organization's capacity to take effective action.* The distinction between conceptual and operational learning and between *Weltanschauung*

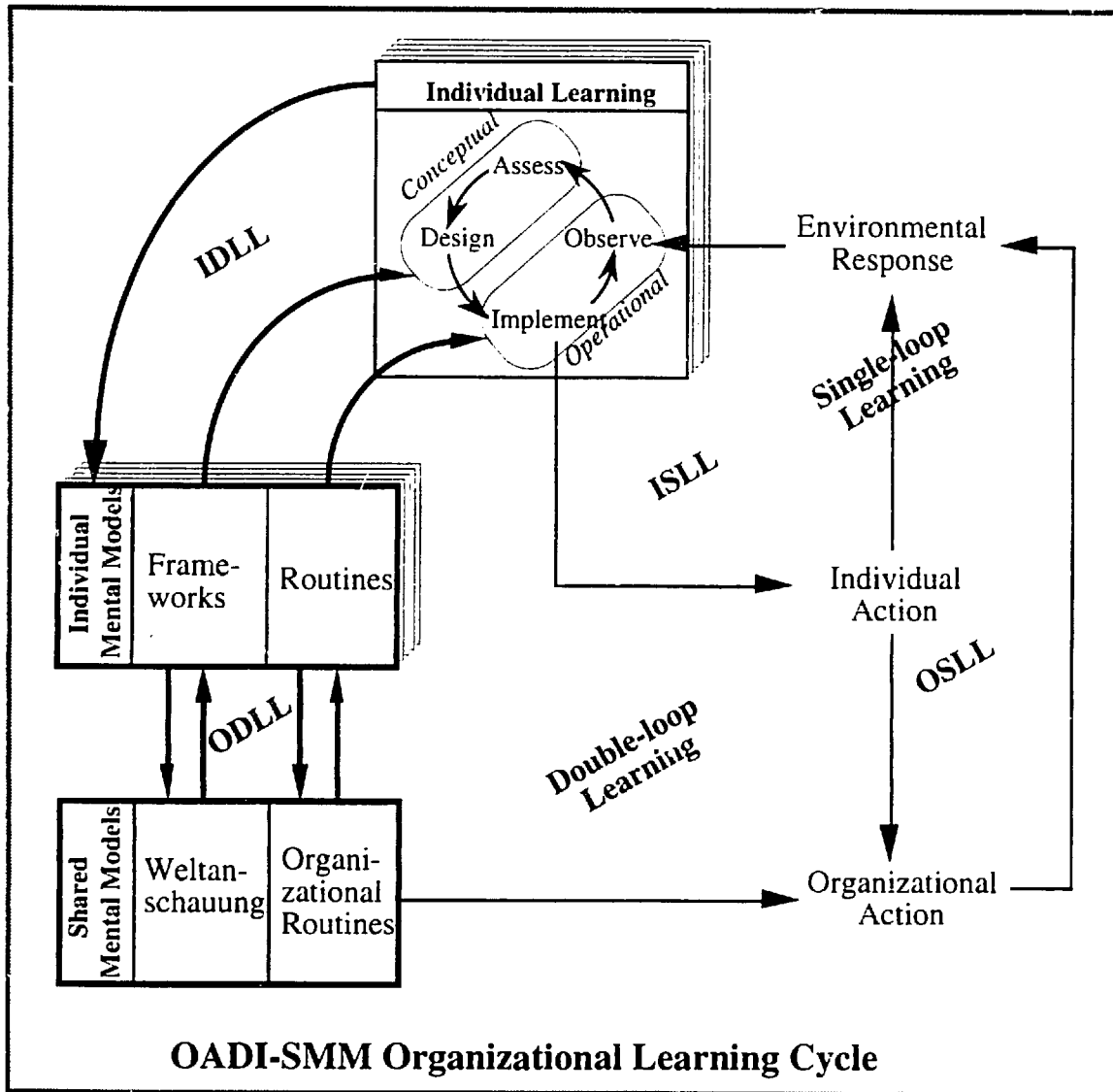
⁵The four "disconnects" identified in March and Olsen's model have been transposed onto this model and are shown in Figure 2.10.

and organizational routines are also integrated throughout the different stages. A box has been drawn around the diagram to emphasize that the whole model is required to represent organizational learning.

2.4.1 The Role of Individuals in Organizational Learning

In the OADI-IMM model, learning is always rooted in concrete experience in the real world (implement and observe stages), so the individual is constantly taking actions and observing his experience. A distinct Individual Action stage that lies outside of the OADI-IMM cycle is modeled to represent those actions that have more direct consequence for the organization than the minute details of all actions taken by an individual. A person may, for example, learn about a new dance step and decide to enroll in a dance class. That person will presumably learn a great deal and take a lot of actions in their individual learning process. We would not, however, include those actions as being relevant Individual Actions for organizational learning nor for individual learning that would have relevance for the organization.

In the OADI-SMM model, I have substituted individual beliefs in March and Olsen's model with the OADI-IMM model of individual learning. The individual learning cycle is the process through which those beliefs change and those changes are then codified in the individual mental models. The cycles of individual learning affect learning at the organizational level through its influence on the *shared mental models* of the organization. In this model, the role of the individual in organizational learning is clear and explicit. Organizations can learn only through its members, but it is not dependent on any specific member (as denoted by the multiple boxes representing individual learning). Individuals, however, can learn without the organization.



An Integrated Model of Organizational Learning: OADI-SMM Cycle

Figure 2.9

Figure 2.9 incorporates Argyris and Schön's concept of single-loop and double-loop learning on both the individual and organizational level. Double-loop learning involves surfacing and challenging deep-rooted assumptions and norms of an organization that have previously been inaccessible, either because they were unknown or known but undiscussable. Individual double-loop learning (IDLL) is traced out in Figure 2.9 as the process through which individual learning affects individual mental models, which in turn affect future learning. Organizational double-loop learning

(ODLL) occurs when individual mental models become incorporated into the organization through shared mental models, which can then affect organizational action. In both cases, double-loop learning provides opportunities for discontinuous steps of improvement where reframing a problem can bring about radically different potential solutions.

2.4.1.1 Groups and Organizational Learning

The model I have proposed includes the individual learner embedded in an organization, but it does not explicitly deal with the issue of groups. While such influences as the development and enforcement of group norms (Feldman, 1984), group polarization (Isenberg, 1986) and other factors have an effect on individuals (Hackman & Oldham, 1976), group effects are not explicitly included in this model.

In terms of mental models, however, if we view a group as a mini-organization, whose members contribute to the group's shared mental models, then the model can represent group learning as well as organizational learning. Bushe and Shani (1991) explore how a group functioning as a steering committee can provide a parallel learning structure within an organization. A group can then be viewed as a collective individual with its own set of mental models who contributes to the organization's shared mental models and learning. This is consistent with the notion that groups themselves are influenced by organizational structure and type of management style such as control vs. commitment (Walton & Hackman, 1986) and, therefore, can be treated as if they were "extended individuals."

2.4.2 The Transfer Mechanism: Shared Mental Models

As we did at the individual level, we want to distinguish between organizational memory and shared mental models. Again, the concepts of static storage and active influence can illuminate the distinction. Organizational memory, broadly defined, includes everything that is contained in an organization that is somehow retrievable. Thus, storage files of old invoices are part of that memory. So are copies of letters, spreadsheet data stored in computers, and the latest strategic plan, as well as what is in the minds of all organizational members. The problem again, is that although it may be a comprehensive definition, it is not very useful in the context of organizational learning.

The parts of an organization's memory that are relevant for organizational learning are those that play an active role in defining what an organization pays attention to, how it chooses to act, and what it chooses to remember from its experience. This is what we mean by mental models and shared mental models. They may be explicit or implicit, tacit or widely recognized, but they have the capacity to affect the way an individual or organization views the world and the actions that are taken. Organizational learning is dependent on individuals improving their mental models; making those mental models explicit enough to be *shared mental models* allows organizational learning to be *independent* of any specific individual. Why are we putting so much emphasis on mental models? *Because the mental models in individuals' heads are where a vast majority of an organization's knowledge (both know-how and know-why) lies.*

To highlight the importance of mental models in organizations, let us carry out the following two thought experiments. Scenario 1: Imagine an organization in which all the physical records of that organization

disintegrated overnight. Suddenly, there are no more reports, no computer files, no employee record sheets, no operating manuals, no calendars—all that remain are the people, buildings, capital equipment, raw materials, and inventory. Scenario 2: Now imagine an organization where all the people simply quit showing up for work. The organization is left intact in every way, but all the people are gone. New people, who are similar in many ways with the former workers, but who have no familiarity with that industrial context, show up. Of the two cases described, which organization will be easier to rebuild to its former status so that it can continue to take actions and to learn?

The obvious and safe answer would be to say “it depends,” but that is not a satisfactory answer. Thinking through the full ramifications of each scenario, one could conclude that having all the people intact will be easier than having the systems and records intact. In terms of the model presented in Figure 2.10, scenario 1 is equivalent to eliminating the organizational memory (including the shared mental models), whereas scenario 2 can be likened to obliterating the individual mental models and their *linkages* to the shared mental models. Thus, when new individuals are put into the organizations, they come in with their own individual learning and mental models stages which have no connections to the remaining organizational memory (sans shared mental models).

Even in the most bureaucratic of organizations, despite the preponderance of written SOP's and established protocols, there is much more about the firm that is unsaid and unwritten; its essence is embodied more in the people than in the systems. Comparatively little is put down on paper or stored in computer memories (Forrester, 1993; Simon, 1991). These intangible and often invisible assets of an organization reside in individual mental models which collectively contribute to the shared mental models.

The shared mental models are what makes the rest of the organizational memory usable. Without these mental models, which include all the subtle interconnections that have been developed among the various members, an organization will be incapacitated in both learning and action.

This assertion is not as radical as it may sound. Actually there is some empirical support for the above assertion in the form of turnover data. As everyone knows, high turnover is costly in terms of time and money since new recruits have to “learn the ropes” while being paid and consuming an experienced person’s time. In fact, the second scenario described above is precisely the case of high turnover taken to an extreme. Companies with a 40-50% annual turnover rate have a hard time accumulating learning because their experience base is continually being eroded.

An example of scenario 1 would be the case of radical changes brought about by a new CEO (e.g., Geneen at IT&T, Welsch at General Electric) or by a hostile takeover. In many such cases, the organization is completely gutted of its previous management style, procedures, and structures and replaced with a different one altogether. Although transitions are times of great upheaval, the organization as a whole usually remains intact.

2.4.2.1 *Weltanschauung and Organizational Routines*

As stated earlier, mental models are not merely a repository of sensory data; they are *active* in that they build theories about sensory experience. Each mental model is a clustering or an aggregation of data which prescribes a viewpoint or a course of action. Conceptual learning creates changes in frameworks leading to new ways of looking at the world. Operational learning produces new or revised routines that are executed in lieu of the old ones. The revised mental models contain not only the new frameworks and

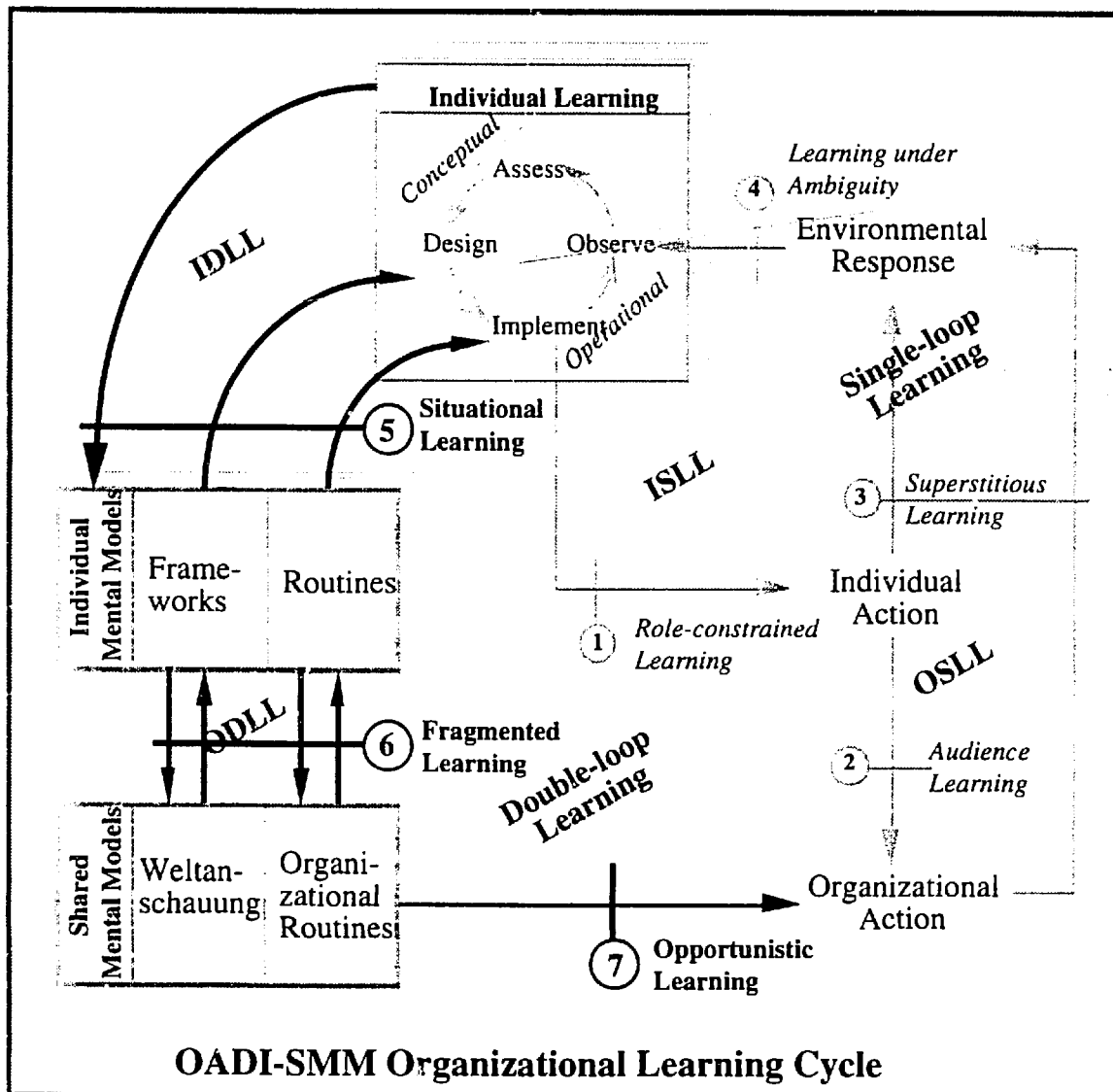
routines, but also knowledge about how the routines fit within the new framework.

Individual mental frameworks become embedded into the organization's own *Weltanschauung*. The organization's view of the world slowly evolves to encompass the current thinking of the individuals within. In similar fashion, individual routines that are proven to be sound over time become standard operating procedures. Like an individual driving a car, the routines become the organization's auto pilot reflexes. The strength of the link between individual mental models and shared mental models is a function of the influence level of a particular individual or group of individuals. In the case of a CEO or upper management, influence can be high due to the power inherent in the positions. Similarly, a united group of hourly workers can have a high degree of influence due to their size.

For example, Proctor & Gamble's *Weltanschauung* can be characterized as one where the company views itself as a strong community player, has a sense of community responsibility, and believes in the importance of its corporate image as well as that of its product brands. Its *Weltanschauung* is also a reflection of its culture, its deep-rooted assumptions and its artifacts, as well as overt behavior rules about what is the "right" thing to do. All of these things moderate its decision-making as it encounters unpredictable, non-routine events. Their Standard Operating Procedures (SOPs), on the other hand, may include things like a marketing plan to launch a new product, procedures for paying suppliers, employee performance reviews, and hiring criteria. These organizational routines allow the organization to respond to routine needs in very predictable ways.

2.4.3 Incomplete Learning Cycles

In March and Olsen's (1975) model they identified four possible disconnects whereby organizational learning cycle would be incomplete. In our integrated model, I have identified three additional types of incomplete learning cycles which affect organizational learning: situational, fragmented, and opportunistic (see Figure 2.10).



Incomplete Learning Cycles

Figure 2.10

2.4.3.1 *Situational Learning*

An individual encounters a problem, improvises on the spot and solves the problem, and moves on to the next task. *Situational* learning occurs when the individual simply “forgets” or does not codify it for later use, that is, when the link between individual learning and individual mental models is severed. Regardless of whether the learning occurred at the conceptual or operational level, it does not change the person’s mental models and therefore has no long term impact—the learning is situation specific. Since the individual’s mental model was not changed, the organization does not have a way of absorbing the learning, either.

Examples of situational learning fall under the broad category of “crisis management,” where each problem encountered is solved but with no learning carried over to the next case. Quality improvement is a counter example which focuses on minimizing situational learning through a systematic data gathering, analysis, and standardization process.

2.4.3.2 *Fragmented Learning*

There are many instances where individuals learn, but the organization as a whole does not. That is, when the link between individual mental models and shared mental models is broken, *fragmented* learning occurs. Organizational learning is fragmented among isolated individuals (or groups), whereby loss of the individuals means loss of the learning as well. Individual mental models are changing, but those changes are not reflected in the organization’s memory, and thus, there is no cohesive picture of what is occurring at the individual level.

Universities are a classic example of fragmented learning. Professors within each department may be the world’s leading experts on management,

finance, operations, and marketing, but the university as an institution cannot apply it to the running of its own affairs. Very decentralized organizations that do not have the requisite networking capabilities to keep the various parts connected are also susceptible to fragmented learning.

2.4.3.3 *Opportunistic Learning*

There are times when organizations purposely try to bypass their standard procedures because their established ways of doing business are seen as an impediment for a particular task. They want to sever the link between shared mental models and organizational action in order to seize an opportunity that cannot wait for the whole organization to change (or it may not be desirable to do so). *Opportunistic* learning occurs when organizational actions are taken based on an individual's (or small group of individuals') actions and not on the widely shared mental models (values, culture, myths, or SOP's) of the organization.

The use of skunk works to develop the IBM personal computer is a good example, where they chose to bypass their normal bureaucratic structure and create an entirely separate, dedicated team to develop the PC, which it was able to do in record time. General Motors' creation of Saturn is another example on a grander scale as are joint ventures, when appropriately structured.

2.5 SUMMARY

The integrated model of organizational learning presented in the preceding section is not a simple one, but its complexity is justified for the following reasons. First, the link between individual and organizational learning needed to be made explicit, which requires including the learning process of the individual. Second, the distinction between conceptual and operational

learning is necessary to capture the differences between learning to do things differently versus learning to think about things in new and different ways. Either type of learning can occur without the other, but both types need to occur for effective organizational learning. Third, an organization's view of the world—*Weltanschauung*—plays an important role in its learning cycle. The *Weltanschauung* affects how individuals within the organization interpret environmental responses, how each translates his or her own mental models into actions, as well as how the organization translates its shared mental models into action.

In the above model of organizational learning, individual mental models play a pivotal role in the whole learning cycle, yet that is precisely one of the areas in which we know the least and where there is little to observe. One challenge is to find ways to make these mental models explicit; another is to manage the way these mental models are transferred into the organizational memory. Clearly, this involves creating new devices and tools for capturing a wide range of knowledge, some of which we are not even consciously aware.

One of the reasons mental models are so elusive is that often, people are not even aware of them. If all one is exposed to is one's own mental model, then one simply thinks, "This is the way the world is. This is the truth." A major hurdle is always the clash between holders of different mental models who can only see that the other person is wrong and must be shown the error of his or her thinking. The attempt to articulate the way the organization currently operates is itself an exercise in surfacing mental models. More opportunities to explore and describe the inner workings of organizations in these ways are needed to complete the learning cycles and enhance organizational learning.

Chapter 2 References

- Anderson, John R. (1983). *The Architecture of Cognition*. Cambridge, MA: Harvard University Press.
- Argyris, Chris, & Schön, Donald (1978). *Organizational Learning: A Theory of Action Perspective*. Reading, MA: Addison-Wesley.
- Beer, Stafford (1972). *Brain of the Firm*. New York: John Wiley and Sons.
- Boulding, Kenneth E. (1956). General systems theory—The skeleton of science. *Management Science*, 2(3), 197-208.
- Bower, G. H., Black, J. B., & Turner, T. J. (1979). Scripts in Memory for Text. *Cognitive Psychology*, 11, 177-220.
- Bruner, Jerome (1966). *Toward a Theory of Instruction*. New York: W.W. Norton.
- Bushe, Gervase R., & Shani, A.B.(Rami) (1991). *Parallel Learning Structures*. Reading, MA: Addison-Wesley Publishing Co.
- Cohen, Michael D. (1991). Individual Learning and Organizational Routine: Emerging Connections. *Organization Science*, 2(1), 135-139.
- Cyert, Richard M., & March, James G. (1963). *A Behavioral Theory of the Firm*. Englewood Cliffs, NJ: Prentice-Hall.
- Daft, R.L., & Weick, K.E. (1984). Toward a model of organizations as interpretation systems. *Academy of Management Review*, 9(2), 284-295.
- Deming, Edwards (1992). *Quality, Productivity and Competitive Position*. Ford Quality Education and Training Center.
- Dewey, John (1926). *Experience and Nature*. Chicago: Open Court Publishing.
- Duncan, Robert B. (1974). Modifications in decision structure in adapting to the environment: Some implications for organizational learning. *Decision Sciences*, 5, 705-725.
- Dutton, Jane E., & Jackson, Susan E. (1987). Categorizing Strategic Issues: Links to Organizational Action. *Academy of Management Review*, 12(1, January), 76-90.
- Feldman, Daniel C. (1984). The Development and enforcement of group norms. *Academy of Management Review*, 9(1), 47-53.
- Fiol, C.M., & Lyles, M.A. (1985). Organizational learning. *Academy of Management Review*, 10(4), 803-813.
- Forrester, Jay W. (1971). *World Dynamics*. Cambridge, MA: Productivity Press.
- Forrester, Jay W. (1993). Systems Dynamics and the Lessons of 35 Years. In K. B. DeGreene (Eds.), *Systems-Based Approach to Policymaking*. Norwell, MA: Kluwer Academic Publishers.
- Freire, Paulo (1974). *Pedagogy of the Oppressed*. New York: Continuum.
- Hackman, J. R. , & Oldham, Greg R. (1976). Motivation through the Design of Work: Test of a Theory. *Organizational Behavior and Human Performance*, 16, 250-279.
- Hedberg, Bo. (1981). How organizations learn and unlearn. In P. C. Nystrom & W. H. Starbuck (Eds.), *Handbook of Organizational Design*. London: Oxford University Press.
- Hedberg, Bo L.T., Nystrom, Paul C. , & Starbuck, William H. (1976). Camping on seesaws: Prescriptions for a self-designing organization. *Administrative Science Quarterly*, 21, 41-65.
- Hilgard, Ernest R. , & Bower, Gordon H. (1966). *Theories of Learning*. New York: Appleton-Century-Crofts.
- Hrebiniak, Lawrence G. , & Joyce, William F. (1985). Organizational adaptation: Strategic choice and environmental determinism. *Administrative Science Quarterly*, 20, 226-240.
- Huber, George, P. (1991). Organizational Learning: The Contributing Processes and the Literature. *Organization Science*, 2(1), 88-115.
- Isenberg, Daniel J. (1986). Group polarization: a critical review and meta-analysis. *Journal of Personality and Social Psychology*, 50(6), 1141-1151.
- Ishikawa, Kaoru (1985). *What is Total Quality Control?* Englewood Cliffs, NJ: Prentice-Hall.
- Jaques, Elliott. (1989). *Req:isite Organization*. Arlington, VA: Cason-Hall Associates.

- Johnson-Laird, P.N. (1983). *Mental Models*. Cambridge, MA: Harvard University Press.
- Kim, Daniel H. (1993). Creating Learning Organizations: Understanding the Link between Individual and Organizational Learning. Forthcoming in *Sloan Management Review* (Fall).
- Kofman, Fred (1992). Lecture Slides. In Cambridge, MA: MIT Sloan School of Management.
- Kolb, David A. (1984). *Experiential Learning: Experience as the Source of Learning and Development*. Englewood Cliffs, NJ: Prentice-Hall.
- Levitt, Barbara, & March, James G. (1988). Organizational Learning. *Annual Review of Sociology*, 14, 319-340.
- Lewin, Kurt. (1951). *Field Theory in Social Science*. New York: Harpers & Brothers.
- March, James G., & Olsen, Johan P. (1975). The Uncertainty of the Past: Organizational Learning Under Ambiguity. *European Journal of Political Research*, 3, 147-171.
- Miles, Raymond E., & Randolph, W. A. (1980). Influence of Organizational Learning Styles on Early Development. In J. R. Kimberly & R. H. Miles (Eds.), *The Organizational Life Cycle*. San Francisco: Jossey-Bass.
- Miles, Raymond E., Snow, Charles C., Meyer, Alan D., & Coleman, Henry J., Jr. (1978). Organizational Strategy, Structure, and Process. *Academy of Management Review*, 3(3), 546-563.
- Miller, Danny , & Friesen, Peter H. (1980). Momentum and revolution in organizational adaptation. *Academy of Management Review*, 3, 546-562.
- Miller, George A. (1956). The Magic Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychology Review*, 63(2), 81-97.
- Miller, George A., Galanter, Eugene, & Pribram, Karl H. (1960). *Plans and the Structure of Behavior*. New York: Henry Holt.
- Nelson, R., & Winter, S. (1982). Organizational Capabilities and Behavior. In R. Nelson & S. Winter (Eds.), *An Evolutionary Theory of Economic Change*. Cambridge, MA: Harvard University Press.
- Norman, Donald A. (1983). Some Observations on Mental Models. In D. Gentner & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Nuttin, Joseph R. (1976). Motivation and Reward in Human Learning: a Cognitive Approach. In W. K. Estes (Eds.), *Handbook of Learning and Cognitive Processes*. Hillsdale, NJ: Erlbaum.
- Perrow, Charles (1986). *Complex Organizations: A Critical Essay*. New York: Random House.
- Piaget, Jean (1970). *Structuralism*. New York: Basic Books.
- Postman, Leo (1976). Methodology of human learning. In W. K. Estes (Eds.), *Handbook of Learning and Cognitive Processes*, 3. New York: John Wiley and Sons.
- Powers, William T. (1973). *Behavior: the Control of Perception*. Chicago: Aldine Publishing.
- Restak, Richard M. (1988). *The Mind*. New York: Bantam Books.
- Roth, George (1992). *Organizational-Level Learning as Culture and Routines: A Model for Empirical and Theoretical Development*. Unpublished manuscript, MIT Sloan School of Management.
- Schein, Edgar (1993). How Can Organizations Learn Faster? The Challenge of Entering the Green Room. *Sloan Management Review*, Winter, 85-92.
- Schein, Edgar H. (1987c). *Process Consultation Volume II: Lessons for Managers and Consultants*. Reading, MA: Addison-Wesley.
- Senge, Peter M. (1989). *Organizational Learning: New Challenges for System Dynamics*. Cambridge, MA: MIT Sloan School of Management.
- Senge, Peter M. (1990b). *The Fifth Discipline*. New York: Doubleday.
- Simon, Herbert A. (1957). *Models of Man*. New York: Wiley.
- Simon, Herbert A. (1981). *Sciences of the Artificial*. Cambridge, MA: MIT Press.
- Simon, Hebert A. (1991). Bounded Rationality and Organizational Learning. *Organization Science*, 2(1), 125-134.
- Sims, Henry P., Gioia, Dennis A., & Associates, & (1986). *The Thinking Organization*. San Francisco: Jossey-Bass.

- Starbuck, William H. (1983). Organizations as Action Generators. *American Sociological Review*, 48, 91-102.
- Tushman, Michael L. , & Anderson, Philip (1986). Technological discontinuities and organizational environments. *Administrative Science Quarterly*, 31, 439-465.
- Tversky, A. , & Kahneman, D. (1974). Judgment under Uncertainty: Heuristics and biases. *Science*, 1985, 1124-1131.
- Tversky, A. , & Kahneman, D. (1987). Rational choice and the framing of decisions. In R. M. Hogarth & M. W. Reder (Eds.), *Rational Choice: The contrast between economics and psychology*. Chicago: University of Chicago Press.
- Walton, R.E. , & Hackman, J.R. (1986). Groups under contrasting management strategies. In P. S. Goodman (Eds.), *Designing Effective Work Groups*. San Francisco: Jossey-Bass.
- Winston, Patrick (1984). *Artificial Intelligence*. Reading, MA: Addison-Wesley.
- Winter, S.G. (1985). The case for 'mechanistic' decision making. In H. Pennings (Eds.), *Organizational Strategy and Change*. (pp. 99-113). San Francisco: Jossey-Bass.

Chapter 3

Representing Mental Models

"The world exists in forms that we establish and define."

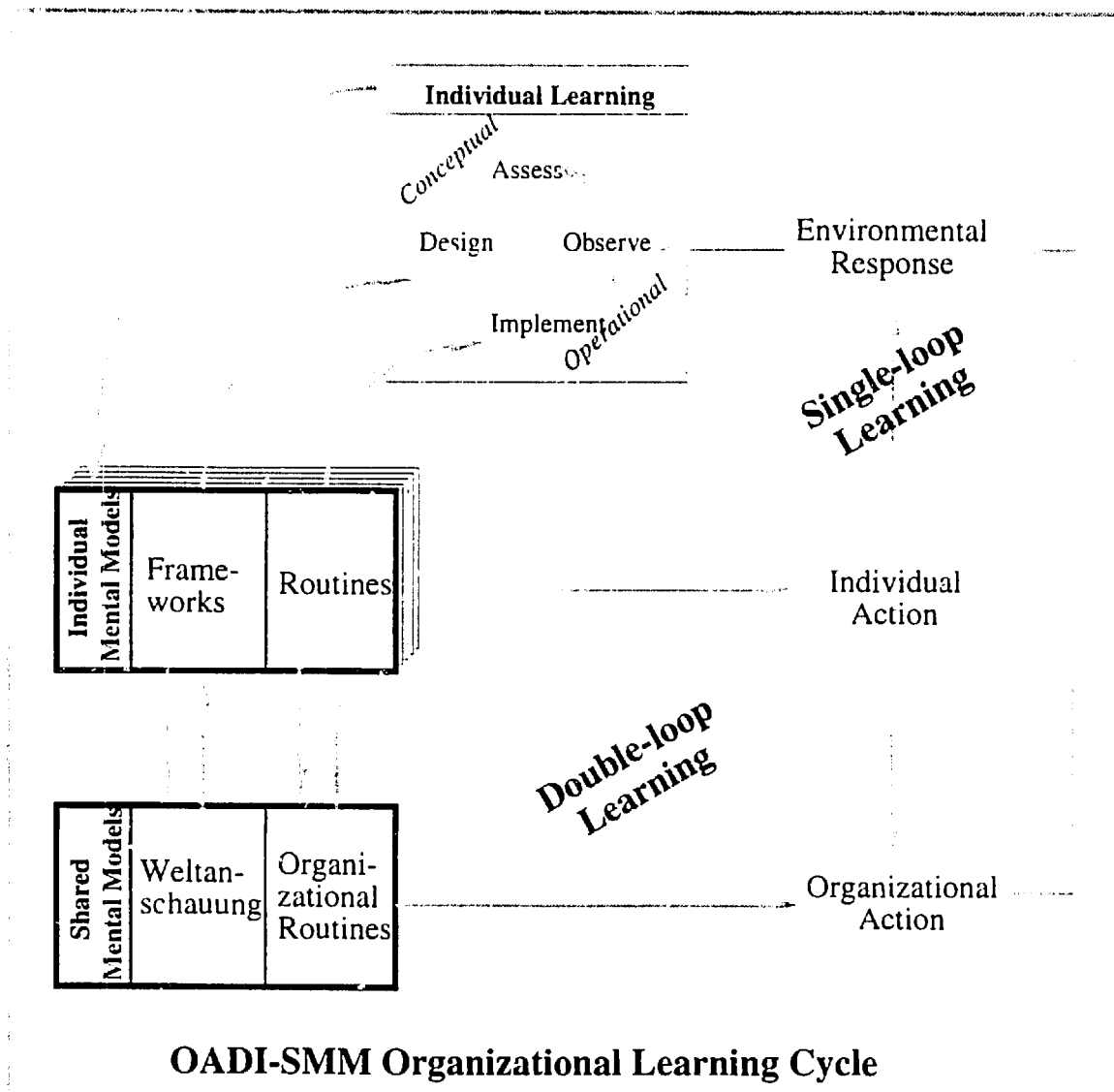
—Oswald Kulpe

3.0 INTRODUCTION

Chapter 2 laid out a framework for integrating individual learning with organizational learning in which mental models play a central role. In this chapter and the next, we will explore in greater depth what we mean by mental models. Within the framework of the organizational learning model presented in Chapter 2, this chapter focuses primarily on operationally representing what mental models are (see Figure 3.1). In Chapter 4, we will focus on the methodology of mapping individual mental models from individual learning and moving from individual mental models to shared mental models.

We should note that when talking about mental models, we cannot say anything definitive about mental models, or even assert that people actually have "mental models." However, we recognize that mental models are a useful construct for us to use when referring to the knowledge that people contain in their heads. Therefore, we will be careful to maintain the distinction between the use of the term "mental models"—the implicit

knowledge people have in their heads—and the use of the term “mental model representations”—the explicit representations of those mental models.



Individual and Shared Mental Models

Figure 3.1

Representing mental models requires having appropriate representational schemes that allow one to clarify one's understanding (implicit mental model) and make it explicit for others to also understand (see Figure 3.2). In other words, we want to operationalize an individual's

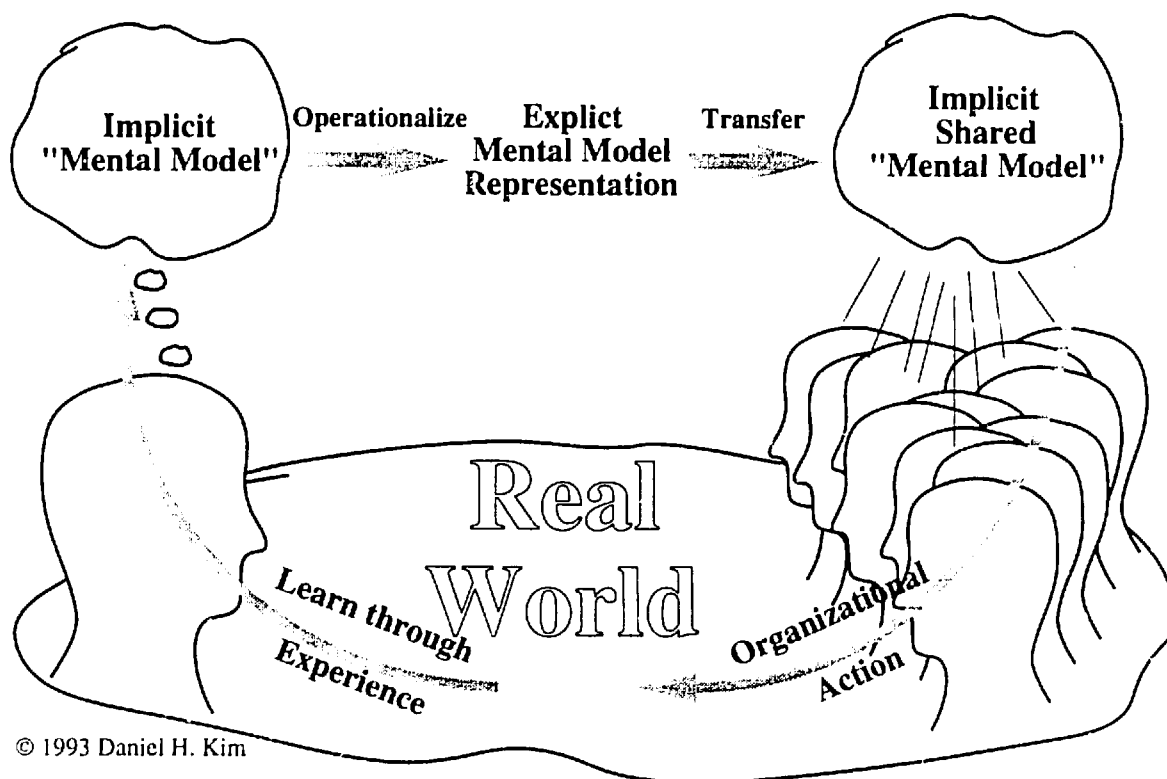
implicit mental model into an explicit representation that will make it accessible and transferable to others.

In many contexts, verbal representations are sufficient for capturing most of the salient features of a learning experience. There are contexts, however, where the English language is not particularly well-suited for describing a phenomenon of interest. I will suggest that verbal language and certain graphics-language representations (e.g., typical process flow charts) work well in addressing *detail* complexity, while a different representational scheme is needed for addressing *dynamic* complexity.¹ Enhancing our ability to deal with dynamic complexity is important because most organizational problems of significance are a result of dynamic complexity, not only detail complexity.

We begin by examining the literature on mental models, first in the physical domain (physical representations, analogies, and autonomous objects), and then in the organizational realm (scripts, stories, schemas, influence diagrams, cause and effect diagrams, etc.). A set of criteria for assessing mental model representations is then drawn from the literature. The issue of usefulness of those representations in different domains is raised, specifically when dealing with detail complexity and dynamic complexity. We then develop a typology that distinguishes between different representational forms (e.g., verbal, graphics-language, and mathematical) and between domains of usefulness (e.g., detail complexity and dynamic complexity). System dynamics mapping representations (causal loop diagrams, systems archetypes, computer models) are positioned as having

¹Senge (1990b) makes an important distinction between *dynamic* complexity and *detail* complexity of systems. A system can have hundreds, perhaps thousands, of parts that have to be managed, but the dynamics of the whole system may be relatively simple. On the other hand, a system with only a dozen or so pieces can be extremely complex and difficult to manage (Sterman, 1989). The complexity lies in the nature of the inter-relationships among the parts whose cause-effect relationships are highly nonlinear and distant in space and time.

strengths in the domain of dynamic complexity. Finally, we present systems archetypes as a useful, mass-consumable set of mental model representations for addressing dynamically complex issues and for building shared mental models. Five of these archetypes are presented (Drifting-Goals, Fixes-that-Fail, Limits-to-Success, Shifting-the-Burden, Tragedy-of-the-Commons), with a summary table of them presented as *dynamic scripts*.²



Explicit Mental Model Representations

Figure 3.2

3.1 LITERATURE ON MENTAL MODELS

If what matters is not reality, but perceptions of reality, then fundamental to learning is a shared mental model and language. Kepner and Tregoe (1965) emphasize the importance of the thought process behind managerial

²Gary Burchill suggested the use of the term "dynamic scripts" to refer to systems archetypes.

decision-making and believe that making that process accessible is of key importance in advancing learning at the individual level as well as at the organizational level. Because mental models result from a mixture of what was learned explicitly and absorbed implicitly, it is difficult to be aware of them in order to be explicit and share them with others.

In Chapter 2, a distinction was made between memory and mental models. Memory was likened to static storage devices which served as repositories of information and experiences that could be retrievable through varying degrees of effort (Powers, 1973; Miller, Galanter, & Pribram, 1960, Anderson, 1983).³ Argyris & Schön (1978, p. 160) describe organizational memory as

a type of map, a map of the organization's past. Organizational memory, as we have seen, may be contained in individual heads, in files, in documents, or more recently, in computer memories. Organizational memory, therefore, may contain information that is scattered and inaccessible to the agents of organizational learning.

On the other hand, I referred to mental model as "active" structures that affect the way a person thinks and the actions she takes. Mental models are both conceptual frameworks that contain the know-why about the way the world operations, *and* the operational routines for translating that into know-how.

In an organizational context, the important issue is how to represent mental models in a way that 1) enhances the individual's learning and 2) can be shared with others in order to advance organizational learning. Various theories and methods have been formulated to represent mental models. In

³Marvin Minsky (1985), in *Society of Mind*, proposes an interesting model of memory where memories are attached to agents called Knowledge-lines or K-lines. "A K-line is a wirelike structure that attaches itself to whichever mental agents are active when you solve a problem or have a good idea." You remember something by activating the appropriate K-line agent(s) that is attached to the relevant "mental states."

this chapter, we will review some of the literature on mental models research and categorize the various approaches into a typology that maps these approaches by type of representational system and their domain of usefulness (detail vs. dynamic complexity). We begin by looking at examples of mental models in two different domains of knowledge—physical and organizational—and then highlight some of the more important features to include in the operationalization of mental models. Later (in section 3.3.1), we will integrate the various features into a set of criteria for assessing the robustness of mental model representations.

3.1.1 Physical Domain of Knowledge

Gentner & Stevens (1983) characterize mental model research as a careful examination of the way people understand some domain of knowledge. The domain they choose to focus on is physical systems for the following reasons:

Our first efforts to capture naturalistic human knowledge must necessarily center on the simplest possible domains. We need to choose domains for which there exists some normative knowledge that is relatively easy to detail explicitly. Therefore, mental models research focuses on simple physical systems or devices....The reason that mental models research has focused on seemingly technical domains is precisely because those domains have proved to be the most tractable to physical scientists are the one for which there exists the best explicit normative models. (Gentner & Stevens, 1983, p.2).

The body of work contained in their book, *Mental Models*, is based on research done on people's mental models of physical systems such as springs, electricity, heat exchangers, a buzzer system, learning to use calculators, and solving basic physics problems.

3.1.1.1 Physical Representations

The primary representational forms Gentner & Stevens used are schematic diagrams, verbal descriptions and the language of mathematics. For example, in solving physics problems, Larkin (1983) demonstrated how differences in

performance of experts and novices were related to the different problem representations they used. Novices used what she called *naive* problem representations composed of real world objects (blocks and springs) whose actions were governed by developments that occurred over time. Experts, on the other hand, were able to construct *physical* representations in addition to the naive representations, which contained abstract entities such as forces and momenta (see Figure 3.3).

<i>Naive Representations (Envisionments)</i>	<i>Physical Representations</i>
Problem representations with qualitative inferencing rules and imagable entities	
"Familiar" entities	Physical entities
Simulation inferencing (follows time flow)	Constraint inferencing
Distant from physics principles	Closely tied to physics principles
Tree structure, single inference sources	Graph structure, redundant inferences sources
Diffused properties of entities	Localized properties of entities

Comparison of Naive and Physical Representations

(source: Larkin, 1983)

Figure 3.3

Both naive and physical representations use rules of inference (qualitative, not directly tied to equations) to create new information, but they differ in the kinds of entities involved and the actual rules of inference used. Physical representations, for example, do not explicitly include time and are closely tied to physics principles. Thus, the physical representation depicts physical principles as constraint relations that can be examined independent of time flow. The naive representation relies on seeing the actual object, such as a toboggan going down a hill, as an entity whose behavior can be simulated as a sequence of events in time. Physical representations have *localized* attributes which are a defining part of the entity, independent of the context.

Naive representations, on the other hand, do not separate the property of an entity (toboggans go down hills) from the environment (the snow is wet). In short, those who use naive representations have greater difficulty solving physics problems than those who use physical representations because of their inability to *abstract* the essential features of the problem from the details of the problem description.

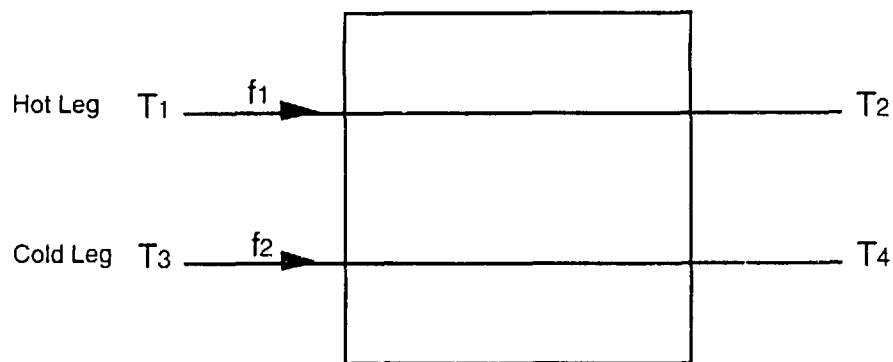
3.1.1.2 Analogies

Gentner & Gentner (1983) studied mental models of electricity as analogies of flowing water or teeming crowds and their effectiveness in capturing certain principles of electricity. By comparing the usage of the two different analogies, they were able to show that the use of the analogy itself affected the thinking: that it was more than just another convenient way of talking about something they already knew through some other thinking process. The subjects who used the water model, for example, were more successful in differentiating between cases where the water analogy had greater explanatory power (batteries and voltages) while those that used the teeming crowd model were more successful in cases involving resistors and current flow.

Analogies can convey overlap in relations among objects without necessarily sharing specific characteristics of the objects themselves. Thus, a water analogy can still convey the concept of a flow of electric current and voltage as a pressure differential without sharing the quality of wetness or the coupling of hydrogen and oxygen atoms. Analogies provide *generic* features that can be applicable in many settings that are very different in terms of the specific details.

3.1.1.3 Autonomous Objects

Williams, Hollan, & Stevens (1983) studied how people invent, modify, and use various mental models in attempting to understand the mechanism of a heat exchanger. They propose several features fundamental to the concept of mental models. One feature is that they are composed of autonomous objects which have an “explicit representation of state, an explicit representation of its topological connection to other objects, and a set of internal parameters.” These autonomous objects have definite boundaries and their behavior (defined as changes in parameter values) is “governed by internal rules reacting to internal parameter changes and to highly constrained external provocation.”



Heat Exchanger Model
(source: Williams, et al., 1983)

Figure 3.4

Mental models are comprised of a collection of connected autonomous objects that are *runnable*. By *runnable*, they mean that the parameter values of an autonomous object can change through qualitative inferences made based on its internal rules and specified topology. For example, a mental model of a heat exchanger would allow a person to “run” the implications of a sudden increase in the temperature of the hot leg liquid on the outgoing temperature of the cold leg liquid. Autonomous objects-based mental models

are seen to aid human reasoning by acting as inference engines for predicting the behavior of physical systems, for producing explanations, and serving as a mnemonic device to help remember things.

3.1.2 Conceptual Domain of Knowledge

Johnson-Laird (1983) distinguishes between physical models (representing the physical world) and conceptual models (representing more abstract matters). We will treat mental models in the organizational domain of knowledge as a subset of the social domain which is a subset of the more general domain of conceptual models.

On a day-to-day basis, people communicate with others through linguistic constructions composed of words and sentences. People usually do not think of such communication as being rooted in mental models that have been built up through prior inputs of other words. When a thought that was conveyed through words (a burning building) is retold to someone else, we are relating a real world object and not just a linguistic construction (article, adjective, noun). The distinction that the mental representation of sentences and texts model aspects of the world and *not* aspects of linguistic structure, is an important one for linguists. In Garnham's (1987, p. 16) words:

Language is a tool that can be used to convey information about the world, but it does not follow that language itself is the system in which situations in the world are represented. Indeed, many people feel uncomfortable with the idea that a sentence *represents* a state of affairs....If language does not represent the world, it is nevertheless true that information conveyed by sentences and texts can be incorporated into mental representations. Furthermore, it is often the case that the knowledge people have of situations has been derived entirely from what they have been told. Linguistic inputs can be the sole external source for setting up a representation of a part of the world.

For those of us who "know" the world through reading texts and listening to others talk of their experiences, our mental representations have been derived primarily through words as input as opposed to direct stimuli of our

other senses. That is, our mental models are created primarily on the basis of linguistic inputs.

Linguistically based mental models

...play a central and unifying role in representing objects, states of affairs, sequences of events, the way the world is, and the social and psychological actions of daily life. They enable individuals to make inferences and predictions, to understand phenomena, to decide what action to take and to control its execution, and above all to experience events by proxy; they allow language to be used to create representations comparable to those deriving from direct acquaintance with the world; and they relate words to the world by way of conception and perception. (Johnson-Laird, 1983, p. 397).

In a general sense, linguistically based mental models affect almost all social interaction since most social activities require verbal communication, oral or written. The domain of linguistic mental models is broad and all-encompassing, which makes it relevant to the study of practically all human endeavors. While we recognize its ubiquitous relevance, we will take a more focused approach in integrating the contributions of linguistics to our concept of mental models. In particular, we will look at two related approaches to representing mental models, scripts and stories.

3.1.2.1 *Scripts*

Schank & Abelson (1977) proposed that part of people's knowledge is represented by hundreds of stereotypic "scripts" which have specific routines associated with each one. A script for dining in a restaurant, for example, may contain routines for making reservations, selecting a mode of transportation, ordering the meal (which may include sub-routines for ordering wine, appetizers and dessert), paying for the meal, etc. Scripts can help people in the planning and execution of conventional activities. They can also enable a person to understand when he or she observes someone else performing another instance of a conventional activity.

If we already have a script of washing clothes at a laundromat, for example, we are quick to understand what the other person is doing when they begin to separate clothes into different piles, go to the change machine and get a handful of quarters, bring over a rolling cart to load the piles of clothes, and search for a machine that is not in use. Without such a script (and no personal knowledge of the routines involved), the actions can appear as disjointed movements that are confusing and incoherent.

When describing familiar activities, people generally agree on the nature of the characters involved, the actions taken, as well as the sequence of the actions (Bower, Black, & Turner, 1979). They also agree on the segmenting of low-level actions into "scenes" which suggests a hierarchical order to their organization in memory. That is, "the script is not a linear chain of actions at one level but rather a hierarchically organized 'tree' of events with several levels of subordinate actions" (Bower, et al., 1979, p. 186). In this tree, answers to *why* questions can be obtained by moving up the tree and answers to *how* questions are to be found by going down the tree.

Analogous to the frameworks and routines in our OADI-IMM model, scripts provide a dual benefit by enabling *understanding* of situations (as schema-based sense-making structures) and by providing an appropriate *guide to behavior* for those situations (Gioia & Poole, 1984). That is, a script provides both the framework or context and the routines or sub-routines providing the know-how for following the script. At the top of the tree is the naming of the script itself which contains the know-why, and at each successive level down there is a routine that embodies the know-how.

3.1.2.2 *Story Model*

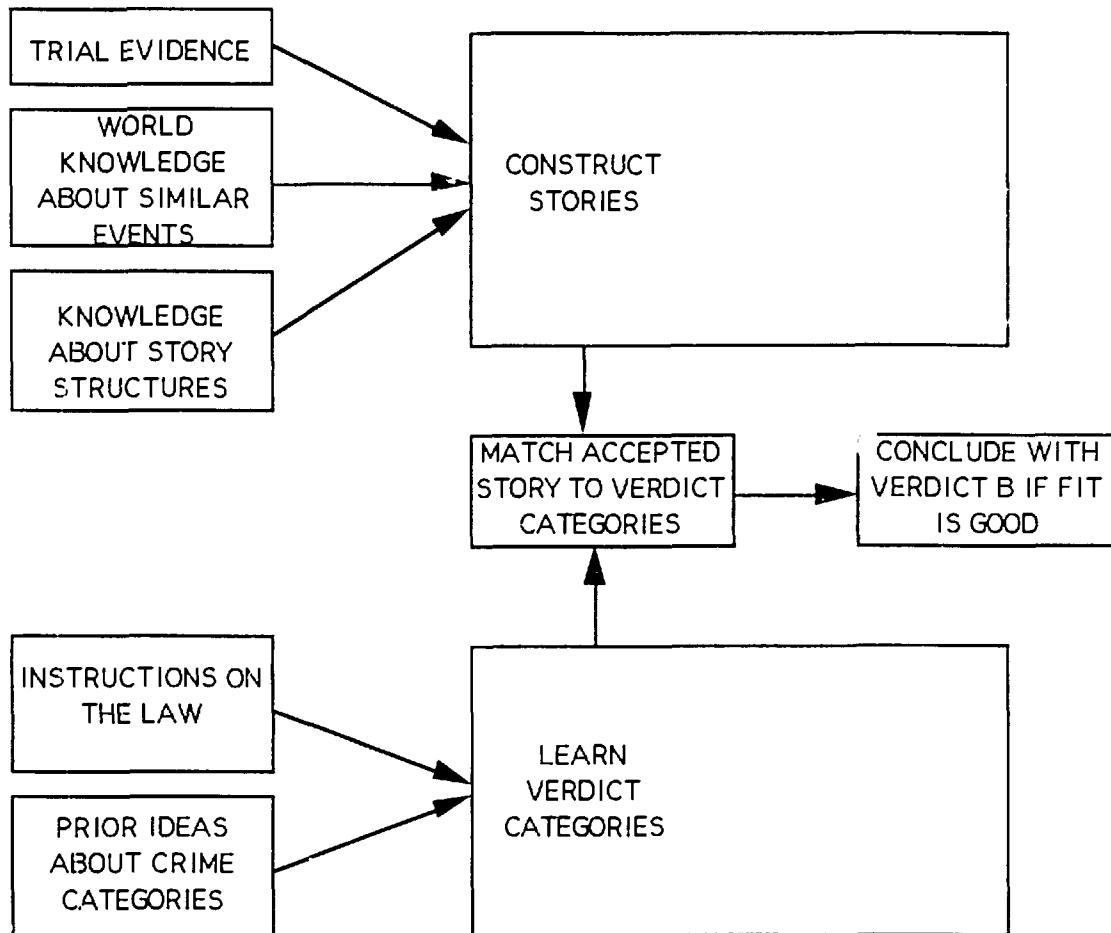
In script theory, a person uses a script to understand situations that are similar to previously encountered ones. Scripts about conventional activities are commonly available through direct experience, observations, reading, conversations, television, etc. In contrast, the story model proposed by Pennington & Hastie (1991) in the context of juror decision-making is based on the hypothesis that jurors impose a narrative story organization on the evidence presented during the trial (see Figure 3.5). Their story model includes the following processes:

...(1) evidence evaluation through story construction, (2) representation of the decision alternatives by learning verdict category attributes, and (3) reaching a decision through the classification of the story into the best fitting verdict category. (Pennington & Hastie, 1991, p. 521).

In addition to the above, the model claims that the jurors base their decisions on the story they have constructed using four certainty principles that determine the choice of story—coverage, coherence, uniqueness, and goodness-of-fit.

In the construction of the story, three types of knowledge will be used according to the theory:

...(1) case-specific information acquired during the trial (e.g., statements made by witnesses about past events relevant to the decision), (2) knowledge about events similar in content to those that are the topic of dispute (e.g., knowledge about a similar crime in the juror's community), and (3) generic expectations about what makes a complete story (e.g., knowledge that human actions are usually motivated by goals). This constructive mental activity results in one or more interpretations of the evidence that have a narrative story form. One of these interpretation (stories) will be accepted by the juror as the best explanation of the evidence. The story that is accepted is the one that provides the greatest coverage of the evidence and is the most coherent, as determined by the particular juror. (Pennington & Hastie, 1991, pp. 521-22).



Sample Story Model for Juror Decision-making

(source: Pennington & Hastie, 1991)

Figure 3.5

Having the greatest coverage means that it accounts for the largest portion of the evidence presented. Coherence has three components: consistency (no internal contradictions), plausibility (corresponds with decision-maker’s knowledge of the world), and completeness (story has all the expected parts). These three components of coherence may be fulfilled to a greater or lesser extent, and the values of all three combine to give the overall sense of coherence.

3.1.2.3 Schemas

Both the script and story models can be considered to be a subset of a larger theory—schema theory. A schema represents a cognitive structure that

contains organized knowledge about specific concepts or types of stimuli, both the attributes of the concepts as well as the relationships among the attributes.

The following is a social schemata in action:

A young woman, casually dressed, walked over to the campus bookstore's requisition desk. 'I'd like to order the books for a course,' she said.

The older woman behind the desk said, 'The books aren't in yet for the fall semester.'

'I know,' the first woman replied. 'I'd like to *order* the books for the course.'

'Oh, certainly. Well, what books does the professor want?' asked the other, helpfully.

'*I am* the professor,' was the frustrated reply.

Assumptions about other people enable us to function. Knowing or guessing that another person is a student, a secretary, or a professor allows us to observe and interpret, to remember and forget, to infer and judge, all in ways that fit our expectations about particular kinds of people. Accumulated general knowledge about categories of people does not do justice to the unique qualities of any given individual, but it makes possible a certain efficiency and adaptiveness in social cognition. (Fiske & Taylor, 1984, p. 139)

The schema concept is seen as a "theory-driven" cognitive process because of the way in which they actively affect the way in which people view data as opposed to a "data-driven" process where the data shapes people's theories. Schemas are theories that help people select, process, and analyze raw data that they perceive. It is based on a fundamental assumption that people actively construct their reality by creating meanings and then adding to it observations from the world. The salesperson and the professor were running up against the salesperson's theory of what a professor looks like. It did not occur to her that a casually dressed woman could be a professor. Those visual cues affected the interaction and eliminated certain possibilities that would not have been eliminated, presumably, if the woman looked more like the salesperson's picture of a professor. This powerful aspect of mental models cannot be underestimated.

3.1.2.4 Influence Diagrams

The mental models representations covered thus far have been represented verbally through the language of words (scripts, stories, and schemas) or mathematically through the language of symbols (physical domain). There are, however, graphical methods for mapping mental models as well. The differences between graphics and verbal language-based representations are shown in Figure 3.6.

Generally speaking, graphics allows one to grasp the whole first, then the parts; with language, it's the reverse. Graphics are also easier to understand without much instruction since comprehensibility is not rule-based. Using purely graphics-based representations, however, can be too limiting without the added flexibility of language. A hybrid system that uses both language and graphics can get the best of both worlds by combining the holistic visual impact of graphics with the concept-rich constructions of language. The first of these *graphics-language* methods to be covered is an influence diagram.

	Graphics	Verbal Language
Manner of recognition:	First, the whole is grasped. Next, elements are analyzed.	First, elements are recognized. Next, whole is constructed.
Ease of understanding:	Understood by almost anybody immediately (pictures).	If rules are not understood, then it is incomprehensible (e.g., foreign languages).

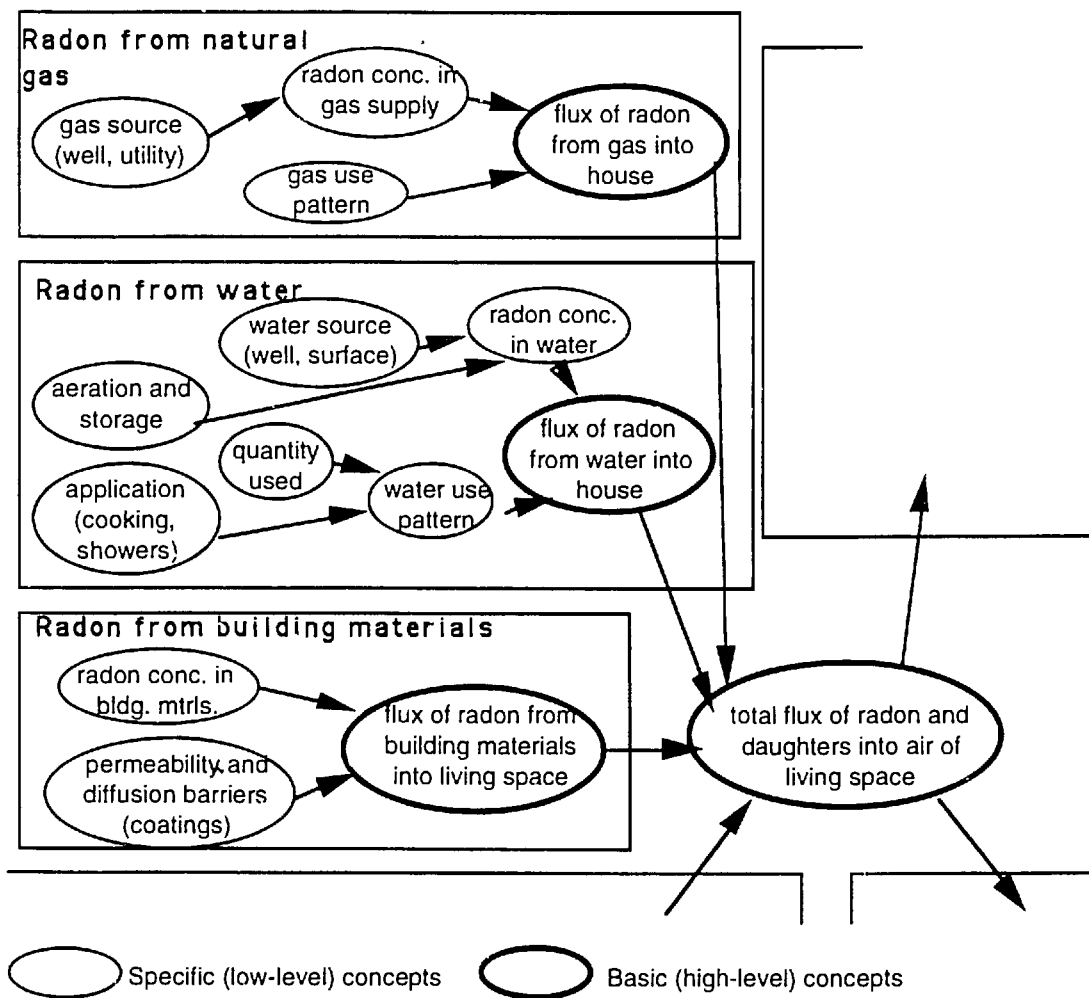
Graphics vs. Language-based Representations

(source: Mizuno, 1988)

Figure 3.6

Bostrom, et al. (1992) laid out a general methodology for characterizing people's mental models using a network representation of expert knowledge.

An *influence* diagram is composed of node-link-node combinations which portray relationships where the value of a concept at the tail of an arrow influences the value of the concept at the arrowhead (see Figure 3.7). In the “Partial Influence Diagram of Radon Risk,” for example, “radon concentration in gas supply” and “gas use pattern” influence “flux of radon from gas into house.” When an influence diagram is completely specified,



Partial Influence Diagram of Radon Risks

(source: Bostrom, et al., 1992)

Figure 3.7

conditional probabilities would be assigned to every link where “*a* influences *b*” if the probability distribution of *b* conditioned on *a* is different from the unconditioned distribution of *b*” (Bostrom, et al., 1992, p. 87).

The basic idea behind their methodology is to assess the completeness and accuracy of people’s mental models about a given issue, such as radon risks, by mapping out their understanding into an influence diagram and comparing it to an expert influence diagram. They outlined the following steps:

...(a) create an expert influence diagram, (b) elicit lay people’s relevant beliefs, (c) map those beliefs into the diagram, and (d) identify gaps and misconceptions. Once risk communications have been composed to address these lacunae, their impact should be evaluated empirically by repeating Steps *b* through *d*. (Bostrom, et al., 1992, p. 89)

In their analysis, they measured *completeness* as a percentage of concepts contained in the expert model that was also in a lay person’s model. They also measured *concurrence* which they defined as a percentage of the lay person’s concepts that also appeared in the expert model. *Accuracy* was then computed as a product of completeness and concurrence. On the basis of their study results, Bostrom, et al. concluded that “people’s understanding of the radon problem seems not only incomplete but also incoherent, in the sense of containing scattered and inconsistent items” (Bostrom, et al., 1992, p. 98). The influence diagram allowed them to explicitly map out people’s understanding of the interconnections among many variables and identify gaps in understanding.

3.1.2.5 Total Quality Management (TQM) Maps

One reason for the relative widespread use and acceptance of Total Quality Management (TQM) tools and methodology is that the tools are designed for “mass consumption.” The tools also fall into the graphics-language category,

which gives them the visual impact of graphics as well as the conceptual basis of language (see Figure 3.8). They are accessible because of three important features: (1) each tool is presented in bite-sized steps, (2) each step has clear guidelines for ease-of-use, and (3) all the tools have been systematically tested in the field. These seven tools are useful for looking at numerical data and constructing explicit maps of the information.⁴

Of the seven QC tools, the cause-and-effect (CE) diagram is the only one in the network system category in Figure 3.8. The tools in the column-row system and the coordinate system are different ways of plotting various kinds of data. The CE diagram, on the other hand, requires a person to graphically

Graphics-language system	Traditional seven QC tools	Seven new quality management tools
Relational System		<ul style="list-style-type: none"> • KJ Method (or Affinity diagram)
Network System	<ul style="list-style-type: none"> • Cause-and-effect diagram 	<ul style="list-style-type: none"> • Relations diagram method • Systematic diagram method • Arrow diagram method • PDPC (Process Decision Program Chart) method
Column-row system	<ul style="list-style-type: none"> • Checksheet 	<ul style="list-style-type: none"> • Matrix diagram method
Coordinate system	<ul style="list-style-type: none"> • Pareto chart • Histogram • Scatter diagram • Control chart • Graphs 	<ul style="list-style-type: none"> • Matrix data-analysis method

Graphics-Language System Classification of TQM Tools
(adapted from Mizuno, 1988)

Figure 3.8

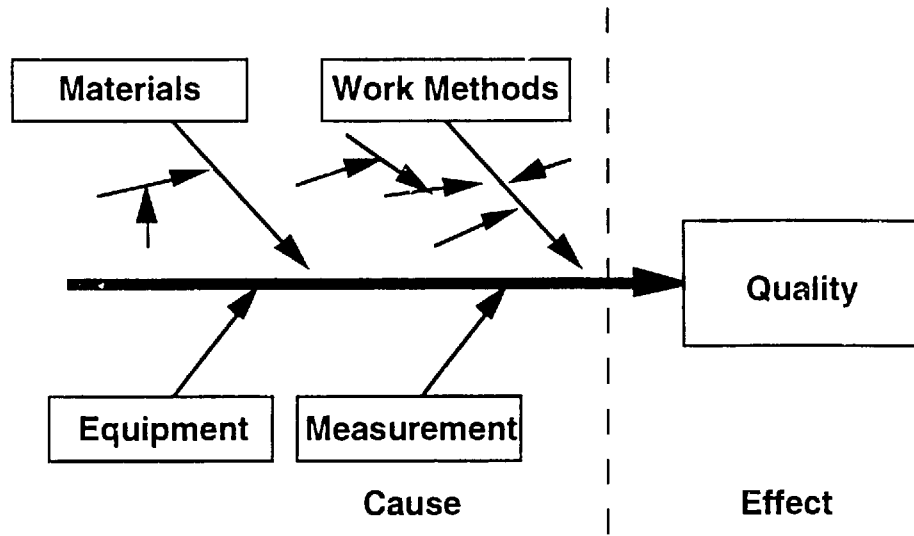
⁴Various guidebooks are available that outline a step-by-step process for using the seven tools of quality control (QC) (Ishikawa, 1982); (Goal/QPC, 1985).

map one's understanding of the causal connections that contributes to the "effect" of interest, like deteriorating quality (see Figure 3.9). Each arrow can have another tier of arrows pointing to it with factors that are causally linked to it. They, in turn, can have arrows pointing to them. The final map looks like tree branches with little twigs sprouting from them. One of the main purposes of the CE diagram is to take quantitative data and sort out the causes of dispersion and organize causal relationships in a way that makes it clear what actions should be taken next.

The seven "new" quality management tools (see Figure 3.8) takes a similar approach as the seven QC tools to looking at qualitative data in a rigorous way. Unlike the seven QC tools, the majority of these tools are in the network system and relational system category which makes them good candidates for representing mental models. The relations diagram helps clarify complex problems or situations by mapping them into intertwined causal relationships and facilitates the finding of appropriate solutions (Mizuno, 1988).⁵ The relations diagram method uses diagrams like the one in Figure 3.10 to solve problems that have a complex cause-and-effect relationship by:

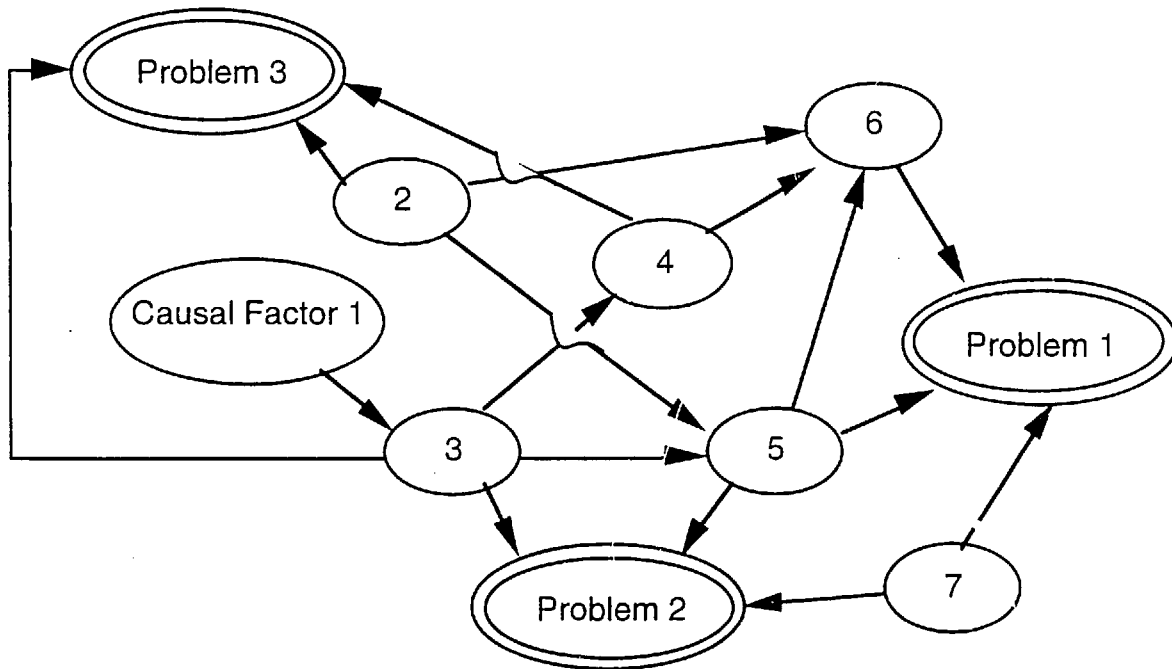
...(1) isolating all factors related to the issue, (2) expressing these factors freely and concisely, (3) identifying logically the cause-and-effect relationships and depicting them using arrows in a relations diagram, (4) producing a complete picture, (5) extracting the key factors. (Mizuno, 1988, p. 88)

⁵A brief description of the other methods in the same category follows. The *systematic* diagram is like a family tree or an organizational chart where one can trace the genealogy of problems or the means to accomplish given objectives. An *arrow* diagram is designed to help establish plans and monitor progress on a project. It can be used in conjunction with and in support of Gantt charts, PERT, and CPM for project management. The *PDPC* or Process Decision Program Chart is used for selecting the best processes to ensure obtaining desired results by reviewing various conceivable outcomes. For a fuller explanation, see Mizuno (1988).



Cause-Effect Diagram
(source: Ishikawa, 1982)

Figure 3.9



A Relations Diagram
(source: Mizuno, 1988)

Figure 3.10

The relations diagram is a logical technique that attempts to incorporate factors and items from a broad perspective into a logical cause-and-effect map of relationships. The relations diagram is considered to be the only QC technique that is effective in tackling cases involving complex inter-relationships.

Although the QC tools have not been explicitly linked to the concept of mental models in the TQM literature, several of them are, in effect, maps of mental models. The relations diagram and cause-and-effect diagram described above, for example, are very similar to the influence diagram discussed earlier.

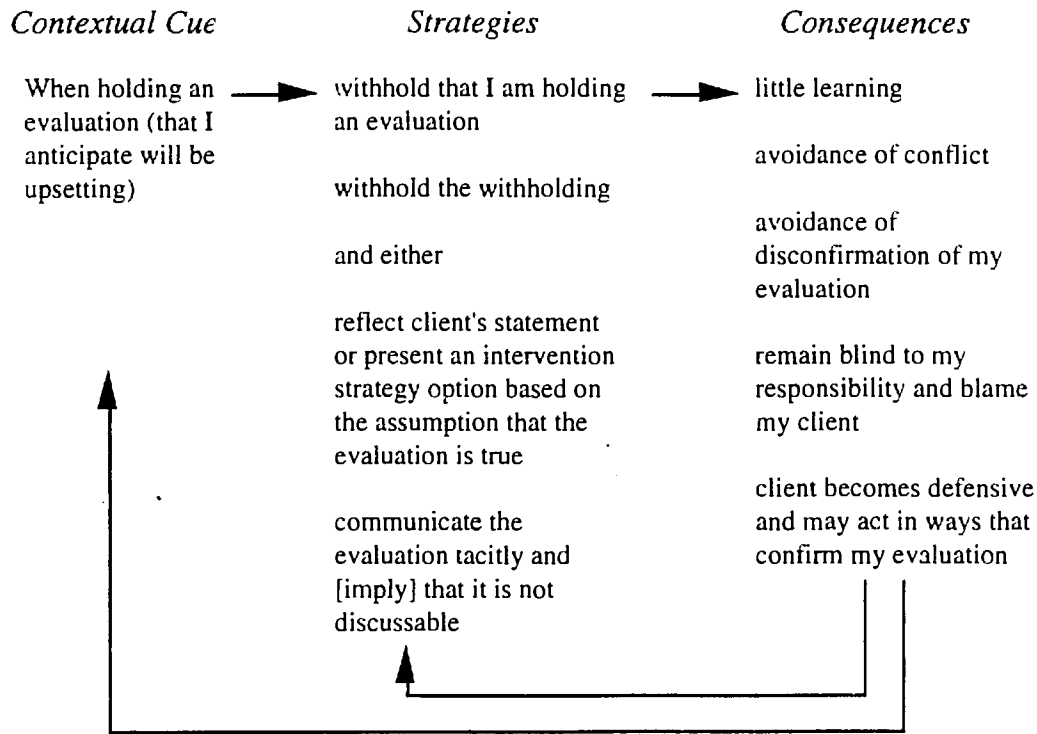
3.1.2.6 *Action Maps*

Argyris & Schön (1978, p. 160) say:

Maps, as we have pointed out, are organized pictures which show how the features of the system have been placed in some sort of pattern which illuminates the interdependence among the parts of the system. By interdependence, we mean the mechanisms by which the parts take from and give to each other the information needed to permit each part to accomplish its organic role and simultaneously help other parts to do the same, thereby creating and maintaining the system.

Action mapping provides this container by the types of information it elicits, as well as the intention of making explicit what is, under most conditions inscrutable. Argyris believes there is much to learn from digging out the buried and hidden “undiscussables” lodged in individual heads and making that information available for consideration. It is the stuff that does not “go without saying,” although individuals may believe that it can be taken for granted.

Argyris maintains that just because people don't talk about their true motives or reasoning behind their actions does not mean that what is left unsaid is not driving their behavior. In Figure 3.11, we see an example of a



Action Map of a Counselor's Practice

(source: Argyris, et al., 1985)

Figure 3.11

counselor's map of a tacit rule that describes "an inter-related set of propositions that told her how to act in the face of negative evaluations and a fear of evoking defensiveness" (Argyris, Putnam, & Smith, 1985, pp. 248-249). This map made explicit a tacit rule that made a counselor who was intent on being client focused, unknowingly act in a very counselor-centered way. Action maps generally include feedback loops which are meant to graphically capture an individual's (or organization's) pattern of behavior that locks itself in a cycle of repeated (and sometimes escalating) actions.

3.1.2.7 System Dynamics

System dynamics is a field of study that provides a methodology for mapping circular relationships and synthesizing disparate types of variables that have traditionally been considered too "fuzzy" to measure. Based on the concept of *feedback* loops, system dynamics provides a methodology for mapping the

ways in which prevailing policies may restrict learning and prevent us from gaining deeper insights into the nature of complex systems (see Appendix A for a list of ten different tools).

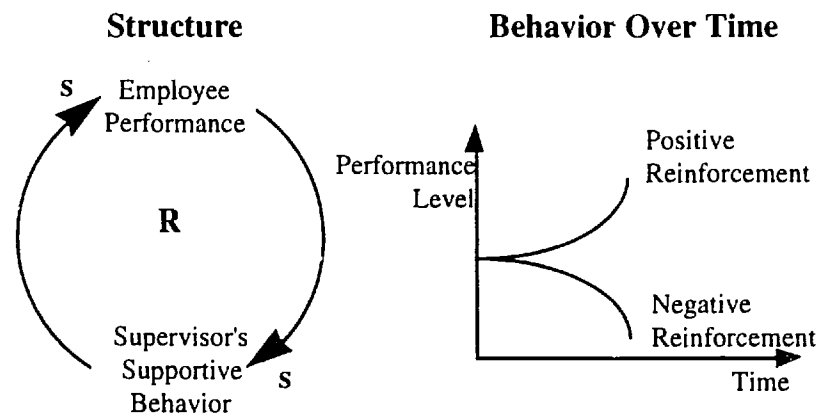
Circular Causality. James D. Watson (1981) describes the process through which he and Francis Crick “cracked” the DNA code. While others were searching for complex structures to explain the diversity of life forms, they explored more simple, geometrical designs and eventually received a Nobel Prize for revealing the double helix structure that is the genetic basis for all life. Through their research, Watson and Crick proved that the infinite variations we see in nature can all be produced by one simple, elegant structure.

Similarly, from a system dynamics perspective, two basic feedback loops—reinforcing and balancing⁶—can be seen as the equivalent building blocks of complex social and economic systems. These simple structures, represented as causal loop diagrams (CLD’s), combine in an infinite variety of ways to produce the complex system behaviors that managers are expected to control (for guidelines on constructing CLD’s, see Appendix B). We can think of CLD’s as sentences which are constructed by linking together key variables with directed arcs and indicating the causal relationships between them. By stringing together several loops, we can create a coherent map of the structure underlying a particular problem or issue. A directed arc or link connects two

⁶The traditional system dynamics notation for signing the links are the plus sign to indicate *additive* change (and positive feedback loop to indicate a process that goes through additive cycles) and the minus sign to indicate a *negating* change (and negative feedback loops to indicate a process that ends up negating change at the end of a cycle). I have chosen to adopt the notation developed by Innovation Associates (Framingham, MA) that is free from the evaluative connotations of the words positive and negative. Instead of a plus we used an “s” to represent a change in the *same* direction. Instead of a minus, we used an “o” to denote a change in the *opposite* direction. Positive loops are referred to as *reinforcing* loops and negative loops are referred to as *balancing* loops. I recommend adopting such a notation for fieldwork because it communicates the message with less ambiguity.

variables together with the arrowhead pointing to the “effect” from the “cause” variable. A directional sign is indicated by an “s” meaning the two variables move in the same direction, or an “o” meaning the two variables move in opposite directions.

Reinforcing Loops: Engines of Growth and Collapse. Reinforcing loops produce both growth and collapse. That is, they compound change in one direction with even more change. For example, in the employee-supervisor reinforcing loop (see Figure 3.12), positive reinforcement from the supervisor is capable of producing good employee performance, i.e., as Employee Performance increases, Supervisor’s Supportive Behavior increases (same direction), leading to even higher “Employee Performance (same direction). This is often referred to as the “Pygmalion effect.” Negative reinforcement, on the other hand, can produce the exact opposite effect. If “Employee Performance” had decreased for some reason, then the “Supervisors’ Supportive Behavior” also decreases (same direction), which leads to further erosion of “Employee Performance” (same direction).



Reinforcing Loop

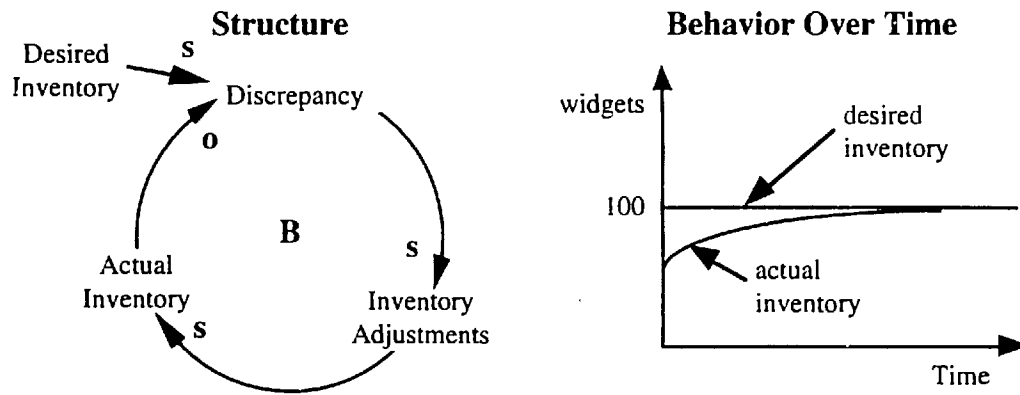
(source: *The Systems Thinker*TM, Vol. 1, No. 1)

Figure 3.12

Balancing Loops: Goal-Seeking Processes. Balancing loops try to bring things to a desired state and keep them there, much like a thermostat regulates the temperature in a house. An example in manufacturing involves maintaining buffer inventory levels between production stages (see Figure 3.13). In this situation, there is a desired inventory level which is maintained by adjusting the actual inventory whenever there is too much or too little. When the "Discrepancy" increases, "Inventory Adjustments" also increases to correct the discrepancy (same direction). This leads to a rise in the "Actual Inventory" (same direction), which reduces the "Discrepancy" (opposite direction). Thus, when we come full circle, the actions tend to bring the system into "balance."

From this feedback perspective, all complex dynamic behavior is produced by some combination of these two basic loops: reinforcing and balancing. The importance of this feedback loop concept is underscored by Weick (1991, p. 86).

Most managers get into trouble because they forget to think in circles. I mean that literally. Managerial problems persist because managers continue to believe that there are such things as unilateral causation, independent, and dependent variables, origins and terminations. Examples are everywhere: leadership style affects productivity, parents socialize children, stimuli affect responses, ends affect means, desires affect actions. Those assertions are wrong because each of them demonstrably also operates in the opposite direction: productivity affects leadership style (Lowin and Craig 1968), children socialize parents (Osofsky 1971), responses affect stimuli (Gombrich 1960), means affect ends (Hirschman and Lindblom 1962), actions affect desires (Bem 1967). In every one of these examples causation is circular, not linear. And the same thing holds true for most organizational events.



Balancing Loop

(source: *The Systems Thinker*TM, Vol. 1, No. 1)

Figure 3.13

The circular interconnected ways in which organizations operate is what makes them such complex systems. That is why, from a system dynamics perspective, if a system is decomposed into its components and each component is optimized, the system as a whole can be guaranteed *not* to be optimal. A common characteristic of many complex systems, however, is that they are often designed with the intention of optimizing the parts rather than the whole. Causal loop diagrams are used to help map the interconnections among all relevant parts of a system, to capture a more integrated view of the system.

System dynamics can help managers gain a more systemic view of their organization by focusing on making their mental models of critical issues explicit, exposing them to challenge, and altering them based on insights gained from this process. Although this can be said to varying degrees of all the different ways in which mental models may be represented, there are domains in which the system dynamics methods excel over the others. This point will be clarified in section 3.2.2 on Domain of Usefulness.

3.2 A TYPOLOGY OF MENTAL MODEL REPRESENTATIONS

Making mental models explicit requires a language or tool with which to capture and communicate them. Different methods are useful in some domains and less so in others. In this section, we will pull together various ideas gathered from the literature reviewed above and create a typology to better understand the differing strengths and domains of applicability of the available methods.

3.2.1 Criteria for Representing Mental Models Based on the Literature

Based on the literature reviewed above, we can generalize a set of criteria for evaluating the usefulness of mental model representations. From the research on physical systems reviewed above, three general features of mental model work are particularly relevant to our model of organizational learning. First, a mental model representation should be sufficiently *abstract* to allow a person to think in terms of principles that are “disembodied” from the physical form (Larkin, 1983). Second, it should be sufficiently *generic* to hold true in various settings even though the specific characteristics of the representations may change (Gentner & Gentner, 1983). Third, it should be *runnable*—that is, represented in such a form as to allow mental simulation for inferring what the future state of a system will be (deKleer & Brown, 1983; Norman, 1983; Williams, et al., 1983). That is, not only do mental models provide us with ways to perceive the world around us in the present moment, they also project us into future considerations and expectations.

Although the above discussion is presented in terms of concepts, not actions, the criteria apply to both mental model *frameworks* and to mental model *routines*. Being abstract means that the know-how is proceduralized into a “macro” routine (e.g., processing payroll checks) and not at a “micro”

level (e.g., getting the checks out of the drawer, putting pen to paper, etc.) Routines can be generic such that a basic routine is applicable to different circumstances (e.g., processing accounts payables or purchase orders). Routines, of course, must also be runnable in order for them to be useful.

In the survey of the literature in the conceptual domain there are also several features worth noting. The concept of scripts reinforces the dual role of mental models in conceptual understanding and developing routines that guide behavior. In schema theory we see mental models as *theory-driven* cognitive processes where a particular schemata affects the way we view data versus a *data-driven* process where the data themselves affect the theory (Fiske & Taylor, 1984). In reality, both processes are always at work, although schema theory puts more emphasis on the former. The TQM literature highlighted the difference between verbal language and *graphics-language* representation systems and showed how graphics-language systems are better suited for explicating complex cause and effect relationships and communicating them to others. Both action maps and causal loop diagrams explicitly treat *feedback* relationships, with causal loop diagrams presenting a more rigorous method for treating multiple feedback loops.

In addition to the many attributes summarized above, there were also some methodologies for *validity testing* of the mental model representations. The story model, for example, included a process for checking the overall coherence of the story which included consistency, plausibility, and completeness. In the use of influence diagrams, there was a method for testing the accuracy of the diagrams created by lay people by assessing completeness and concurrence.

Mental model representation criteria summary:

- Provides a level of abstract representation

- Allows transferability through generic representation
- Enables mental simulation (“runnable”)
- Are theory-driven
- Are testable

Each of the representational systems reviewed embodies a particular perspective on the world and utilizes a specific form for capturing its view of the world. Clearly, no single representational scheme can be equally well-suited to model all domains of relevance. The combination of representational form and the worldview employed determine what domains each mental model representational system is capable of addressing. In the next section, we will draw a distinction between two different domains of usefulness and show why a system dynamics approach is well-suited for addressing domains where dynamic complexity is high.

3.2.2 Domains of Usefulness: Dynamic vs. Detail Complexity

Most of the mental model representation methods discussed above under the organizational domain of knowledge are suitable for mapping static relationships that deal with large amounts of detail. For example, the story model (presented in section 3.1.2.2) provided a way to make sense out of a very detail-rich case by providing a coherent storyline. The use of scripts, likewise, allows one to remember large amounts of information that are linked to scripts which act as triggers (e.g., when a person triggers a restaurant script, that in turn triggers sub-routines that are applicable in that situation). With the exception of systems maps, the representational methods are appropriate for dealing with situations that are rich in detail complexity, not dynamic complexity. And yet, arguably, the areas of significant importance in which organizations most need to learn are specifically those areas which

have complex short-term vs. long-term trade-offs, complex inter-relationships, and non-obvious causal interdependencies, i.e., while a network of proximate causal relationships might be identifiable by participants in the system, the critical inter-relationships for some other pattern of behavior are not obvious .

In this next section, we will clarify what we mean by dynamic complexity. We will then propose why system dynamics provides a methodology and set of tools that are particularly well-suited for mapping dynamic complexity. Finally, we will present a typology that summarizes the various representational maps that have been presented in this chapter. The typology distinguishes between the different representational systems by language type and by the domain of usefulness.

3.2.2.1 *Characteristics of Dynamic Complexity*

Ackoff defines a mess as follows:

I am going to call this thing a mess. Then we say that what reality consists of are messes, not problems.

Now what is a problem? Let's take a mess for a moment, which is what you're confronted with in the morning when you come to work, and let's analyze it. Remember what analysis is—to take something apart. So if we take a mess and start to break it up into its components, what do we find that those parts are? The parts are problems. Therefore, a problem is an abstraction obtained by analyzing a mess.

Then what is a mess? That's the significant thing—a mess is a system of problems. Now, the significance of this is that the traditional way of managing is to take a mess and break it up into problems and solve each problem separately, with the assumption that the mess is solved if we solve each part of it.

But remember...if you break a system into parts and make every part behave as effectively as possible, the whole will not behave as effectively as possible. Therefore, the solution to a mess does not consist of the sum of the solutions to the problems that make it up. And that is absolutely fundamental.⁷(Ackoff, p. 13)

⁷This is taken from an essay by Russell Ackoff entitled "The Second Industrial Revolution."

In short, Ackoff's term "organizational messes" describes situations that are high in dynamic complexity. Figure 3.14 is an attempt to classify "organizational messes" into a grid based on Organizational Complexity (the number of organizational units and level of complexity of their interconnections) and Time Lags of Process (inherent time delays of the activities involved). "Fuzzy" variables often accompany Ackoff's "messes" because such systems are rarely clear-cut or well-defined by traditional measures. Organizational settings that are high in dynamic complexity have three characteristics: a high degree of organizational complexity, long time delays, and fuzzy variables.

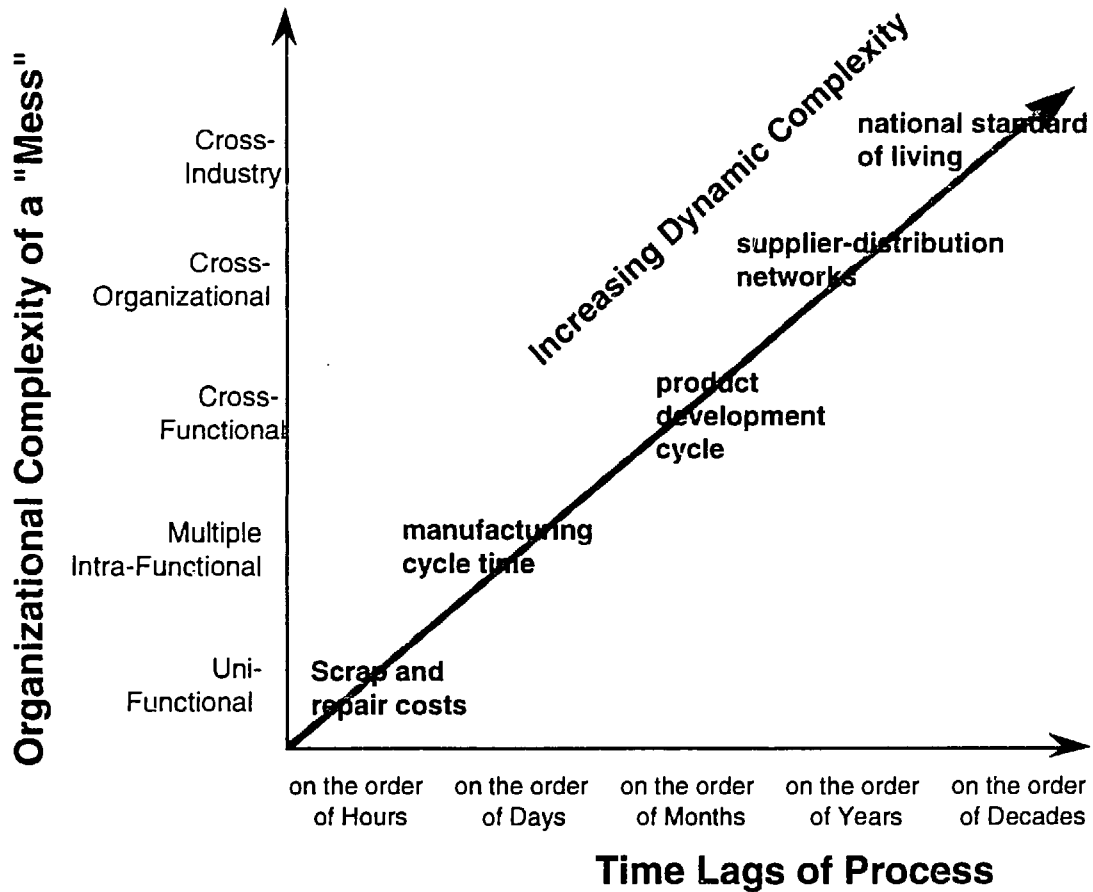
3.2.2.2 *Organizational Complexity*

Schneiderman (1988) showed that the rate of improvement in a wide range of TQM projects is primarily a function of the organizational complexity of the project, not the specifics of the project itself. To appreciate how organizational complexity can present great difficulties in communicating, consider

an organization with six levels below the senior executive and a span of control of three. This makes 1,093 people. More importantly, there are 586,778 potential two-person interfaces that represent potential internal supplier-customer relationships. These 1,093 people depend on one other to get their job done, but their interdependencies are not explicit (Baker, 1989, p. 13).

Managing even a few of those supplier-customer relationships can be a daunting task. Actions taken without an understanding of the interdependencies, however, can produce undesirable results. For example, managers at Xerox who bought into the concept of Just-in-Time (JIT) manufacturing "solved" their inventory problems by demanding their suppliers to hold the inventory until they were ready to accept them. By "injecting" a good idea into one part of the system without a full appreciation

of the whole, they severely strained the good supplier relations painstakingly developed over many years (Hutchins, 1986).



Dynamic Complexity of "Messes"
 (source: Kim 1990)

Figure 3.14

3.2.2.3 Time Delays

When time delays are relatively short, understanding the dynamics of the system is usually pretty straightforward. For example, reducing defects at a specific production step means getting real-time data and analyzing it. The process step usually takes minutes or hours, not days or months. Thus, it is feasible to collect data and be confident about causal conclusions drawn from the data. When the time delays of a project are extremely long, such as in product development, running real-time experiments becomes impractical

and current data is of limited usefulness. One can tweak individual steps within the process but cannot gain much insight about their implications on the process as a whole.

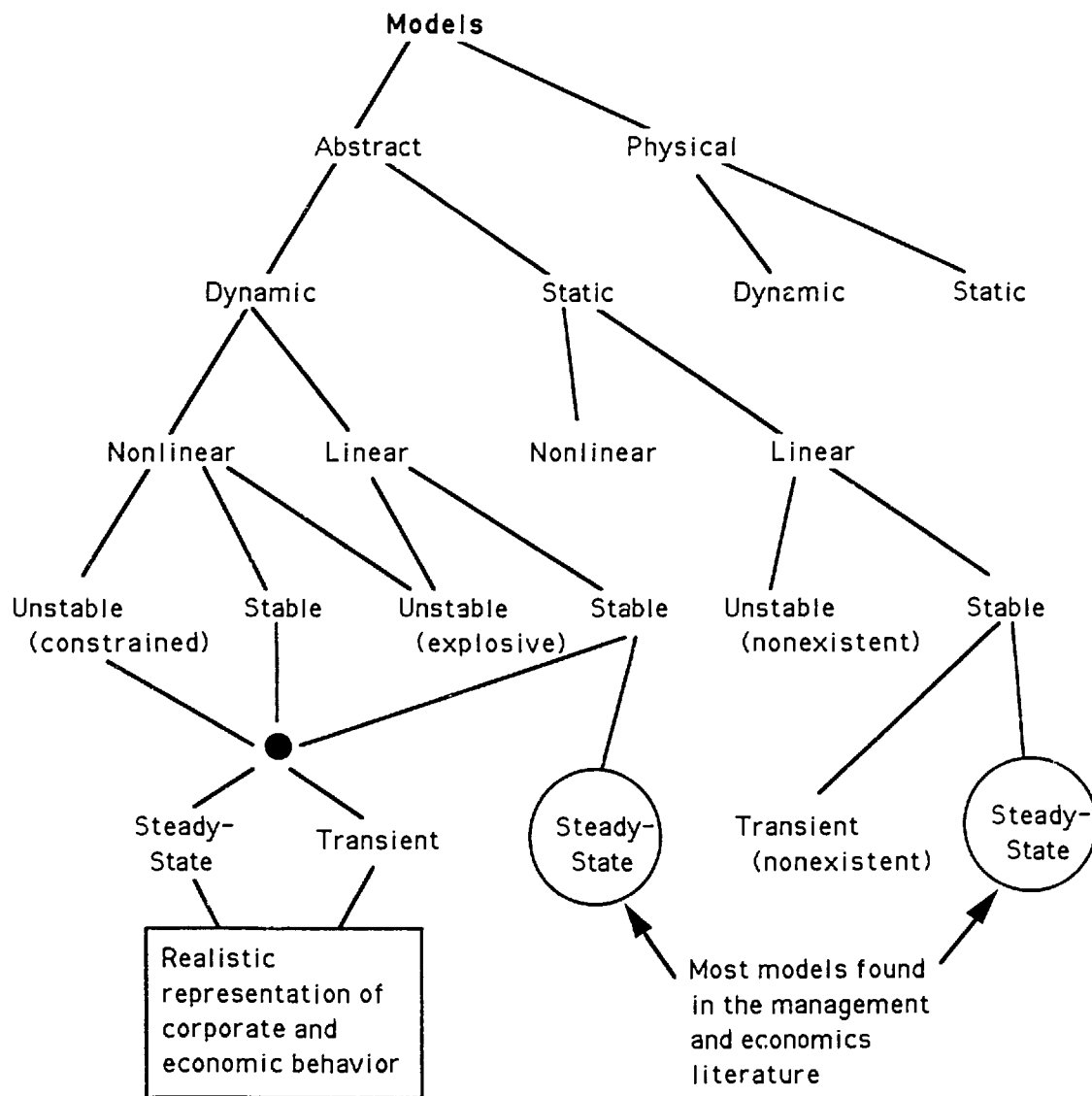
3.2.2.4 *Fuzzy variables*

The three laws of TQM are “look at the data, look at the data, and look at the data.” Getting good, reliable data is not an easy thing to do, even on mechanical processes which have universally accepted units of measure such as units per minute, or pounds per unit. The task of collecting data becomes much more difficult as the measured item becomes less and less tangible. Fuzzy variables include such notions as time pressure, morale, productivity, creativity, and other information that may be available only at the intuitive level.

3.2.3 **Mapping Dynamic Complexity**

Mapping dynamic complexity requires a set of tools that is designed to map complex interrelationships and time delays in a system. System dynamics provides a set of tools and a methodology that are particularly well-suited to addressing the three characteristics of dynamically complex systems defined above. Fundamentally, system dynamics is about understanding the interconnected feedback structures and dynamic behavior of complex systems. System dynamics embraces a particular view of reality that is powerful for understanding dynamic phenomena and encompasses many tools that are well-suited for tackling dynamic complexity. Prominent among them is the use of computer models, which enable the compression of time in order to simulate the long-term consequences of policies, and provide a format for quantifying the effects of fuzzy variables, such as morale, on performance figures, such as weekly sales.

Forrester (1961) presented a classification of models that provides an overview of the different categories of models that are of interest in differentiating between detail and dynamic complexity (see Figure 3.15). In the diagram, he identifies the kinds of models he feels are likely to realistically represent corporate and economic behavior—abstract, dynamic models that produce both transient and steady-state behaviors. I see those to



Classification of Models
(source: Forrester, 1961)

Figure 3.15

be in the domain of dynamic complexity. The other class of models which he identifies as static, linear, and stable or dynamic, linear and stable are in the domain of detail complexity. System dynamics models are generally built to address dynamic complexity.

System dynamics computer models are created through a rigorous thinking process which requires at least five different types of thinking skills—dynamic, generic, structural, operational, and scientific (Richmond (1990). *Dynamic* thinking is the ability to see and to deduce behavior patterns rather than focusing on and seeking to predict events. *Generic* thinking leads us away from thinking in terms of specifics and seeing commonalties across diverse settings. *Structural* thinking, one of the most disciplined of the systems thinking tracks, requires that people think in terms of units-of-measure, or dimensions, and that they rigorously adhere to physical conservation laws, as in the distinction between a stock and a flow in a structural diagram. The fourth skill, *operational* thinking, means thinking in terms of how things really work—not how they theoretically work, and it grounds students in reality. Lastly, *scientific* thinking has to do with quantification (as opposed to precise measurement) and rigorous testing of hypotheses.

Each of these thinking skills is required in system dynamics. Kim (1990) proposes a set of ten systems thinking tools that explicitly maps the system dynamics tools that are currently in use into several of the critical thinking skills identified by Richmond (see Appendix A).

3.2.4 A Typology

The typology contained in Figure 3.16 maps the various representational systems described above by representational system and domain of usefulness

(detail vs. dynamic complexity). This particular grid was selected in order to highlight the domain of organizational issues that we will be focusing on in Chapters 5 and 6 (which are high in dynamic complexity) and how the representational system (graphics-language feedback system) used differs from the others that have been presented. The purpose of this typology is to show relative strengths and not to categorically typecast each into a single box. Stories, for example, can convey a great deal of dynamic complexity, just as causal loop diagrams can represent a lot of detail complexity. The nature of each representational form, however, makes one stronger in capturing detail complexity and the other in dynamic complexity.

		Detail Complexity	Dynamic Complexity
Representational System	Verbal Language		<ul style="list-style-type: none"> • Analogies • Scripts • Stories • Schemas
	Graphics-language	Network System	<ul style="list-style-type: none"> • CE diagram • Influence diagram • Relations diagram
		Feedback System	<ul style="list-style-type: none"> • Action maps
	Mathematical		<ul style="list-style-type: none"> • Linear Programs

A Typology of Mental Model Representational Systems

Figure 3.16

For the purposes of this dissertation, we will focus on the dynamic complexity domain and explore how system dynamics can contribute to the process of building shared mental models.

3.3 SHARED MENTAL MODELS

We have looked at great length what the attributes of mental models are and developed a typology of representational systems in Figure 3.16. Up to now, we have not addressed the issue of individual versus shared mental models. If a representational system works at the individual level, there is no reason for it not to work at the shared level. So, why do we even make the distinction between individual and shared mental models? Couldn't the explication of an individual mental model be considered an automatic contribution to a common organizational pool of available mental model representations? The answer is a qualified "no."

On the face of it, the proposition seems valid. If an individual who is a member of an organization maps out his or her mental model and makes it available to other members, it certainly has the *potential* to be a shared mental model. But the real question is whether it is *actually* shared inside the minds of other members of the organization. This is an issue of diffusion and is dependent, in part, on the *accessibility* of the mental model representation. Shared mental models, therefore, are those that have been internalized by many of the organizational members into a *Weltanschauung* that informs their view of the world and organizational routines that are consistent with that view. Simply having explicit organizational maps developed and available does not guarantee the existence of a shared mental model in the same way that having a company vision statement in print does not guarantee having a shared vision (Senge, 1990b).

3.3.1 Accessibility

Even though system dynamics has had a long tradition of recognizing the importance of mental models in organizational decision-making, the

development of system dynamics tools has focused primarily on preserving the rigor of the mathematical computer modeling discipline. The traditional textbooks have treated the computer model as the ultimate output of the system dynamics method. Both Richardson & Pugh (1981) and Richmond, et al. (1987) focus on developing models in a particular modeling language (DYNAMO and STELLA, respectively); Roberts (1983) presents a more general treatment of simulation models; Randers (1980) covers the principles underlying the system dynamics method.

Of course, it isn't the sophistication of the modeling method or the elegance of the mathematical formulations that is important, but rather the relevance and significance of the issues modeled. As Forrester (1961, p. 116) writes:

An elaborate and accurate model is of little value if it relates to questions and behavior that are of no consequence to the success of the organization. On the other hand, a simple and even inaccurate model may still be tremendously valuable if it yields only a little better understanding of the reasons for major success and failure. Our mental models are examples. The simple and rather inaccurate dynamic models in the minds of skilled managers have been far more effective in carrying our industrial civilization to its present heights than have the formal mathematical models thus far used in management and economic science. The manager's working models in the form of verbal descriptions and thought processes are more attuned to important objectives of the future and more perceptive of the behavior mechanisms of actual organizations than have been the abstract models developed for the explanation of the past.

Throughout the system dynamics literature, causal loop diagrams play a secondary role to computer simulation models in representing mental models (Forrester, 1971; Richardson & Pugh, 1981). Causal loop diagram mapping are presented as rich in problem conceptualization when building the model and in communication of model insights after the modeling project has been completed. It is recognized that the formulation of and testing of the computer model can generate insights about the system and alter people's mental models, but the computer models themselves do not

remain in people's minds. Causal loop diagrams are often used to encapsulate general principles and lessons arising from computer models.

In *The Fifth Discipline*, Senge (1990b) proposes a different role for causal loop diagrams as a managerial tool for preliminary diagnosis and collaborative inquiry. He presents a set of "nature's templates" called systems archetypes that provide a feedback loop perspective on complex dynamics that are summary representations of many models built over the years. We can liken them to "dynamic scripts" in the way they embody particular interrelated dynamics such that when one recognizes a specific archetype, such as Shifting-the-Burden, it will trigger a line of inquiry guided by that dynamic script. A distinct advantage of these archetypes are that they don't require a steep learning curve. The archetypes provide a way for lay managers to map and enrich their mental models of complex systems by making explicit the feedback relationships and clarifying their understanding of them. We will cover systems archetypes in greater detail in the following section.

The archetypes meet the five basic criteria for mental model representations that we developed from the literature review. Each archetype has a storyline, which provides a level of *abstract* representation and aids a person in recognizing and remembering its general features. They are *generic* representations of dynamic phenomena that recur in diverse settings (Senge, 1990a; Kim & Burchill, 1992), making it generically applicable and thus easy to identify occurrences. The causal loops embody a set of dynamic relationships that one can trace through and *run* mental simulations of the possible dynamic outcomes of a situation. Each archetype provides a *theory-driven* set of prescriptions that can guide actions and data gathering to *test* the causal relationships hypothesized in the archetype. In summary, the archetypes are

like *dynamic scripts* that provide a way of making sense of new learning as well as organizing what has already been learned (as mental models).

3.3.2 Systems Archetype Descriptions

In the following descriptions of five systems archetypes, we will describe the basic storyline of each archetype, present the structure of the feedback loops which comprise it, and offer several examples which illustrate how the archetype is used to create an abstract, visual representation of a particular issue.

3.3.2.1 Limits-to-Success Archetype

“It was the best of times, it was the worst of times, it was the age of wisdom, it was the age of foolishness...” wrote Charles Dickens in *A Tale of Two Cities*. The Limits-to-Success archetype describes a similar paradox that many companies face. A rapidly growing company finds itself too busy to invest its profits in internal development, but when sales begin to slow, it no longer has the resources (money and people) to spend on needed improvements. The “best of times” for investing in resource development always seems like the “worst of times” for actually carrying out such plans, and vice versa.

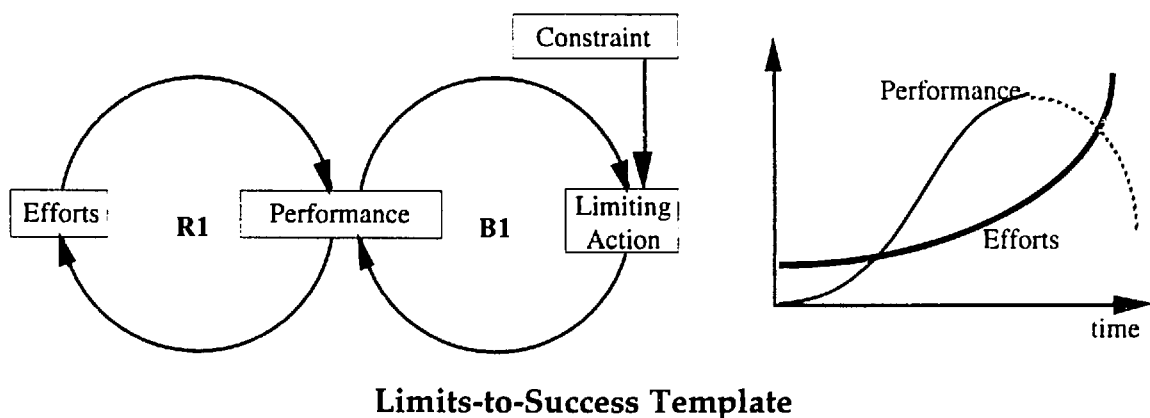


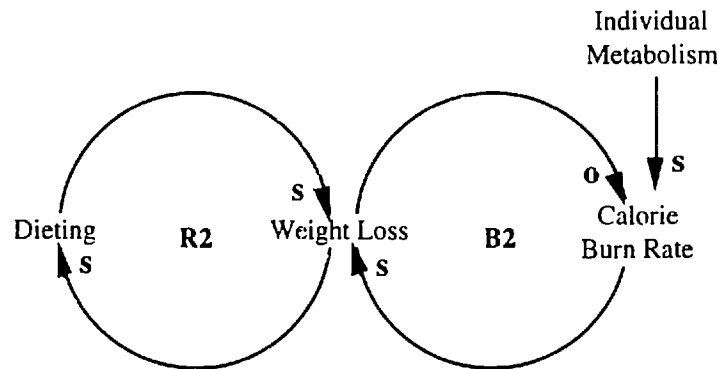
Figure 3.17

In a typical Limits-to-Success scenario (see Figure 3.17), a system's performance continually improves as a direct result of certain efforts. As performance increases, the efforts are redoubled, leading to even further improvement (loop R1). When the performance begins to plateau, the natural reaction is to increase the same efforts that led to past gains. But the harder one pushes, the harder the system seems to push back: it has reached some limit or resistance which is preventing further improvements in the system (loop B1). The real leverage in a Limits-to-Success scenario doesn't lie in pushing on the "engines of growth," but in finding and eliminating the factor(s) limiting success.

In a rapidly-growing company, for example, initial sales are spurred by a successful marketing program. As sales continue to grow, the company redoubles its marketing efforts and sales rise even further. But after a point, pushing harder on marketing has less and less impact on sales—the company has hit some limit, such as market saturation or production capacity. To continue its upward path, the company may need to invest in new production capacity or explore new markets.

3.3.2.2 *Diets and Weight Loss*

The business and popular press provides many examples of situations in which rapid success is followed by a slowdown or decline in results. For example, a person on a diet usually finds that losing the first ten pounds is easier than losing the last five, and, in general, the first diet a person undertakes is usually more successful than any subsequent weight-loss program.



The Dieting Bind

Figure 3.18

On a diet, ingesting fewer calories leads to weight loss, which encourages the person to continue to cut back on food intake (loop R2 in Figure 3.18). But, over time, the body adjusts to the lower intake of food by lowering the rate at which it burns the calories. Eventually the weight loss slows or even stops. The limiting process is the body's metabolic rate. To continue losing weight, the person needs to take actions that will increase metabolism.

One method for increasing the metabolic rate is to perform exercise. However, exercise alone will not create the desired weight loss, because intense exercise burns simple sugars and not the stored fat that is the real target for weight loss. Intense exercise is counterproductive towards the dieter's goal because it *increases* appetite (leading to a higher food intake) while only temporarily raising the metabolic rate. A higher-leverage action is to engage in steady exercise such as long, brisk walks that will gradually increase the metabolic rate to a permanently higher level.

3.3.2.3 *Service Capacity Limit*

People Express airlines is one of the best-known casualties of the Limits-to-Success archetype (Sterman, 1988). Its tremendous growth was fueled by a rapid expansion of fleet and routes along with unheard-of low airfares. As

the fleet capacity grew, People Express was able to carry more passengers and boost revenues, allowing it to expand fleet capacity even more (loop R3 in Figure 3.19). The quality of service was initially very good, so the positive experience of many fliers increased word-of-mouth advertising and the number of passengers.

The growth process at People Express was seen as physical capacity—expanding fleet size, employees and routes. The limiting factor, however, was service capacity—the ability to invest time and money in training employees—which became more difficult to sustain as the company grew (loop R4).

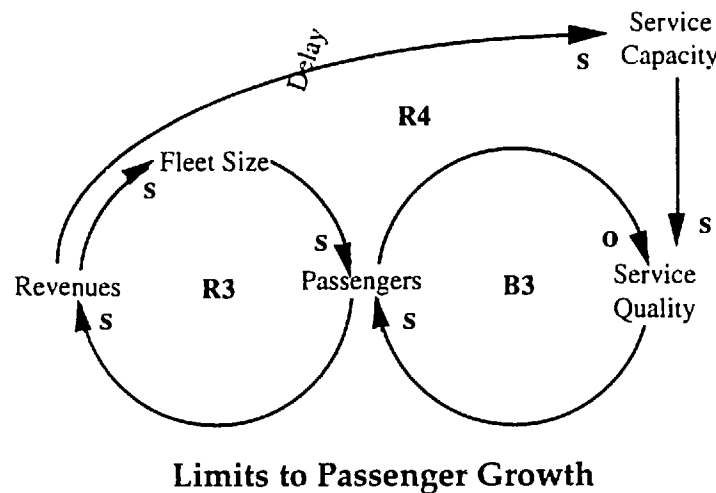


Figure 3.19

The number of passengers flown eventually outstripped the airline's capacity to provide good service. As a result, quality suffered and it began to lose passengers (loop B3). When competitors began matching low rates on selected routes, People Express' market competitiveness suffered further. Focusing only on the reinforcing side of the structure turned People Express' initial rapid growth into accelerating collapse, which contributed to the airline's demise.

Simply hiring more employees was not the solution to People Express' service capacity problems. Similar to the dieter's reliance on intense exercise, it masked the real need for a steady, long-term commitment to hire and train the necessary people to bring service quality up to a high and sustainable level.

3.3.2.4 *Using the Archetype*

The Limits-to-Success archetype is most effective when it is used in advance of a systemic collapse to see how the *cumulative* effects of continued success might lead to future problems. Using the template to trace through the dynamic implications of a current or proposed strategy can highlight potential problems by revealing pressures that are building up in the organization as a result of the growth.

The lesson of the Limits-to-Success archetype is that the systemic leverage lies in removing the limit (or weakening its effects), not on continuing to drive the reinforcing growth process. An awareness of the causal links between the growth processes and the limiting factors can provide insight into possible ways to manage the balance between the two.

3.3.3 **Shifting-the-Burden**

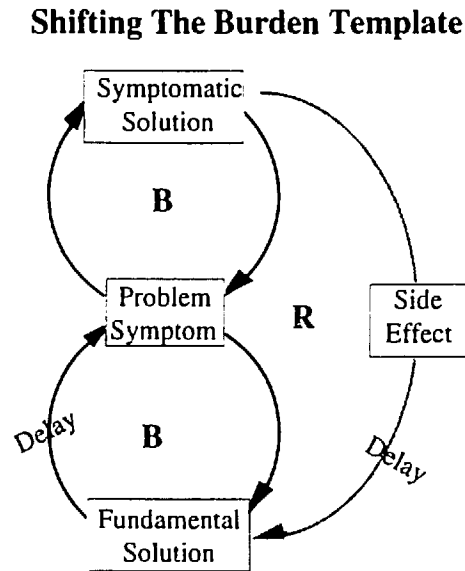
The story of Helen Keller, who, though stricken blind, deaf, and mute from a childhood illness, graduated from Radcliffe College and became an international lecturer and author, is much more than an inspirational human interest story. It illustrates a pervasive dynamic that is rooted in an archetypal structure, Shifting-the-Burden.

Helen's parents, believing that their young child was helpless, assumed all caretaking responsibilities for her. Their actions, though well-intentioned, shifted the burden of responsibility for Helen's welfare to them. Every

problem or failure on Helen's part brought the parents rushing to her aid. As a result, Helen failed to learn even the simplest survival skills. Each incident further reinforced her parents' belief that she was indeed helpless. All three were caught in a system that was eroding Helen's ability (and desire) to cope with the world while shifting the responsibility for her well-being to her parents. Only through the intervention of her teacher, Ann Sullivan, was Helen able to break out of the dynamic and begin developing her own capabilities.

3.3.3.1 The Structure

The basic structure of this archetype is shown in Figure 3.20. The archetype usually begins with a problem symptom that prompts someone to intervene and "solve" it. The solution (or solutions) that are obvious and most able to



In the "Shifting the Burden Template", a problem symptom is "solved" by applying a symptomatic solution which diverts attention away from the fundamental solution.

Shifting-the-Burden Template

Figure 3.20

be immediately implemented usually relieve the problem symptom quickly. However, these symptomatic solutions have two specific negative effects: they divert attention away from the real or fundamental source of the problem; and the symptomatic solution causes the viability of the fundamental solution to deteriorate over time, reinforcing the perceived need for more of the symptomatic solution.

In the Helen Keller story, her parents' intervention is the symptomatic solution, Helen's failure to cope with real world is the problem symptom, the development of Helen's own abilities to care for herself is the fundamental solution, and the side-effect is that her parents assume increasing responsibility for her well-being. This particular type of Shifting-the-Burden structure, in which responsibility is shifted to a third party, is known as Shifting-the-Burden-to-the-Intervener. Over time, the role of the intervener increases until it becomes an essential part of the system. In Helen's case, her parents' actions reinforced the underdevelopment of her abilities and therefore strengthened their role as "protectors."

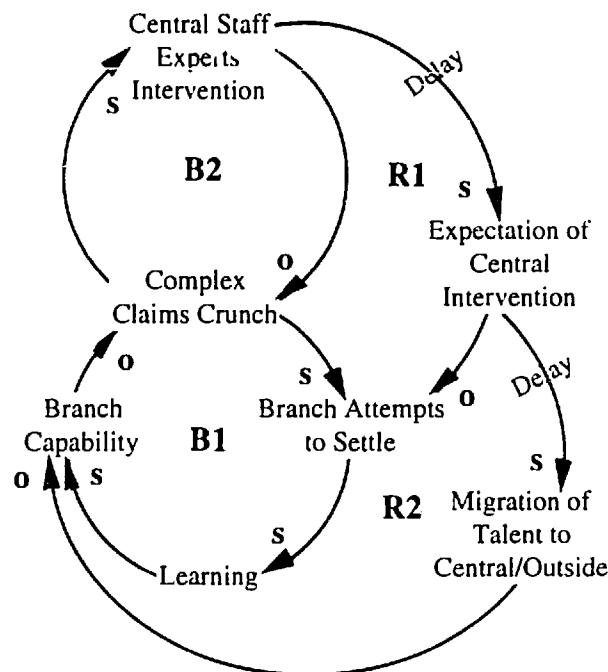
Another common side effect that occurs in Shifting-the-Burden situations is that the person may become "addicted" to the symptomatic solution. For example, a person who turns to alcohol or drugs to boost his self-esteem or help deal with stress may end up developing an alcohol or drug addiction.

3.3.3.2 *Central vs. Local*

The Shifting-the-Burden archetype and its variants—Addiction and Shifting-the-Burden-to-the-Intervener—comprise perhaps the single most pervasive systems structure. Figure 3.21 illustrates a classic example of this dynamic.

A claims office in a local branch of a large insurance company is faced with a large, complex claim that requires more expertise than it possesses. The central office responds by sending out its corps of experts who take care of the complex claim while the branch office goes about its other, more routine business. Although the occurrence of large claims may be infrequent—making it hard to justify keeping such experts in every branch—over time the interventions can result in deteriorating branch capability.

Central Support vs. Branch Capability



In this example of a "Shifting the Burden" archetype, the symptomatic problem is a complex claim that the branch cannot handle alone. Experts from the central office help out, but over time the branch's ability to handle difficult claims atrophies.

Central Support vs. Branch Capability

Figure 3.21

The reason is that after a while, an implicit operating norm develops that says if a person wants to handle complex, technically challenging claims she has to either join the central office or move to a different firm. Gradually, the most talented people take either of the two options. Unless these people can

be replaced by equally capable adjusters, the talent of the branch office gradually erodes, making it rely even more on central support. The cycle is reinforcing—as the central staff becomes better at intervening, the branch seeks their help more often.

3.3.3.3 *Using the Archetype*

In theory, any one of the four elements of the template—problem symptom, symptomatic solution, side effect, and fundamental solution—can help us identify a Shifting-the-Burden structure at work. Side effects, however, are usually very subtle and difficult to detect from inside the system. Solutions such as alcohol use, increased marketing, oil imports, or federal insurance are more readily identified, but there may not be complete agreement on whether they are “symptomatic” or “fundamental.” Identifying problem symptoms such as high stress, falling revenues, energy shortage, or bank failures (see Figure 3.22) is probably the easiest way to begin filling out a Shifting-the-Burden template.

Problem Symptom	Symptomatic Solution	Fundamental Solution	Side Effect(s)
Slow/declining revenue growth	Increased Marketing	New Products	Diverts resources away from R&D; increased reliance on marketing
Bank Failures	FDIC, FSLIC	Prudent Banking Practices	Responsibility for protecting deposits is shifted to government
Employee Performance Problem	Manager "Provides" Solution	Necessary Training for Employee	Growing dependence on manager; decreasing confidence of employee
Low Self-Esteem	Drug Use	Invest time in personal development	Drug Addiction; further debilitation of personal development

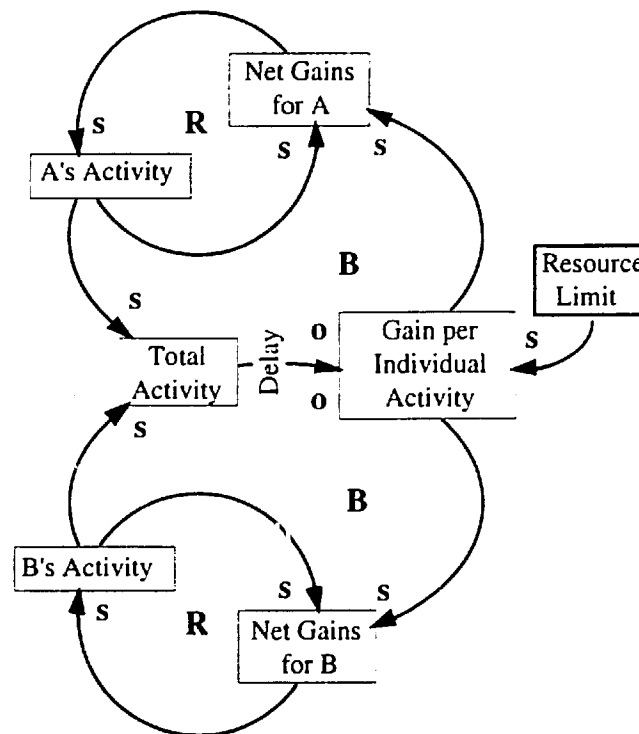
Table of Shifting-the-Burden Examples

Figure 3.22

Keeping in mind that the “rightness” of a solution depends on one’s perspective, it can be helpful to ask whether we are seeing the situation from the parents’, Helen Keller’s, or Ann Sullivan’s point of view. Examining a problem or issue from these different viewpoints can help us understand why a Shifting-the-Burden archetype is operating and discover a solution that is fundamental, not symptomatic.

3.3.4 Tragedy-of-the-Commons

At the heart of the Tragedy-of-the-Commons structure lies a set of reinforcing actions that make sense for each individual player to pursue (see Figure 3.23).



Tragedy-of-the-Commons Template

Figure 3.23

As each person continues his individual action, he gains some benefit. For example, each family heading to the community pool will enjoy cooling off in the swimming area. If the activity involves a small number of people

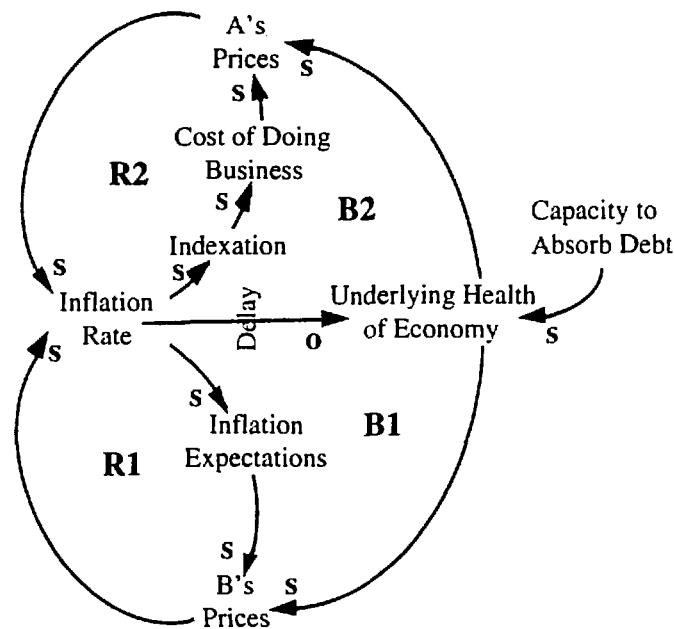
relative to the amount of "commons" (or pool space) available, each individual will continue to garner some benefit. However, if the amount of activity grows too large for the system to support, the commons becomes overloaded and everyone experiences diminishing benefits.

Traffic jams are a classic example of how a "public" good gets overused and lessened in value for everyone. Each individual wishing to get quickly to work and back uses the freeway because it is the most direct route. In the beginning, each additional person on the highway does not slow down traffic because there is enough "slack" in the system to absorb the extra users. At some critical level, however, each additional driver brings about a decrease in the average speed. Eventually, there are so many drivers that traffic crawls at a snail's pace. Each person seeking to minimize driving time has in fact conspired to guarantee a long drive for everyone.

This structure also occurs in corporate settings all too frequently. A company with a centralized salesforce, for example, will suffer from the Tragedy-of-the-Commons archetype as each autonomous division requests that more and more efforts be expended on its behalf. The division A people know that if they request "high priority" from the central sales support they will get a speedy response, so they label more and more of their requests as high priority. Division B, C, D, and E all have the same idea. The net result is that the central sales staff grows increasingly burdened by all the field requests and the net gains for each division are greatly diminished. The same story can be told about centralized engineering, training, maintenance, etc. In each case, either an implicit or explicit limit is keeping the resource constrained at a specific level, or the resource cannot be added fast enough to keep up with the demands.

3.3.4.1 Brazil's Inflation Game

When the shared commons is a small, localized resource, the consequences of a Tragedy-of-the-Commons scenario are more easily contained. At a national level, however, the Tragedy-of-the-Commons archetype can wreak havoc on whole economies. Take inflation in Brazil, for example. Their inflation was 367% in 1987, 933% in 1988, 1,764% in 1989, and 1,794% in 1990. With prices rising so rapidly, each seller expects inflation to continue. Therefore, seller B will raise his price to keep up with current inflation and hedge against future inflation (see Figure 3.24). With thousands of seller B's doing the same thing, inflation increases and reinforces expectations of continued inflation, leading to another round of price increases (R1).



Brazil's Inflation

Figure 3.24

Inflation also leads to indexation of wages, which increases the cost of doing business. In response to rising business costs, Seller A raises her price,

which fuels further inflation (R2). Since there are thousands of Seller A's doing the same thing, their collective action creates runaway inflation. The underlying health of the economy steadily weakens as the government and businesses perpetuate endless cycles of deficit spending to keep up with escalating costs. Over time, everyone grows increasingly preoccupied with using price increases to make profits rather than investing in ways to be more productive. Eventually the economy may collapse due to high debts and loss of global competitiveness, resulting in dramatic price adjustments (B1 & B2).

3.3.4.2 *Common "Commons"*

The most challenging part of identifying a Tragedy-of-the-Commons archetype at work is coming to some agreement on exactly what is the commons that is being overburdened. If no one sees how his or her individual action will eventually reduce everyone's benefits, the level of debate is likely to revolve around why individual A should stop doing what he is doing and why individual B is entitled to do what she is doing. Debates at that level are rarely productive because effective solutions for a Tragedy-of-the-Commons situation never lie at the individual level.

In the sales force situation, for example, as long as each division defines the commons to include only its performance, there is little motivation for anyone to address the real issue—that the collective, not individual, action of each division vying for more sales support is at the heart of the problem. Only when there is general agreement that managing the commons requires coordinating everyone's actions can issues of resource allocation be settled equitably.

3.3.4.3 *Managing the Commons*

Identifying the commons is just the beginning. Other questions that help define the problem and identify effective actions include: What are the incentives for individuals to persist in their actions? Who, if anybody, controls the incentives? What is the time frame in which individuals reap the benefits of their actions? What is the time frame in which the collective actions result in losses for everyone? Can the long term collective loss be made more real, more present? What are the limits of the resource? Can it be replenished or replaced?

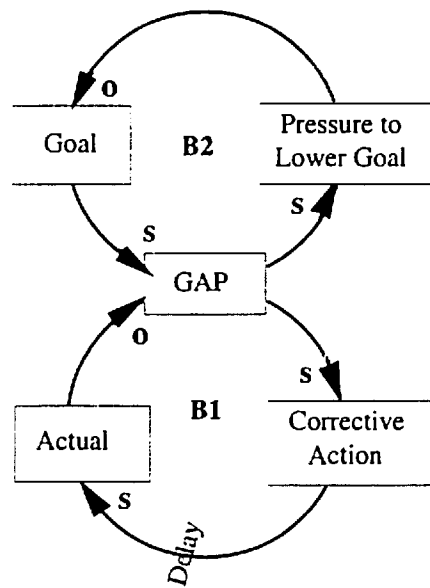
The leverage in dealing with a Tragedy-of-the-Commons scenario involves reconciling short-term individual rewards with long-term cumulative consequences. Evaluating the current reward system may highlight ways in which incentives can be designed so that coordination among the various parties will be both in their individual interest as well as the collective interest of all involved. Since the time frame of the commons "collapse" is much longer than the time frame for individual gains, it is important that interventions are structured so that current actions will contribute to long-term solutions.

3.3.5 **Drifting-Goals**

The Drifting-Goals archetype is helpful for trying to understand why an organization is not able to achieve its desired goals. Drifting-Goals occurs when the gap between a goal and the actual performance is reduced by lowering the goal. Because this often happens over a long period of time, the gradual lowering of the goal is usually not apparent until the decreasing performance measure has drifted so low that it becomes a crisis.

3.3.5.1 The Structure

The Drifting-Goals archetype works in the following manner (see Figure 3.25). There is a certain goal—implicit or explicit—which is compared to the current state of affairs. If a gap persists, corrective actions are taken to improve the current state and bring it in line with the goal. This forms the basic balancing loop (B1) at the heart of any system that strives for equilibrium. A delay between corrective action and actual state represents the fact that results may take from minutes to years to materialize, depending on the specific situation.



Drifting-Goals Archetype

Figure 3.25

Of course, there is more than one way to close the gap. In the Drifting-Goals archetype, a second balancing loop is driven by pressure to lower the goal. As the gap increases (or persists over a period of time), the pressure to lower the goal increases. If the pressure is high and persistent, the goal may be lowered, thereby decreasing the gap (loop B2). The critical difference between the two loops is that lowering the goal immediately closes the gap, whereas corrective actions usually take time.

3.3.5.2 Budget Deficits, Drifting Goals

The federal budget deficit provides a good example of the Drifting-Goals archetype at work (see Figure 3.26). A gap between the previously stated acceptable deficit level and the actual deficit can be closed by either reducing government spending (B3) or increasing tax revenues (B4). Bipartisan compromises, however, have usually resulted in *increased* government spending, mixed results in terms of taxes, and, consequently, higher deficits. The rising deficits have created an intolerable gap between the actual and stated maximum acceptable deficit, creating pressure to raise deficit targets, and eventually resulting in higher maximum acceptable deficits (B5).

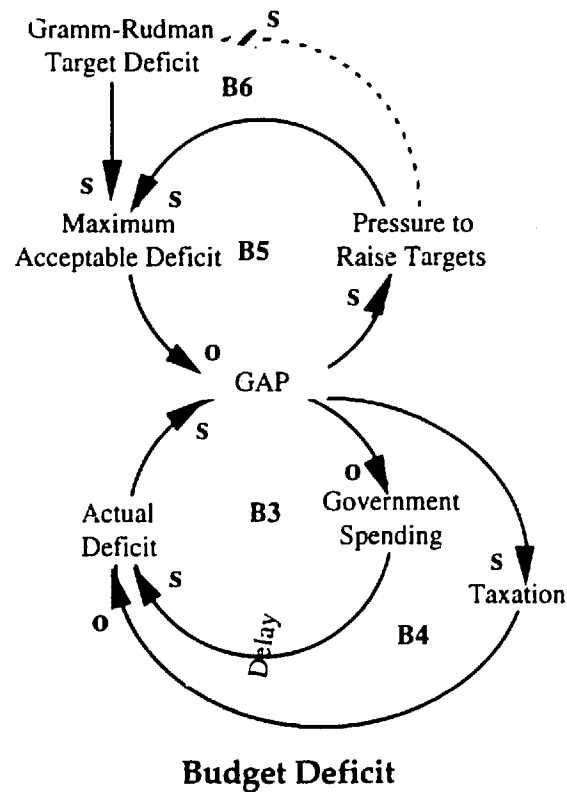


Figure 3.26

The Gramm-Rudman-Hollings bill (GRH) was believed by many to be a viable solution to the growing budget deficit problem. By making deficit reductions a law, it was intended to force bipartisan cooperation to eliminate deficit spending. If GRH target numbers are not met, mandatory cuts go into

effect, indiscriminately cutting billions of dollars from federal government programs and services. The GRH targets were meant to set a standard that lay outside of the current deficit-reinforcing system and would not, therefore, be susceptible to internal pressures.

In a recent interview (Scheibla, 1989), Senator Rudman reacted to pressures to lower the goal by suggesting that GRH targets may be changed as “long as they are accompanied by a major deficit reduction.” The net effect of Rudman’s revised proposal would be to re-introduce GRH targets into the Drifting-Goals archetype, which would negate its ability to anchor the deficit goal to an external source (B6).

3.3.5.3 *Using the Archetype*

Drifting or oscillating performance figures usually provide the signal that the Drifting-Goals dynamic is occurring and that real corrective actions necessary to meet the targets are not being taken. It may also mean current targets are being set more by past levels of performance than by some absolute standard (zero defects) or by something outside of the system (customer requirements).

A critical aspect of evaluating a Drifting-Goals scenario in an organization is to determine what drives the setting of the goal(s). In quality improvement efforts, for example, the quality goal can be affected by competitors’ quality, by customers’ expectations of quality, or by internal pressures. The relative strength of each potential influence will determine whether the quality will drift up, down, or oscillate. Goals located outside the system, like the original GRH targets, will be less susceptible to drifting goals pressures.

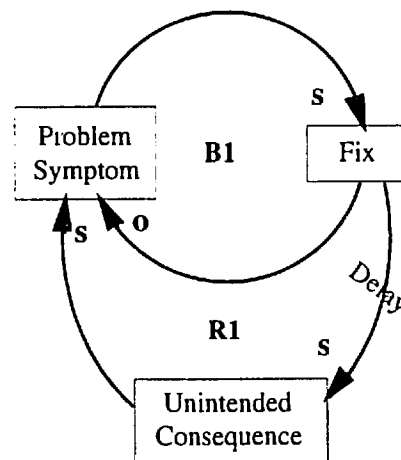
3.3.6 **Fixes-that-Fail**

“Today’s problems come from yesterday’s solutions” sums up the Fixes-that-Fail archetype. Its central theme is that most decisions carry short-term and

long-term consequences, and the two are often diametrically opposed. Oftentimes, a short-term solution will have unintended consequences that result in additional problems (or a worsening of the original problem).

3.3.6.1 *The Structure*

In a typical Fixes-that-Fail situation (see Figure 3.27), a problem symptom occurs that demands immediate resolution. A solution is quickly implemented that alleviates the symptom (loop B1). The relief is usually temporary, however, and the symptom returns, often worse than before. This happens because there are unintended consequences of the solution that unfold over a long period of time (loop R1), or as an accumulated consequence of repeatedly applying the solution, which exacerbate the original problem symptom.



Fixes-that-Fail Archetype

Figure 3.27

3.3.6.2 *Expediting Customer Orders*

Expediting customer orders, a common practice in many manufacturing firms, illustrates the Fixes-that-Fail archetype. For example, if a large semiconductor manufacturer is experiencing some production problems and

is running behind schedule on some shipments, their customers literally will have to shut down their production lines until they receive the chips. Therefore, once Customer A discovers the delivery delay, it will call the company and demand that its product be delivered immediately. The semiconductor company responds by assigning an expeditor to the complex task of tracking down A's order and pushing it through the line (see Figure 3.28). The company produces over a hundred different kinds of integrated circuits, and Company A has many different types on order. Finding and expediting A's order involve departments throughout the factory, and result in disruptions throughout the production line. Eventually, Company A's order is rushed through, resulting in a satisfied customer (loop B2).

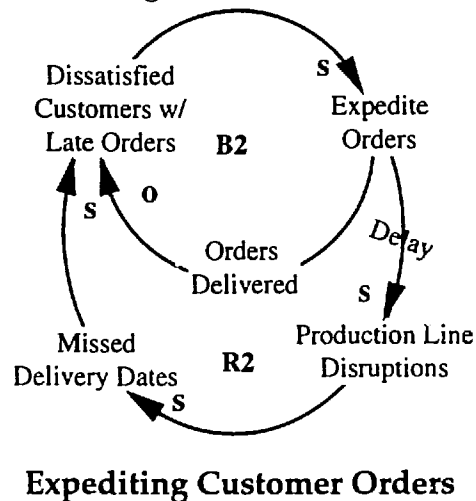


Figure 3.28

But no sooner has A's order left the warehouse when company B calls demanding to receive *its* orders immediately, and the process begins all over again. At the same time, somebody *else* is expediting for company C. Each customer's problem is resolved, but the number of problems rapidly increases. As a result, the production line is continually being disrupted—leading to more missed delivery dates and more customer calls (loop R2).

3.3.6.3 *Using the Archetype*

In most instances of Fixes-That-Fail, people are aware of the negative consequences of applying a quick fix. But the pain of not doing something right away is often more real and immediate than the delayed negative effects. If the long-term/short-term tradeoff was indeed one-for-one, where solving one problem today would create another one tomorrow, this strategy might be tolerable. But the reinforcing nature of unintended consequences ensures that tomorrow's problems will multiply faster than today's solutions.

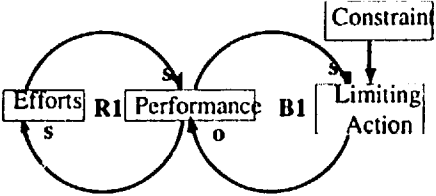
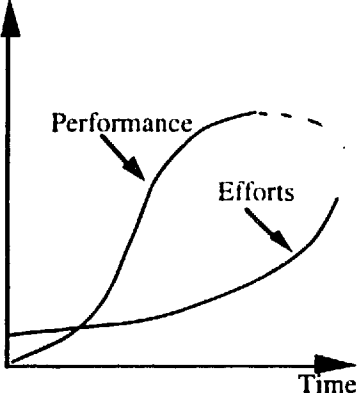
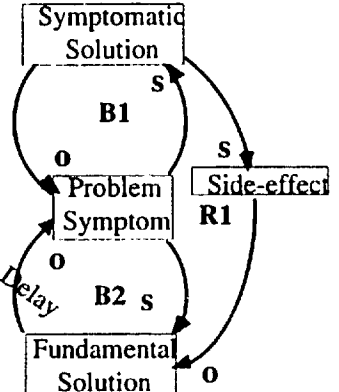
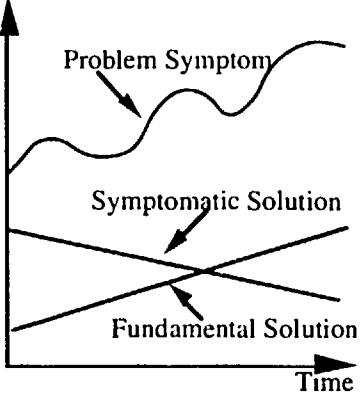
Breaking the Fixes-that-Fail cycle usually requires two actions: acknowledging up front that the fix is merely alleviating a symptom, and making a commitment to solve the real problem now. Launching a two-pronged attack of applying the fix *and* planning out the fundamental solution will break the problem-solution-problem cycle. The archetype can also be used to evaluate potential fundamental solutions, by mapping out any potential unintended side-effects.

3.3.7 **Summary Table of Systems Archetypes as Dynamic Scripts**

The archetypes can be viewed as dynamic scripts, each with a testable theory about dynamic behavior. The causal loop diagram "templates" provide a general framework for identifying "core categories" of data. The dynamic behavioral patterns provide a graphical representation of the dynamic hypothesis which each archetype is posing. Finally, there are also some prescriptive actions that are suggested by each systems archetype that supports a theory-driven inquiry into actions and their consequences.

Archetype	Dynamic Scripts and "Core Categories"	Dynamic Behavioral Patterns	Prescriptive Actions
<p style="text-align: center;">Drifting Goals</p>	<p>The Drifting-Goals archetype states that a gap between a goal and current reality can be resolved in two ways: by taking corrective action to achieve the goal, or by lowering the goal. It hypothesizes that, over time, the continual lowering of the goal will lead to gradually deteriorating performance.</p> <p><i>Core Categories: Goal, Actual, Gap, Corrective Action, Pressure to Lower Goal.</i></p>		<ul style="list-style-type: none"> Anchor the goal to an external frame of reference to keep it from sliding (e.g. benchmarking, voice of the customer). Determine whether the drift in performance is the result of conflicts between the stated goal and implicit goals in the system (such as current performance measures). Establish a clear transition plan from current reality to the goal, including a realistic time frame for achieving the goal.
<p style="text-align: center;">Fixes that Fail</p>	<p>The Fixes-that-Fail archetype states that a solution that is used to quickly solve a problem symptom can have unintended consequences that exacerbate the problem. It hypothesizes that, over time, the problem symptom will return to its previous level or become worse.</p> <p><i>Core Categories: Problem Symptom, Fix, Unintended Consequences.</i></p>		<ul style="list-style-type: none"> Focus on identifying and removing the fundamental cause of the problem symptom. If a temporary, short-term solution is needed, develop a two-tier approach of simultaneously applying the fix and planning out the fundamental solution. Use the archetype to map out potential side-effects of any proposed interventions.

Systems Archetypes as Dynamic Scripts
Figure 3.29a

Archetype	Dynamic Scripts and "Core Categories"	Dynamic Behavioral Patterns	Prescriptive Actions
<p style="text-align: center;">Limits to Success</p> 	<p>The Limits-to-Success archetype states that a reinforcing process of accelerating growth (or expansion) encounters a balancing process as the limit of that system is approached. It hypothesizes that, as the limit approaches, continuing efforts will produce diminishing returns.</p> <p><i>Core categories: Performance, Efforts, Limiting Action, Constraint.</i></p>		<ul style="list-style-type: none"> • Focus on removing the limit (or weakening its effects), rather than continuing to drive the reinforcing process of growth. • Use the archetype to identify potential balancing processes <i>before</i> they begin to effect growth. • Identify links between the growth processes and limiting factors to determine ways to manage the balance between the two.
<p style="text-align: center;">Shifting the Burden</p> 	<p>The Shifting-the-Burden archetype states that a problem symptom can be resolved in one of two ways: a symptomatic solution or a fundamental solution. It hypothesizes that once a symptomatic solution is taken, it produces a side-effect that systematically undermines the ability to apply a fundamental solution.</p> <p><i>Core categories: Problem Symptom, Symptomatic Solution, Fundamental Solution, Side-Effect.</i></p>		<ul style="list-style-type: none"> • Focus on the fundamental solution. If necessary, use the symptomatic solution only to gain time while working on the fundamental solution. • Elicit multiple viewpoints to differentiate between fundamental/symptomatic solutions and to gain consensus around an action plan. • Use the archetype to explore potential side-effects of any proposed solution.

Systems Archetypes as Dynamic Scripts
Figure 3.29b

Archetype	Dynamic Scripts and "Core Categories"	Dynamic Behavioral Patterns	Prescriptive Actions
<p>Tragedy of the Commons</p>	<p>The Tragedy-of-the-Commons archetype identifies the causal connections between individual actions and the collective results (in a closed system). It hypothesizes that, if the individual use of a common resource becomes too large for the system to support, the "commons" will become overloaded and everyone will experience diminishing benefits.</p> <p><i>Core categories: A's Activity, Net Gains for A, B's Activity, Net Gains for B, Total Activity, Gain per Individual Activity, Resource Limit.</i></p>		<ul style="list-style-type: none"> • Establish methods for making the cumulative effects of using the common resource more real and immediate to the individual users. • Re-evaluate the nature of the commons, to determine if there are ways to replace or renew (or substitute for) the resource before it becomes depleted. • Create a final arbiter who manages the use of the common resource from a whole-system level.

Systems Archetypes as Dynamic Scripts
Figure 3.29c

3.4 SUMMARY

In this chapter we have highlighted the important role that mental models play in our model of organizational learning. We were careful to distinguish mental model representations from mental models, noting that we can never say anything definitive about what people actually have in their heads but it is useful for us to refer to them as mental models. The literature review of mental model representations provided some general attributes—abstract, generic, runnable, theory-driven, and testable—which provide a set of criteria with which to assess the usefulness of mental model representation systems.

System dynamics methods of causal loop diagrams, systems archetypes, and computer models were proposed as being particularly powerful representational systems in domains where dynamic complexity is high. In particular, the use of systems archetypes as dynamic scripts was proposed for helping people learn and map their new understanding into their mental model and for communicating the new learning to others.

In the next chapter, we will outline a methodology that can be used to help make individual mental models of dynamically complex systems explicit and to build shared understanding to advance organizational learning.

Chapter 3 References

- Ackoff, Russell, *"The Second Industrial Revolution,"* (Unpublished manuscript).
- Anderson, John R. (1983). *The Architecture of Cognition*. Cambridge, MA: Harvard University Press.
- Argyris, Chris, Putnam, R., & Smith, D.M. (1985). *Action Science*. San Francisco: Jossey-Bass.
- Argyris, Chris, & Schön, Donald (1978). *Organizational Learning: A Theory of Action Perspective*. Reading, MA: Addison-Wesley.
- Baker, Edward M. (1989). The Evolution to Total Quality Excellence. *Dialogue, Winter*, 7-13.
- Bostrom, A., Fischhoff, B., & Morgan, G. M. (1992). Characterizing Mental Models of Hazardous Processes: A Methodology and an Application to Radon. *Journal of Social Issues*, 48(4), 85-100.

- Bower, G. H., Black, J. B., & Turner, T. J. (1979). Scripts in Memory for Text. *Cognitive Psychology*, 11, 177-220.
- deKleer, Johan, & Brown, John Seely (1983). Assumptions and Ambiguities in Mechanistic Mental Models. In G. D. & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Erlbaum Associates.
- Fiske, Susan T., & Taylor, Shelley E. (1984). *Social Cognition*. New York: Random House.
- Forrester, Jay W. (1961). *Industrial Dynamics*. Cambridge, MA: Productivity Press.
- Forrester, Jay W. (1971). *World Dynamics*. Cambridge, MA: Productivity Press.
- Garnham, Alan (1987). *Mental Models: A Representation of Discourse and Text*. New York: Halsted Press.
- Gentner, Dedre, & Gentner, Donald (1983). Flowing Waters or Teeming Crowds: Mental Models of Electricity. In D. Gentner & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Erlbaum Associates.
- Gentner, D. , & Stevens, A.L. (1983). *Mental Models*. Hillsdale, NJ: Erlbaum.
- Gioia, Dennis A., & Poole, Peter A. (1984). Scripts in Organizational Behavior. *Academy of Management Review*, 9(3), 449-459.
- Goal/QPC (1985). *Memory Jogger*. Methuen, MA: Goal/QPC.
- Hutchins, D. (1986). Having a Hard Time with Just-in-Time. *Fortune*(June 9), 64-66.
- Ishikawa, Kaoru. (1982). *Guide to Quality Control*. Ann Arbor, MI: UNIPUB.
- Johnson-Laird, P.N. (1983). *Mental Models*. Cambridge, MA: Harvard University Press.
- Kepner, Charles H. , & Tregoe, Benjamin B. (1965). *The Rational Manager*. New York: McGraw-Hill.
- Kim, Daniel H. (1990). *Toward Learning Organizations: Integrating Total Quality Control and Systems Thinking*. Cambridge, MA: Pegasus Communications.
- Kim, Daniel H. & Burchill, Gary (1992). Systems Archetypes as a Diagnostic Tool. In *Proceedings of the 1992 International System Dynamics Conference*. Utrecht, Netherlands:
- Larkin, Jill H. (1983). The Role of Problem Representation in Physics. In D. Gentner & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Miller, George A., Galanter, Eugene, & Pribram, Karl H. (1960). *Plans and the Structure of Behavior*. New York: Henry Holt.
- Minsky, Marvin (1985). *The Society of Mind*. New York: Simon and Schuster.
- Mizuno, Shigeru (1988). *Management for Quality Improvement: The 7 New QC Tools*. Cambridge, MA: Productivity Press.
- Norman, Donald A. (1983). Some Observations on Mental Models. In D. Gentner & A. Stevens (Eds.), *Mental Models* Hillsdale, NJ: Lawrence Erlbaum Associates.
- Pennington, N. , & Hastie, R. (1991). A Cognitive Theory of Juror Decision Making: The Story Model. *Cardozo Law Review*, 13, 519-557.
- Powers, William T. (1973). *Behavior: the Control of Perception*. Chicago: Aldine Publishing.
- Randers, Jorgen (Ed.). (1980). *Elements of the System Dynamics Method*. Cambridge, MA: MIT Press.
- Richardson, George P., & Pugh, Alexander L. (1981). *Introduction to System Dynamics Modeling with DYNAMO*. Cambridge, MA: MIT Press.
- Richmond, Barry (1990). Systems Thinking: A critical set of Critical Thinking Skills for the 90's and beyond. In D. F. Andersen, G. P. Richardson, & J. D. Sterman (Ed.), *System Dynamics 1990*, 2 (pp. 934-947). Chestnut Hill, MA: System Dynamics Society.
- Richmond, Barry, Vescuso, Peter, & Peterson, Steve (1987). *An Academic User's Guide to STELLA*. Lyme, NH: High Performance Systems.
- Roberts, Nancy (1983). Testing the World with Simulations. *Classroom Computer News*, 3(3).
- Schank, R.D. & Abelson, R. (1977). *Scripts, Plans, Goals, and Understanding*. Hillsdale, NJ: Erlbaum.
- Scheibla, S.H. (1989). So Why Ain't It Dead? Gramm, Rudman & Hollings on the deficit dragon. *Barrons*, 69(July 10), 6-7.
- Schneiderman, Arthur M. (1988). Setting Quality Goals. *Quality Progress*, April, 51-57.
- Senge, Peter M. (1990a). The Leader's New Work. *Sloan Management Review*, 12(1), 7-23.

- Senge, Peter M. (1990b). *The Fifth Discipline*. New York: Doubleday.
- Sterman, John D. (1988). Strategy Dynamics: The Rise and Fall of People Express. In Cambridge, MA: MIT Sloan School of Management.
- Sterman, John D. (1989). Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment. *Management Science*, 35,(3), 321-339.
- Watson, James (1981). *The Double Helix: A Personal Account of the Discovery of the structure of DNA*. London: Weidenfeld & Nicolson.
- Weick, Karl E. (1991). The Nontraditional Quality of Organizational Learning. *Organization Science*, 2 (1), 116-124.
- Williams, Michael D., Hollan, James D., & Stevens, Albert L. (1983). Human Reasoning About a Simple Physical System. In D. Gentner & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Erlbaum Associates.

Chapter 4

A Methodology for Linking Individual Mental Models to Shared Mental Models

"The real voyage of discovery consists not in seeking new landscapes but in having new eyes."

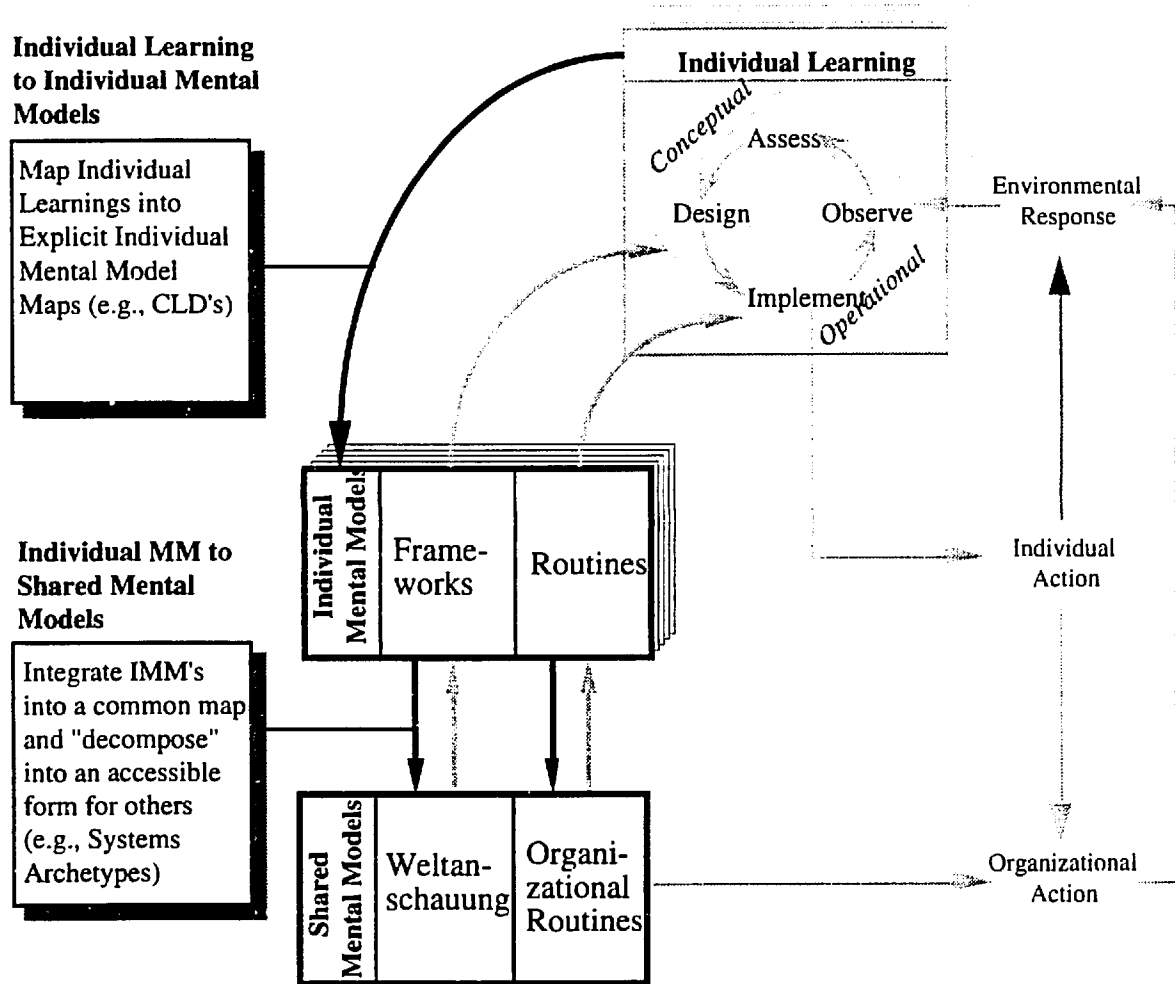
—Marcel Proust

4.0 INTRODUCTION

In Chapter 3 we reviewed a number of representational systems that operationalized mental models into an explicit form. We focused on a specific type of representational system from the field of system dynamics, namely, causal loop diagrams and systems archetypes. In this chapter, we will focus on the process by which one maps individual learning into individual mental models and how individual mental models can become shared mental models.

We begin in section 4.1 with an overview of the research, distinguishing between the scientific method and other ways of knowing and linking the OADI learning cycle with the scientific method. The Ladder of Inference and the grounded theory building approach is introduced, and their role in helping the research, and the researcher, stay grounded in the data is explained. Sections 4.2 and 4.3 present the methodology for mapping individual mental models and for diffusing them into shared mental models,

respectively. We conclude with a discussion on the strengths and weaknesses of this approach and possible next steps.



**Linking Individual Learning to Organizational Learning
through Shared Mental Models**

Figure 4.1

4.1 RESEARCH PROCESS AND APPROACH

The methodology used for mapping mental models draws on many different traditions: system dynamics modeling (Forrester, 1961; Randers, 1980), process consultation (Schein, 1987c; 1988), the clinical perspective (Schein, 1987a), action science (Argyris, Putnam, & Smith, 1985), grounded theory building (Strauss, 1987; Burchill, 1993) and TQM (Shiba & Graham, 1993).

In both the TQM and product development studies, the research teams were trained in the mapping methodology presented in this chapter, they collected interview data, and they participated in the analysis/synthesis process. In the TQM study, masters students were trained to gather interview data and to analyze the data as part of their thesis work (Brown & Tse, 1992; Peterson & Tok How, 1993; Balz & Garberding, 1993; Ehrler & Jansen, 1993). In the product development study, the line managers involved in the project were trained to conduct the interviews and participate in the analysis/synthesis of the data (Giancola, 1992; Roberts, 1992). In the TQM cases, the researchers were outsiders coming in to talk with the company insiders and eliciting their stories about their TQM implementation false-start experience. The analysis/synthesis was then performed by the researchers without the involvement of the interviewees. The product development case, on the other hand, involved company managers doing the data gathering themselves. Managers also played an active role in the analysis.

As part of the training, both groups were exposed to the problem-solution and problem-articulation models of management described in Chapter 1, the OADI cycle of learning described in Chapter 2, and the role of mental models in learning covered in Chapter 3. They were also trained in using the Ladder of Inference (Argyris, 1990), force-field analysis (Lewin, 1951), semantics (Thornton, 1992), causal loop diagrams (Kim & Burchill, 1992), and systems archetypes (Senge, 1990b). A brief description of each of the tools and methods is provided in the following sections, along with a general discussion of the scientific approach that was used.

4.1.1 Alternative Ways of Knowing

Stone (1978) discusses four different ways of “knowing” through which individuals establish, defend, or change their beliefs about the world (and therefore their mental models): tenacity, authority, intuition, and science. Knowing through *tenacity*, Stone asserts, is based on the inertia of prior beliefs; that is, we believe something is true simply because we have always believed it to be true. Any evidence we encounter that is contrary to those established beliefs is discounted. Corporate sacred cows are usually perpetuated through this kind of knowing—pure dogma based on tradition and/or religion.

Schein (1987b, p. 92) offers insights into England’s moralism-pragmatism scale that make useful distinctions between levels of authority, including the scientific method. He puts forth the following typology of authority:

1. Pure dogma, based on tradition and/or religion.
2. Revealed dogma—that is, wisdom based on trust in the authority of wise men, formal leaders, prophets, or kings.
3. Truth derived by a “rational-legal process, as when we establish the guilt or innocence of an individual by means of a legal process that acknowledges from the outset that there is no absolute truth, only socially determined truth.
4. Truth as that which survives conflict and debate.
5. Truth as that which works, the purely pragmatic criterion.
6. Truth as established by the scientific method, which becomes, once again, a kind of dogma.

Authority is another way of knowing. “Instead of simply holding on doggedly to one’s beliefs, appeal is made to some highly respected source to substantiate the views held” (Cohen & Nagel, 1936, p. 193). Two forms of authorities can be consulted: a recognized expert in a given field (e.g., a medical doctor or a lawyer) and a higher authority who is believed to be infallible or whose opinion is seen as final (e.g., a philosopher or a religious leader), referred to as “revealed dogma” by Schein (1987b). Although the first kind of authority may be based on more reasonable assumptions than those

experts were more trained to know about their specialized areas, both are inadequate when it comes to resolving differences of opinion among different experts. Reliance on experts also does not protect against the possibility of their being wrong. This is one of the main reasons why total quality management (TQM) is based on management by facts to replace management by opinion (which usually means somebody in a position of authority), by expertise or hierarchy.

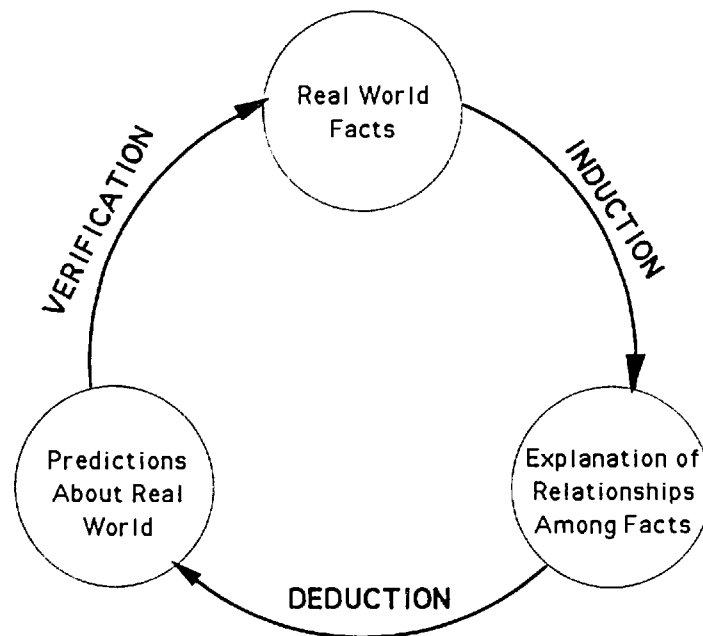
Knowing through *intuition* relies on the “appeal to ‘self-evident propositions’—propositions so ‘obviously true’ that the understanding of their meaning will carry with it indubitable conviction of their truth” (Cohen & Nagel, 1936, p. 194). The problems with this approach is that self-evident truths can end up being wrong (high quality equals high cost) or two “truths” that contradict each other can not be resolved on the basis of intuition alone. Another problem with knowing through intuition is that the process and the reasons for knowing are not made explicit or known, making it difficult to transfer the knowledge or examine its basis when we encounter contradictory evidence. This is particularly problematic for organizations since they lose all the intuitive knowledge of their best managers when they leave.

4.1.2 The Scientific Method

According to Hempel (1965, p. 141), “Science aims at knowledge that is *objective* in the sense of being intersubjectively certifiable, independently of individual opinion or preference, on the basis of data obtainable by suitable experiments or observations.” The *scientific method* is characterized by a continuous cycle of observing facts about the real world, building explanations about the relationships among the facts, making predictions

about the real world based on that understanding, and verifying those prediction by making more observations (see Figure 4.2).

A hallmark of the scientific method is the testing of hypothesis in such a way that it allows new facts to discredit them. It is this possibility of continually testing and changing beliefs that are no longer supported by currently available facts that distinguishes the scientific method from the other methods of “knowing” described above. The scientific method is not immune, however, from becoming another form of “knowing through tenacity.” All ways of knowing are susceptible to that possibility. As Schein (1987b) points out, truth, as established by the scientific method, becomes, once again, a kind of dogma. It is only through the continuous practice of open questioning and testing that separates the scientific method from the other ways of knowing.

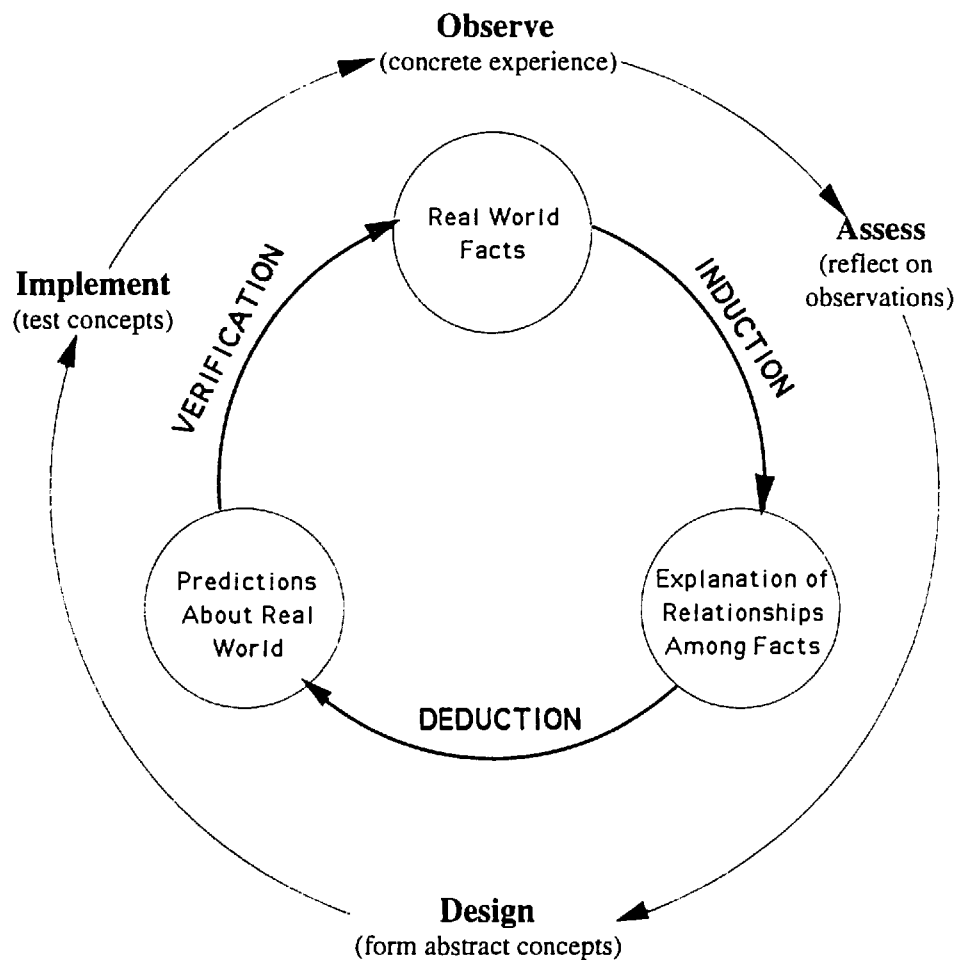


Model of the Scientific Method

(source: Stone, 1978)

Figure 4.2

The familiar P-D-C-A quality improvement cycle of TQM and our OADI model of individual and organizational learning are both rooted in the induction-deduction-verification cycle of the scientific method shown in Figure 4.3. In the OADI cycle, observations are made in the real world, rooted in concrete experience. In the assessment phase, explanations about the observations are created through an inductive process of moving from the specifics (a broken pump) to general concepts and explanations (infrequent maintenance leads to equipment failures).



The OADI Learning Cycle and the Scientific Method

Figure 4.3

This is the process through which we build (implicit and explicit) mental models of the world. Based on our mental models, we make predictions

about the world through a deductive process and design action plans to bring about the predicted behavior or outcome. We then implement those designs and verify whether they matched our predictions or not by making further observations. Our mental models are always “active” and affect each phase of the learning cycle.¹

How do mental models based on other ways of “knowing” differ from those based on the scientific method? This is a significant point—*there is no difference*. The mental models themselves are the same. The difference lies in the *process* by which those mental models are used, tested, and changed. It’s not the “what” you know that is different but the “way” of knowing. The scientific method is a more powerful way of knowing (and learning) because of its openness to falsifiability.

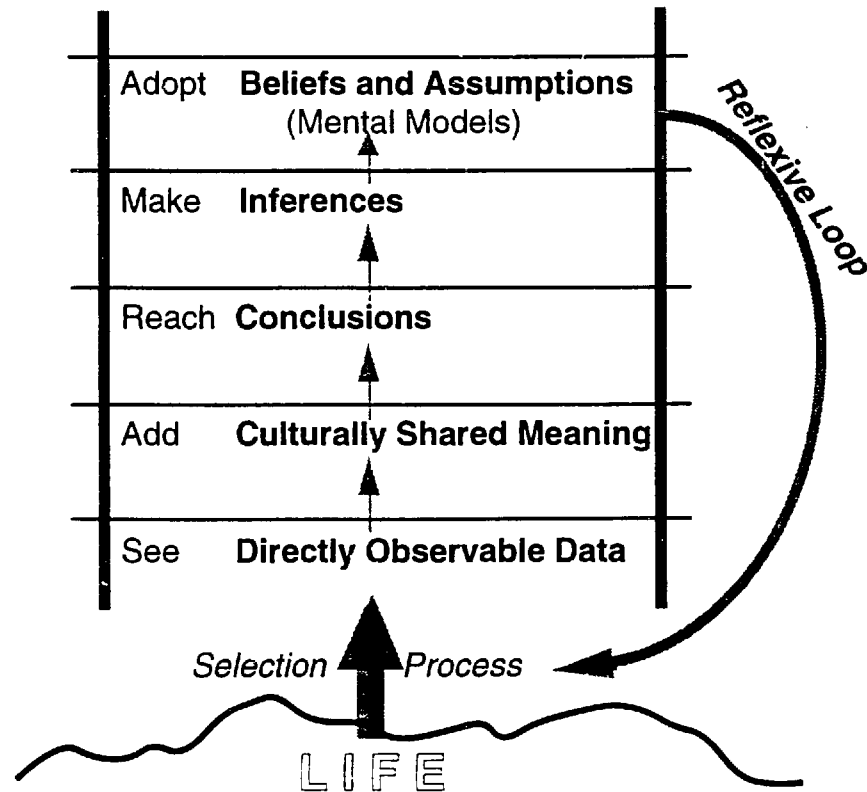
But, even those following the scientific method can become trapped in a particular way of looking at the world through one’s own paradigmatic lens (Kuhn, 1962; Schein, 1987b). The following section on the Ladder of Inference and the reflexive loop addresses this point.

4.1.3 Ladder of Inference and the Reflexive Loop

How can we deconstruct our ingrained ways of looking at the world and begin to see how we are actively constructing our interpretation of reality? One tool that can help is the Ladder of Inference, developed by Chris Argyris (1990) which provides a framework for seeing how mental models are constructed (see Figure 4.4). It graphically depicts the process people use to draw conclusive opinions and judgments from data, showing that individual evaluations are, in reality, highly abstract and inferential. At the bottom of

¹My motivation in reviewing such basics is to make clear the link between the model of individual and organizational learning, the research methodology used, and the scientific method. The learning cycle is rooted in the scientific method.

the Ladder of Inference is directly observable data: those things that can be objectively observed. From that data, we add culturally shared meaning—that is, we interpret and make sense of an event by the norms of our culture.



The Ladder of Inference
(adapted from Isaacs (1992))

Figure 4.4

For example, suppose Bob, a colleague, walked into a 9:00 meeting at 9:15. The directly observable data is that Bob physically entered the room 15 minutes after the scheduled start time. What do we say to ourselves when we notice this? When managers are asked this question, typical responses are:

- "He's late."
- "He doesn't care."
- "His previous meeting ran late."
- "He's not a team player."
- "He's disorganized."

If we locate the responses on the Ladder of Inference, we see that most of them are on the higher rungs of the ladder, reflecting the conclusions and inferences of different individuals based on the single piece of directly observable data.

There is nothing inherently wrong about drawing inferences and conclusions from the events we observe. In fact, the ability to move quickly up the Ladder is what enables humans to make sense of the incredibly complex, infinitely-detailed world in which we live. It is impossible for us to see and absorb everything—we are constantly selecting out a narrow slice of life to focus on and understand. What we don't often realize, however, is that our set of beliefs and assumptions directly affect the selection process by which we receive new observable data. Argyris calls this process the *reflexive loop* because it happens subconsciously and involuntarily.

For example, if we have concluded that Bob doesn't really care about meetings and is not a team player (our mental model of Bob), what do we begin to notice about him? We take note of all the times he shows up late, and we ignore or aren't aware of all the times he is on time. We notice that Bob does not say much at meetings, but don't register the fact that a few people always dominate the conversation and that there are others who say even less than Bob. We continually filter out any information that doesn't fit in with the mental model we have created about Bob. In fact, *all* the data we see confirm our beliefs and assumptions about Bob. We leap from data up to beliefs and assumptions, and then operate as if the assumptions *are* the data. We literally believe we see "directly observable instances" of Bob being uncaring. The reflexive loop can also be called the paradigm-creating loop, because it is the process through which, over time, we develop a shared set of corporate assumptions and beliefs about reality.

Womack, et al., (1990, p. 77) give a striking example of how this paradigm-creating process literally affects our ability to see. The authors describe the lean production manufacturing system, which uses less material, requires smaller inventories, has a shorter design time, and produces fewer defects than the traditional mass production system. The authors tell the story of a General Motors plant manager's reaction after seeing a lean production plant in Japan: He "reported that secret repair areas and secret inventories had to exist behind the plant, because he hadn't seen enough of either for a 'real' plant." In actuality, there is no rework area in that plant—they drive the cars right off the assembly line and onto the ships. The GM manager's paradigm of a "real" plant kept him from seeing that there might be an alternative way to produce cars.

The Ladder of Inference was presented and used as part of the interview training for both the TQM study (Chapter 5) and the Product Development study (Chapter 6) for several reasons. First, it made those involved in the research more aware of how their own ways of thinking can affect what they see and hear. During the interview process, the Ladder of Inference provided them with a visual framework for seeing at what level of inference the interviewees were telling their stories and served as a reminder to get down to the level of observable data whenever examples were given at the higher rungs. In the data analysis and mapping steps, the Ladder of Inference was useful for checking for any "leaps of logic" that may have been used in creating links between variables.

4.1.4 Grounded Theory Building

The purpose of this methodology is to help elicit and map an individual's mental model. In other words, it is a process of exploration and discovery

which requires an inductive methodology that can build a theory from the ground up. Thus, we want to be careful to first listen to the data *before* we start making deductions and drawing our own conclusions. Dewey (1926) wrote of this need for “intellectual disrobing” of the habits we take on and wear when we assimilate the culture of our own time and place, to inspect them critically to see what they are made of and what wearing them does to us. In other words, he was asking people to surface and to test their assumptions. This effort, he believed, could only take place when one was grounded in tangible experience—a basic tenet of grounded theory building.

Dewey asserted that the discipline required of a scientific work in physics or astronomy, the careful record of its calculations, and the deductions that were derived from past observations and experiments were more than just a record. These records of past observation also served as an indication, an assignment of further observations and experiments to be performed. They acknowledge that the work is ongoing, that it is rooted in real experience, and can be tested further. Dewey was responding to the philosophers in his day who were deriving theories and following trends from the same set of beliefs and prejudices, taking themselves farther and farther from reality. In effect, they were operating within a paradigm, blinded by it and unable to see or to value what was outside of it.

In much the same way, our study of the complexities of management and organizational learning must be grounded in observable data; the more complex the interrelationships, the more important the data becomes. Dewey called on the rigors of the scientific method as a model for the kind of study—disciplined, open to refutation, and continuously observing and testing one’s own and other’s theories—that science has been engaged in to learn about the physical world.

In normal science, the subject matter is the physical world, which gets broken down into smaller and smaller fragments for closer scrutiny. Observations are recorded, hypotheses and predictions are made to explain the natural phenomena being studied, and more observation is conducted. But the subject matter of organizations is people in their work environments, and the fragmentation that is helpful in normal science is counter-effective in the study of social systems. Compared to the complexities of organizational learning,

[t]he areas investigated by normal science are, of course, minuscule...By focusing attention upon a small range of relatively esoteric problems, the paradigm forces scientists to investigate some part of nature in a detail and depth that would otherwise be unimaginable." (Kuhn 1962, p. 24).

We need to take a broader perspective because in the social sciences, more than in the physical and natural sciences, the importance of the interconnections is far greater than the importance of the pieces. We need to develop a different approach because

a paradigm can insulate the community from those socially important problems that are not reducible to the puzzle form, because they cannot be stated in terms of the conceptual and instrumental tools the paradigm supplies...One of the reasons why normal science seems to progress so rapidly is that its practitioners concentrate on problems that only their own lack of ingenuity should keep them from solving. (Kuhn, 1962, p. 37)

The rapid progress comes at a price because we do not ever get the whole picture but only pieces of it. As Strauss (1987, p. 6) suggests,

One of our deepest convictions is that social phenomena are complex phenomena. Much social research seems to be based on quite the opposite assumption: either that, or researchers working in various research traditions describe or analyze the phenomena they study in relatively uncomplex terms, having given up on the possibility of ordering the "buzzing, blooming confusion" of experience except by ignoring "for a time" its complexity.

Although Strauss (1987, p. 2) says that "*qualitative* researchers tend to lay considerable emphasis on situational and often structural contexts, in contrast to many *quantitative* researchers, whose work is multi-variate but often weak

on context," neither research needs to remain in that state. Qualitative research can be more multi-variate just as quantitative research does not need to be devoid of structural context. As Strauss (1987, p. 10) sees it:

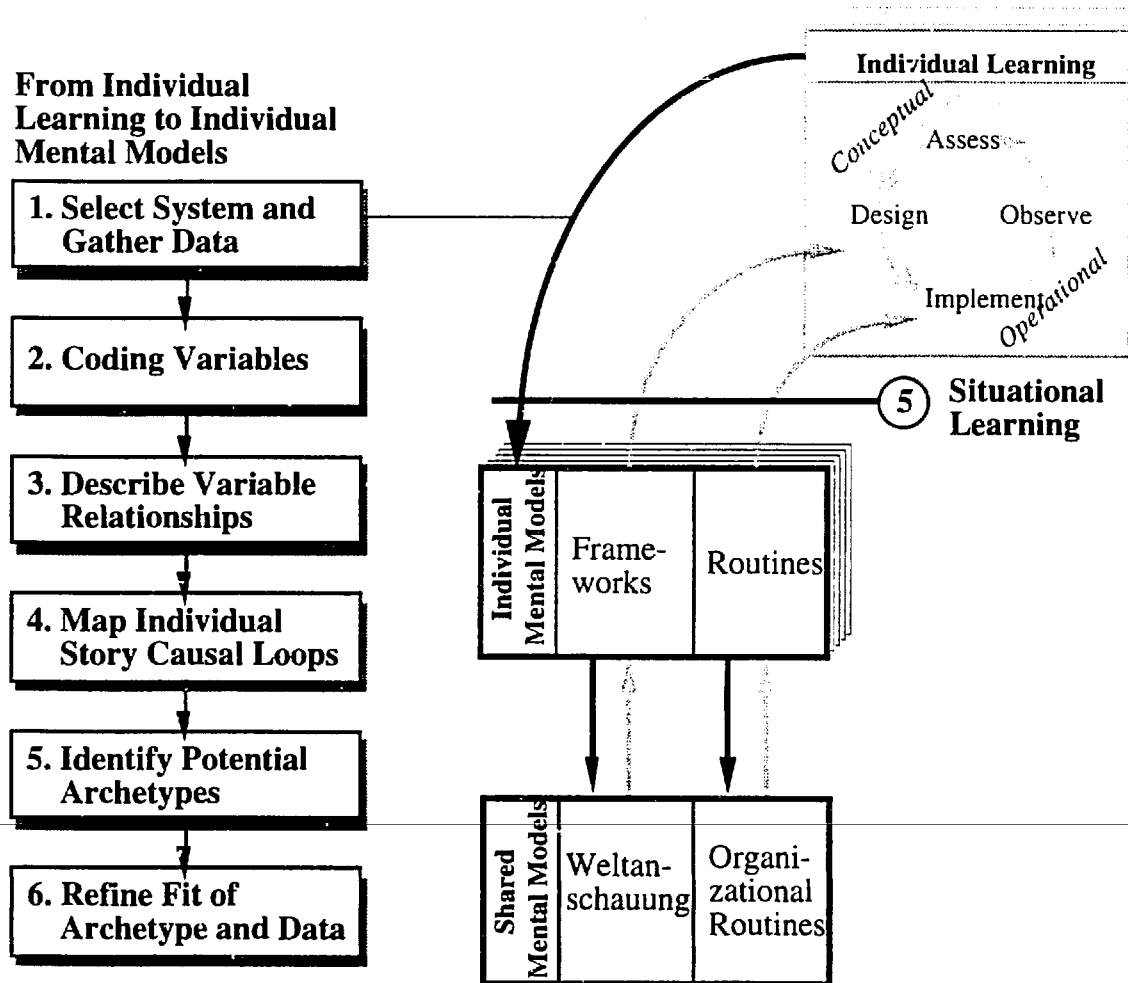
[t]he basic question facing us is how to capture the complexity of reality (phenomena) we study, and how to make convincing sense of it. Part of the capturing, of course, is through extensive data collection. But making sense of complex data means three things. First, it means that both the complex interpretations and the data collection are guided by successively evolving interpretations made during the course of the study. (The final products are analyses done at a relatively high level of abstraction: that is, *theories*.) The second point is that a theory, to avoid simplistic rendering of the phenomena under study, must be conceptually dense—there are many concepts, and many linkages among them. The third point: It is necessary to do detailed, intensive, microscopic examination of the data in order to bring out the amazing complexity of what lies in, behind, and beyond those data.

Grounded theory building is characterized by a process that involves an exhaustive coding process that is rooted in the data. In sections 4.2 and 4.3, we will describe a methodology that is consistent with Strauss' three points *and* strong on context.

4.2 METHODOLOGY FOR MAPPING INDIVIDUAL MENTAL MODELS

Closing the *situational* learning disconnect identified in Chapter 2 requires developing the ability of individuals in the organization to transfer learning from a specific situation into more general maps that can guide them in future situations. Making mental models explicit and clear, however, depends on having the appropriate tools for the type of knowledge being mapped. Although the English language is useful for communicating on many different levels, it can be wholly inadequate for descriptions of complex phenomena of a dynamic nature. Thus, English may be perfectly adequate for making one's mental model of a Shakespearean play explicit, and at the same time, be grossly ineffective in explicating a mental model of how the wage-price spiral affects capital investment decisions.

As described in Chapter 3, the understanding of dynamically complex systems requires an appropriate set of tools with which to make mental models of such systems explicit. In this section we outline a six-step methodology for mapping individual mental models into individual causal loop systems maps (see Figure 4.5), and in the following section (4.3), we explicate the process of developing shared mental models from those



From Individual Learning to Individual Mental Models

Figure 4.5

individual systems maps. The methodology is based on grounded theory methods (Glaser, 1978; Glaser & Strauss, 1967; Strauss, 1987) and system dynamics methods (Forrester, 1961; 1979; Randers, 1980; Richardson, 1981;

Richardson & Pugh, 1981; Vennix, 1990). It also draws on the recent work on inductive systems diagrams (Kendrick, 1992) and on systems archetypes as a diagnostic tool (Kim & Burchill, 1992).

In describing the steps below, examples will be drawn from the different TQM studies whenever it is appropriate (Brown & Tse, 1992; Peterson & Tok How, 1993; Balz & Garberding, 1993; Ehrler & Jansen, 1993).²

4.2.1 Step 1: Select System and Gather Data

There are two primary tasks associated with this stage of the research: identifying the system or event to be investigated and collecting the appropriate data. The focus of the investigation is established by identifying significant problems and their symptoms. (The symptoms themselves are not expected to represent the high leverage intervention points but will serve as the starting point of the investigative trail.) A significant problem can be one which is important to the organization and has persisted despite efforts to eliminate it, or one which is present in a variety of organizational settings and is therefore difficult to define. In describing the problem, we start with the symptoms which are readily evident and attempt to characterize them, preferably in measurable terms.

There are three main advantages to clarifying the target early in the investigative effort. First, since the system is bounded, identifying the most appropriate informants is more manageable. Second, the informants are able to reflect on their *actual* experience rather than hypothetical or abstract events. Third, the amount of time required to conduct the analysis is minimized.

²The author and Gary Burchill helped supervise these master's theses as part of an ongoing study of TQM implementation false-starts.

Semi-structured interviews are conducted by two field researchers with one person leading the interview while the other is responsible for taking detailed notes. The lead interviewer is responsible for the flow of the interview. The note taker's job is to capture the conversation as accurately as possible.³ After the interviewee finishes telling his or her story in chronological order, the field researcher leads the interviewee through a force-field analysis (described below). In the final step of the interview, the person is asked to identify the dominant forces at work and trace through the causal explanations behind those forces.

The force-field diagram and analysis played a critical role in the interview process by providing a clear, structured (but not restrictive) framework for focusing the data-gathering. The following section explains the force-field concept and method in greater detail.

4.2.1.1 Field Theory and Force-field Analysis

In the words of Lewin (1951, p. 45) "field theory is probably best characterized as a method: namely, a method of analyzing causal relations and of building scientific constructs." The core view of field theory comes from the psychological perspective, owing to Kurt Lewin's training as a psychologist. Field theory began with the boundary zone of an individual life force and extended into the areas of planned change and organizational development. The field now spans the continuum between the individual level and social level of analysis and intervention.

³Although tape recorders were used in some of the interviews, the note-takers were asked to take notes as if they had no tape as a back up. This was to counter balance the tendency to "relax" the active note-taking since it was going to be "on the tape."

Field theory places a great deal of emphasis upon the formal properties of scientific constructs and applies the rigors of the mathematical approach to individual psychology. For example, Lewin (1951, p. ix) believed that

before a system can be fully useful, the concepts have to be defined in a way that (1) permits the treatment of both "qualitative" and "quantitative" aspects of phenomena in a single system, (2) adequately represents the conditional-genetic (or causal) attributes of phenomena, (3) facilitates the measurement (or operational definition) of these attributes, and (4) allows both generalization to universal laws and concrete treatment of the individual case.

In field theory, the "method of construction" serves as a guide with which appropriate "elements of construction" are developed as well as ways of combining these elements into a system of concepts. For example, conceptual type or dimension is analogous to physical dimensions such that psychological position can be thought of in terms of spatial relations of regions where psychological concepts which have the conceptual dimension of position are things like group belongingness of an individual. Thus, all psychological variables that have the same dimension can be grouped and understood on the same plane. This would correspond to grouping things at a higher level on the Ladder of Abstraction (see section 4.2.2.2).

The concept of systems as "quasi-stationary equilibria" is somewhat similar to the system dynamics view of systems as being in dynamic disequilibrium seeking equilibrium. In field theory, quasi-stationary equilibria is maintained because the direction of the resultant forces around level L is toward L, their strength increasing with the distance from L. The resultant forces in the neighborhood of L is said to have the character of a "positive central force field" which is defined as a constellation of forces directed toward one region which keeps the system more or less where it is.

Any actions taken (or forces applied) on the system to change a certain level L will create internal resistance from the system that will tend to keep it at L.⁴

From a field theory point of view, successful change includes three aspects: unfreezing the present level, moving to the new desired level, and refreezing group life on the new level. Since staying at any level is determined by a force field, permanency implies making sure that the new force field is made relatively secure against change. Force-field analysis is a technique that provides a framework for analyzing the forces that keep a given system in its current quasi-stationary state and determining which forces to attempt to alter in order to unfreeze the system. By developing a diagram which has all the driving forces and resisting forces displayed on it, one can get a quick idea why the system is frozen in its current position.

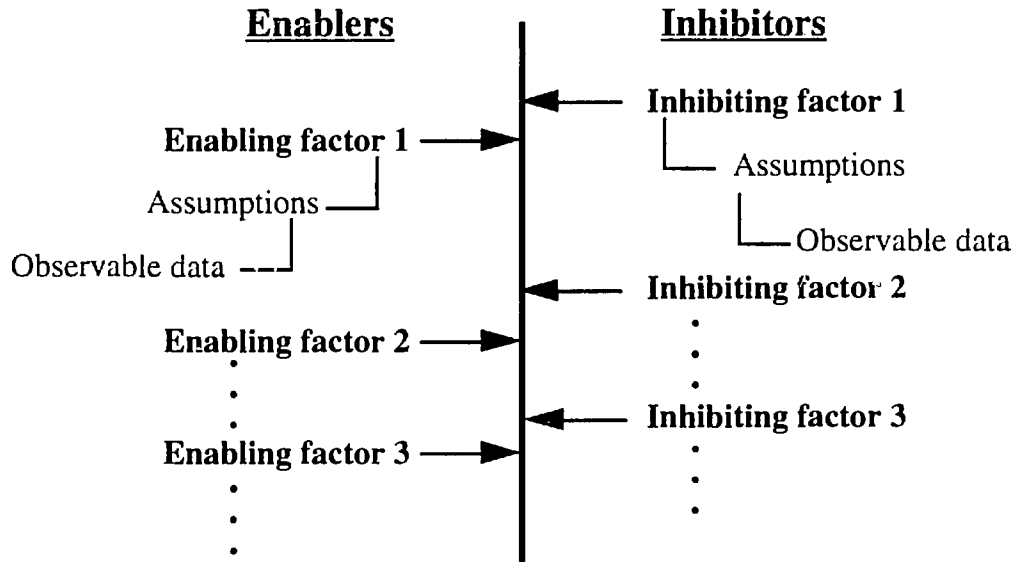
Force Field Diagram.

A force field diagram is constructed by drawing a line down the middle of a page and listing the situational forces on either side of the line with the enabling forces on the left side and the inhibiting forces on the right side. Arrows pointing into the center line are drawn to represent each force listed to represent the pressure that these forces are exerting on the balance.

We added a couple of modifications to the method by integrating some of the ideas from the Ladder of Inference framework (see Figure 4.6). For each of the factors that the interviewees offered as being an enabler or inhibitor, they were asked to state the assumptions that led them to include a particular factor and to provide a directly observable example that illustrated the validity of that assumption.

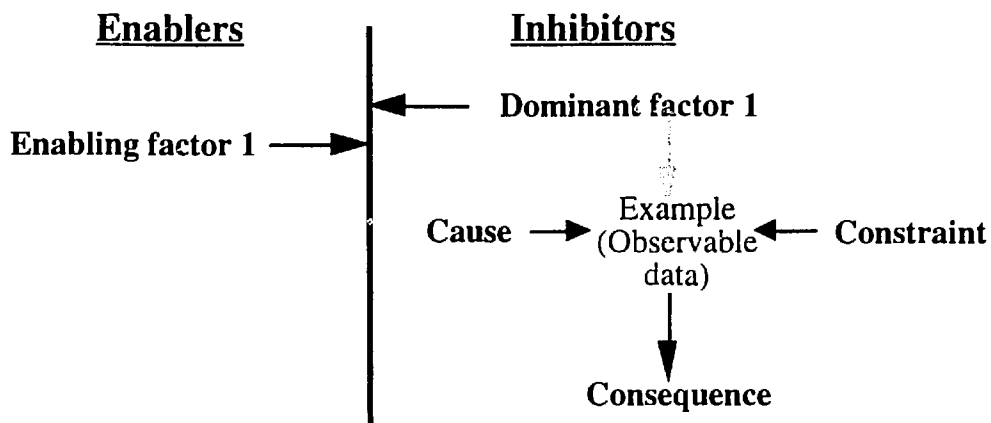
⁴A similar principle identified in system dynamics is compensating feedback.

In terms of our research goal of getting grounded data through interviews, the modified force-field method provided a way to maintain focus and provide the data needed to build systems maps without making it so structured that the interviewees did not have the freedom to say things that we did not have the foresight to ask.



Force Field Diagram

Figure 4.6



Causes, Constraints, and Consequences

Figure 4.7

An additional step was added to the force-field analysis based on the difficulty experienced in constructing causal connections among the variables with earlier interview data. Each interviewee was asked to identify what each felt to be the three most dominant forces and to provide the causes, consequences, and constraints for each force (see Figure 4.7).

4.2.1.2 Case Settings

In the TQM setting, we began our study by choosing to focus on well-defined TQM implementation “false starts” in order to bound the system we would be investigating. In the product development setting, we focused on a specific product development team responsible for developing one product.

Our data collection relied mostly on informant interviews. In the TQM setting, we interviewed three different people in each organization, which allowed us to cross-check the individual stories for accuracy. We asked each interviewee to begin with a chronology of their experience (the story of the “false start” in the TQM setting and history of involvement in the product development setting). A time line was used at times to anchor specific reference points.

4.2.2 Step 2: Coding Variables

After gathering the initial round of data, the goal is to extract important variables from the “raw” interview data.

4.2.2.1 Open Coding

The main purpose of open coding is to produce concepts that fit the data. This is accomplished by scrutinizing every sentence to find as many concepts that relate to the issue under investigation as possible. This often involves a

line by line, word by word analysis. Strauss (1987, p. 30) offers the following rules of thumb:

1. Look for in-vivo codes, terms used by the people who are being studied.
2. Give a provisional name to each code, in-vivo or constructed. Do not be concerned initially about the aptness of the term—just be sure to name the code.
3. Ask a whole battery of specific questions about words, phrases, sentences, actions in your line-by-line analysis.
4. Move quickly to dimensions that seem relevant to given words, phrases, etc.
5. These dimensions should quickly call up comparative cases, if not then concentrate on finding them.
6. Pay attention to the items in the coding paradigm, as previously listed.

The open coding process generates a lot of concepts very quickly. It is necessary to reduce the large number of concepts generated into a manageable number of variables. This step is one of the main difficulties faced by novices in creating systems maps—the selection of variable names. A useful guideline is to borrow from a TQM technique called the KJ (or affinity) diagram on the use of semantics for creating variables for inclusion in an archetype (Mizuno, 1988).

4.2.2.2 *Semantics as a way of “Cleaning” Verbal Data*

According to Shiba (1992), we need to distinguish among and understand the four characteristics of language processing that is relevant in generating variable names:

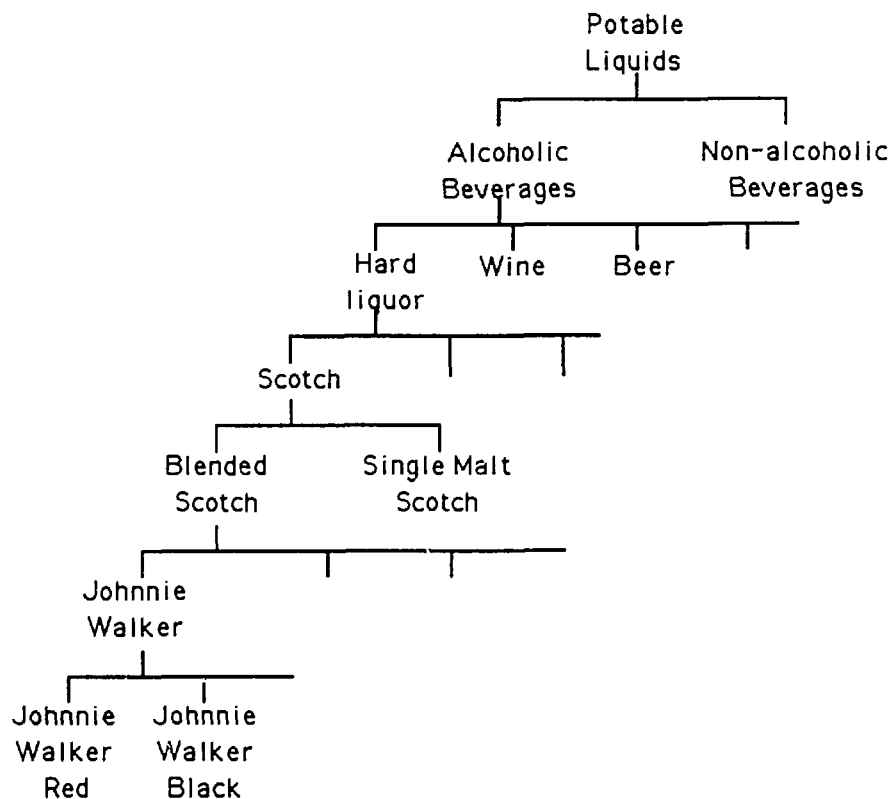
1. The dual role of language
2. The use of inference and judgment
3. The ladder of abstraction
4. Multi-valued vs. two-valued thinking⁵

Shiba distinguishes between the language of reports and the language of affection. The first is for communicating information; the latter is for communicating emotions. When using language to transmit the same

⁵There are similarities between these characteristics and the typical guidelines given for constructing causal loop diagrams (Richardson & Pugh, 1981, pp. 28-29).

meaning of information to everyone, one needs to use the language of reports. In our everyday communication, we usually use a mixture of both, so it is not surprising that most people include both when formulating variable names. "Cleaning" verbal data into a commonly understood and usable form requires stripping away qualifying adjectives (e.g., smart managers) as well as judgment and inference (which is the equivalent of walking down the Ladder of Inference).

The Ladder of Abstraction is another important semantic tool because it makes the user very conscious of how far away the variables are from concrete facts and directly observables (see Figure 4.8). For example, Johnnie



Ladder of Abstraction

Figure 4.8

Walker Red is a specific brand of Scotch. If we are talking about alcoholic beverages, it would be very low on the Ladder of Abstraction. Systems maps

are generally constructed at a highly-aggregated level while most people's day-to-day experience is rooted at a much lower level. The Ladder of Abstraction coupled with the Ladder of Inference can serve as a useful guide for walking people down the ladder to a level at which the dynamics makes more sense to them.⁶

The fourth guideline is a very familiar one. We want to use terms that can vary over time, not just switch back and forth like an on-off switch.

4.2.2.3 Case Setting

The following is an example of a coding session from one of the TQM study cases (Ehrler & Jansen, 1993, p. 22):

Interview Transcript:

Interviewer: What is an example of where "Improving your Process" was a reinforcing force for doing TQM?

Interviewee: Communications. We made a lot of assumptions about the material we provided to financial and operations people, that it was good in their desired. What they said was completely opposite to what we were giving them.

Coded Transcript:

What is an example of where ["Improving your Process"]¹ was a reinforcing force for [doing TQM]²?

[Communications]³. [We made a lot of assumptions about the material we provided to financial and operations people]⁴, [that it was good]⁵, [in their desired format]⁶. [What they said]⁷ was completely opposite to [what we were giving them]⁸.

The in-vivo codes (1-8) identified above are then placed on a list and referred to as coding continues, noting similar concepts encountered in later transcripts with the same coding number. Using the Ladder of Abstraction, similar concepts at different levels are grouped under one variable. For example, [communications]³, [what they said]⁷, and [talking about the way

⁶Although they may appear to be the same, there is a distinct difference between the Ladder of Inference and the Ladder of Abstraction. The Ladder of Abstraction distinguishes between the conceptual levels of aggregation of variables whereas the Ladder of Inference distinguishes between levels of meaning imposed on the data.

they send us information]15 can all be lumped into the variable name “Communications.”

4.2.3 Step 3: Describe Variable Relationships

For each significant variable identified, the next step is to identify other variables that are directly linked with it. This is where the force-field analysis in the interview is particularly helpful. The line of questioning that asks for causes, consequences, and constraints provides a rich description of the interviewees’ knowledge about the linkages among the variables. Pairwise directed arcs are then constructed using the “s” and “o” notation described in section 3.1.2.7. The signed links are rechecked with the data to ensure their “groundedness” (see Figure 4.9).

4.2.4 Step 4: Map Individual Story Causal Loops

Once all the interview data has been coded, the researcher should have a very good “feel” for the data. Relevant variables have been coded and listed, relationships between variables have been identified, and the pairwise directed arcs have been verified with the available data.

Step 4 is a crucial step in the overall process because here, for the first time, the system dynamics principle of feedback loop causality is integrated into the grounded theory building process. How does one begin to infer loops? How do we determine that the loops inferred from the data are justified? How do we check that the loops are formulated soundly? These are just a few of the questions that must be addressed in Step 4.

Step 4 is divided into 5 sub-steps. In order to begin developing complete causal loops from the directed arc fragments (which can have several variables strung together), the identification of a common “story” that runs through a group of fragments is helpful. The story helps “guide” the causal

loop diagram construction process by providing a theme against which the inclusion of a fragment can be assessed. A separate causal loop diagram is constructed for each story identified. Each causal loop diagram should be compared to the collected data to ensure they accurately represent the available facts. Additionally, the diagrams should be investigated for “leaps of inference,” i.e., can the diagram describe the patterns of events without explanation? Or, in terms of the Ladder of Inference, have we jumped too high on the ladder as we move from one variable to the next? The sub-steps involved in this process are listed below, followed by an extended example.

Step 4a: Name and Describe the Story

Step 4b: Construct Causal Loop with Fragments Relevant to Story

Step 4c: Check the Diagram with the Story

Step 4d: Fine-Tune the Loop Detail

Step 4e: Recheck the Story Diagram for “leaps of inference.”

4.2.4.1 An Extended Example

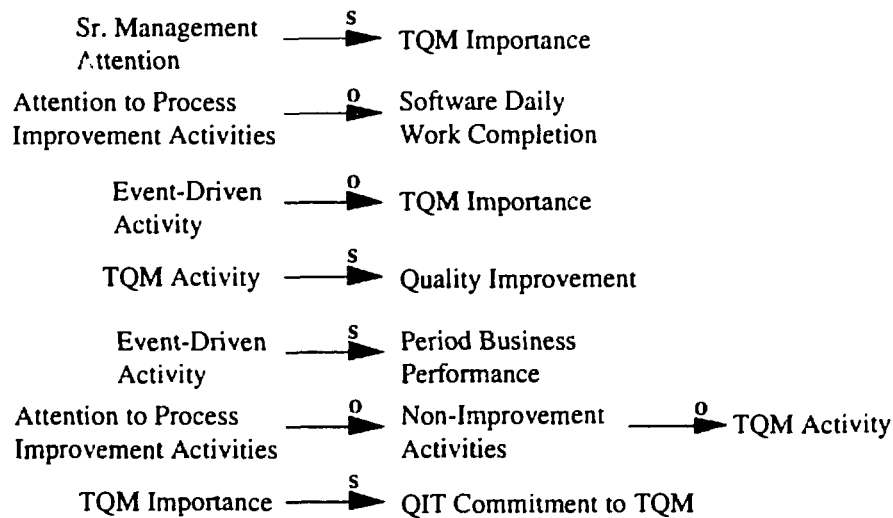
This extended example is based on one of the stories from Test, Inc., one of the TQM study sites covered in Chapter 5 (Brown & Tse, 1992). We will use the “Directive without Commitment” story to go through each of the five sub-steps outlined above.

Step 4a: Name and Describe the Story

The modeling process begins with reviewing the recorded interview session and notes. The primary issues and problems described in the interview are selected as potential “stories” for a causal loop diagram. Stories are selected which are recurring or systemic in nature and consequently have feedback characteristics, as opposed to one-time events which tend to be linear (no feedback). Each story is condensed until the essence or central points can

be told concisely. This “central story” is then used to focus the modeling efforts.

The story of “Directive without Commitment” describes an attempt to implement a TQM effort without real commitment by those who were responsible for launching the activity. Senior management charged a group of managers to implement TQM at a division by forming a Quality Improvement Team (QIT). In principle, the QIT team understood the long-term business implications of investing in TQM. The team, however, was composed of individuals who were very busy with daily demands on their time and were more concerned with outside (non-TQM) activities. The established group norms were that it was okay not to prepare for the meetings, that it was okay to be interrupted when doing TQM work, and that attendance at meetings was not all that important. The daily pressures of getting the work done was much more visible than the need to invest energy in things beyond the current period performance indicators, i.e., shipments.



Partial List of Directed Arc Fragments

Figure 4.9

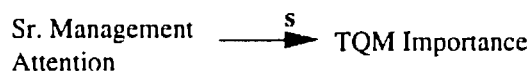
There was no feeling of ownership. If anything, there seemed to be a real rebellion against any serious attempts to make a commitment. Even if people had become interested in being more active, there were no training resources available for people to acquire TQM skills.

In constructing the loop diagram, we draw from work completed by following steps 1-3 above. A partial list of directed arc fragments are shown in Figure 4.9 as an example of the type of abstracted data that is available at this stage of the process.

Step 4b: Construct Causal Loop with Fragments Relevant to Story

Relevant fragments of the story are identified and then linked to form a causal loop diagram. The central story is told as a focus for the causal loop construction, and the elements are linked together to reflect the flow of the central story. The fragments or elements are examined in both the cause and effect direction. The question, "Why did this happen?" is asked in order to uncover further upstream causality. The question "what did this affect?" is asked in order to determine further downstream causality. The elements are linked in a feedback or causal loop structure until the causal loop reflects the story being told. Delays between cause and effect connections are inserted where appropriate.

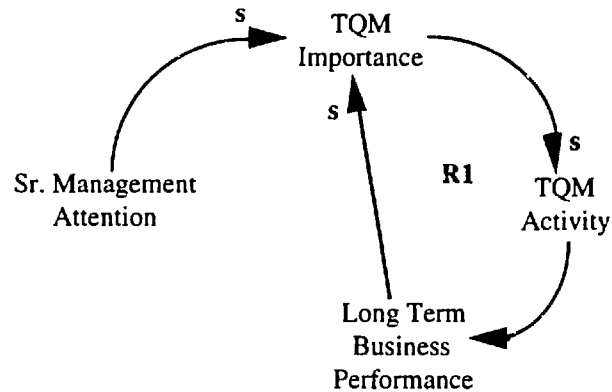
We started with the variable Sr. Management Attention, which is linked to TQM Importance with an "s" since they tried to raise the importance of TQM by focusing on it and launching a QIT (see Figure 4.10a).



Extended Example: Directive without Commitment

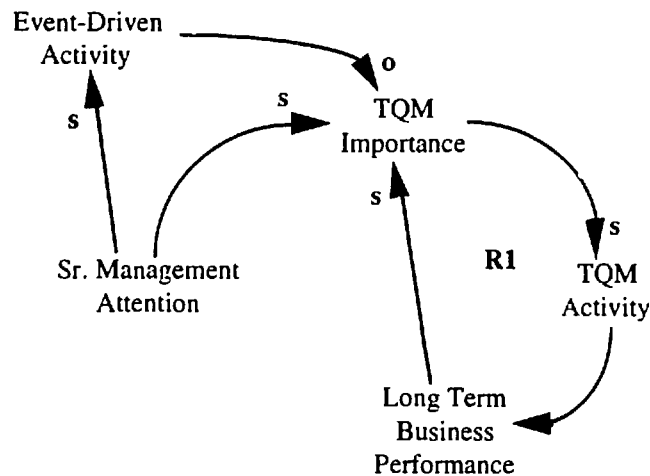
Figure 4.10a

The belief was that as awareness of the importance of TQM increased, TQM activity would increase. This would eventually lead to improved long term business performance which should further heighten the importance of TQM. The additional fragments create a reinforcing loop (R1 in Figure 4.10b).



Extended Example: Directive without Commitment

Figure 4.10b



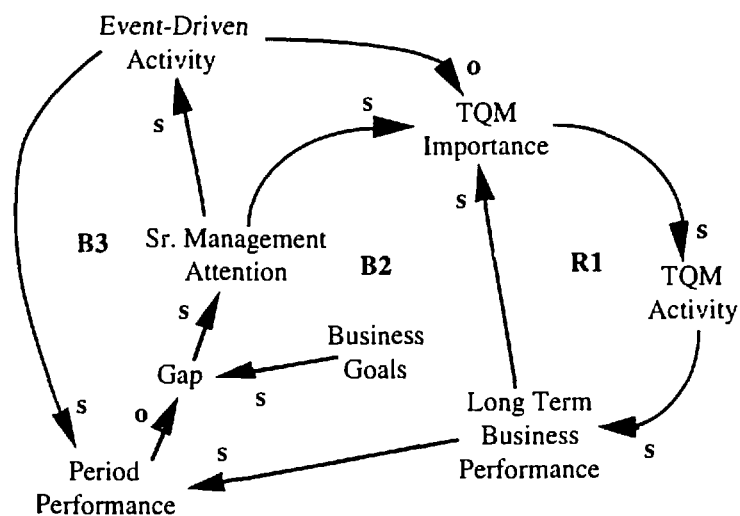
Extended Example: Directive without Commitment

Figure 4.10c

But in the story, senior management is also concerned with the day-to-day activities that have to get done. A link between senior management attention and event-driven activity is supported by the data. Another link in the story is between event-driven activity and TQM activity; as event-driven

activity increases, it takes time away from TQM activity, as was the case with the QIT members (Figure 4.10c).

We now look at the data to find any evidence that links senior management attention with anything else that would place it as part of a feedback loop. The link between Event-Driven Activity and Period Business Performance is identified in the data and added to the diagram. So is the link between Long-Term Business Performance and Period Business Performance. The belief that senior management attention is driven by a gap in period performance relative to business goals is also added to the diagram, which closes several feedback loops (B2 and B3 in Figure 4.10d).



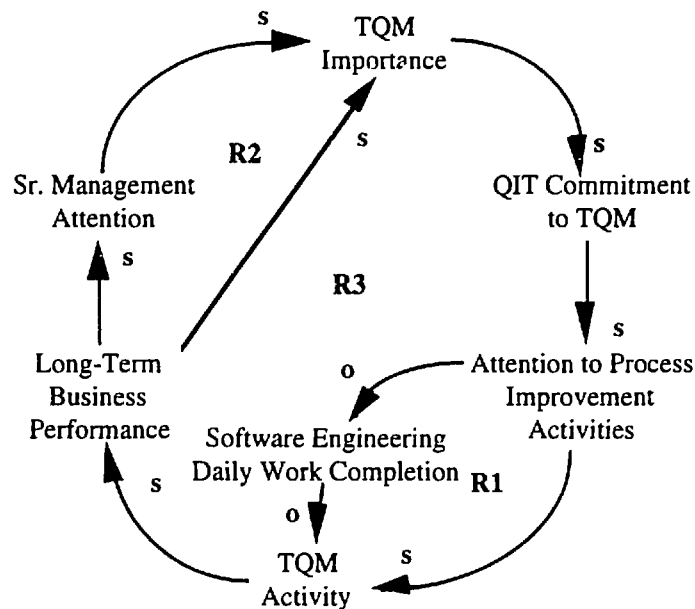
Extended Example: Directive without Commitment

Figure 4.10d

Step 4c: Check the Diagram with the Story

The causal loop diagram is checked against the story to see that a walk through the diagram actually describes what the story is telling. Changes are made in the diagram where appropriate to better reflect the dynamics of the story.

In checking the diagram with the story, the link between senior management attention and event-driven activity was not explicitly supported by the data. And, although it seemed logical, including Period Performance in *this story* about Directive without Commitment did not make sense with respect to the story line. The story included a bit about QIT Commitment to TQM as an important piece as well as the high daily workloads of the managers involved on the QIT team. The revised model is shown in Figure 4.10e.



Extended Example: Directive without Commitment

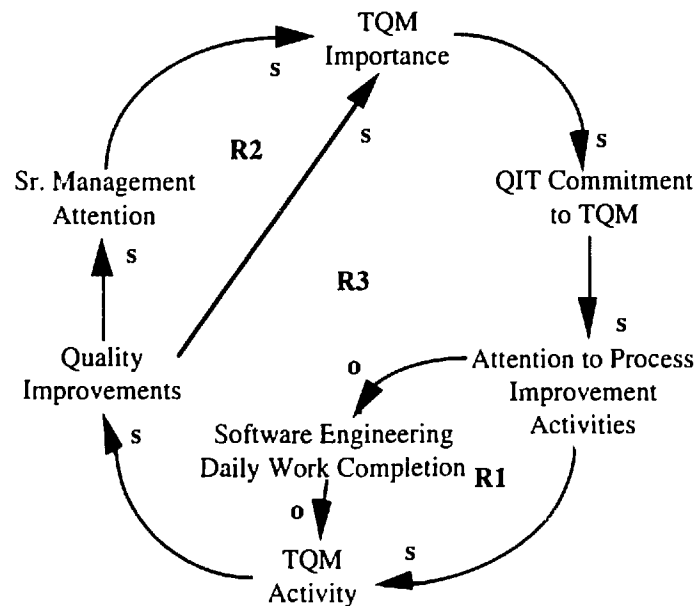
Figure 4.10e

Step 4d: Fine-Tune the Loop Detail

First, the logic flow is traced through the diagram to make sure that there are no interruptions or “leaps” in the logic flow, but that every cause-and-effect relationship represents the logical next step in the development of the story. Next, the level of detail or “abstraction level” is checked to make sure that all elements are telling the story at the same level of detail. For example,

an element such as “costs due to excessive engineering change orders” should not be in the same causal loop as “company business performance,” but instead should be abstracted to “costs of poor quality.”

In terms of abstraction levels, Long-term Business Performance is at a high level relative to Software Engineering Daily Work Completion. In addition, the time frame of this story is too short for such a measure to really play a role. Quality Improvements is at a more appropriate abstraction level and time frame (see Figure 4.10f).

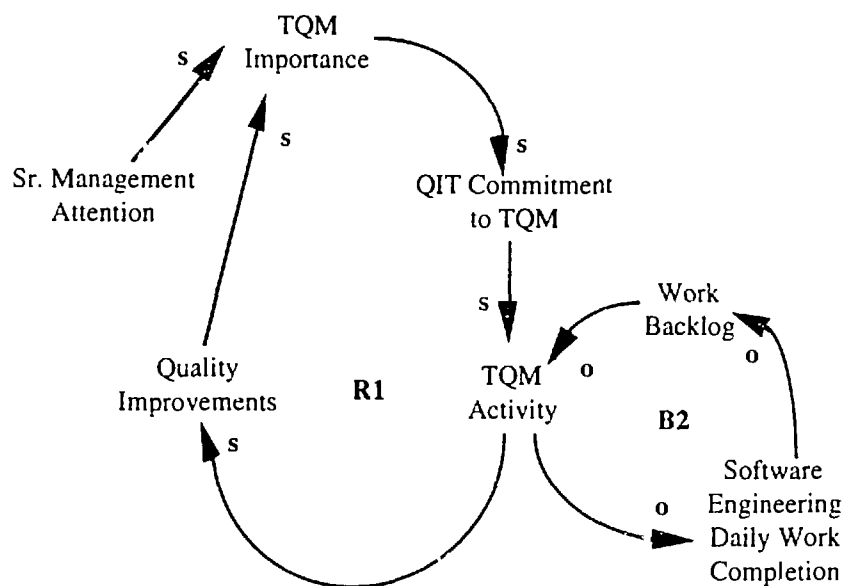


Extended Example: Directive without Commitment

Figure 4.10f

Reviewing the storyline again revealed an inconsistency in the latest diagram. There are three reinforcing loops and no balancing loop. Tracing through the diagram suggests that TQM activities would continue to grow, once management focused their attention on it. The story indicated however that daily pressures distracted the QIT team from staying focused on the TQM work, suggesting that a balancing force is at work.

In Figure 4.10g, Senior Management Attention is pulled out of the loop because in this story once they delegated the task, they really were not part of the story anymore. A new link to Work Backlog is added to represent the pressures that accumulate whenever time is diverted from the daily work completion. This balancing loop captures the dynamics in the story of team members being interrupted and missing meetings when they were pressed by other daily activities.



Extended Example: Directive without Commitment

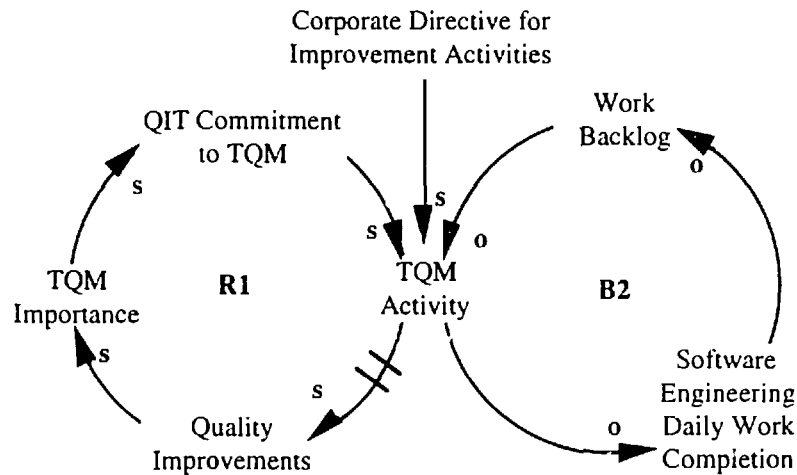
Figure 4.10g

Step 4e: Recheck the Story

If any changes result from fine-tuning the loop detail, then the story is rechecked against the original notes or interview tapes to ensure that the story is still adequately represented by the causal loop diagram.

A final check results in a slight modification. As the managers interviewed in the study had said, senior management did launch a TQM activity by having a QIT team form, so the causal arrow is moved from TQM Importance to TQM Activity. The variable name is changed from Senior

Management Attention to Corporate Directive for Improvement Activities to better represent how the action was perceived at the division level. The final diagram is shown in Figure 4.10h.



Extended Example: Directive without Commitment

Figure 4.10h

4.2.5 Step 5: Identify Potential Archetypes

Up to this point, the process has been primarily an inductive one. Data is gathered and analyzed, out of which concepts emerge. At every step, the work is verified against the available data. In this step, we shift to a deductive approach by using systems archetypes as dynamic scripts. Using the storyline of an archetype as a guide, we try to identify potential archetypes (see "System Archetypes as Dynamic Scripts" in Figure 3.29) that are embedded in the data but not yet constructed into the causal loop diagram, or identify them in the causal loop diagrams developed in steps 1-4 above.

With the knowledge gained from previous steps, a reference mode describing the behavior of symptoms over time should be drawn. The inventory of systems archetypes should be reviewed against the reference

modes to help identify potential fits. More than one candidate archetype can be selected. It is possible that none will appear to fit.

Note: The purpose of this step is not to attempt to “force fit” the data into an archetype. Up to this point, we have allowed the data to do the “talking” and guide the loop construction process. We now want to see if any of the archetypes will provide us with a lens that will help us “mine” more relationships from the data.

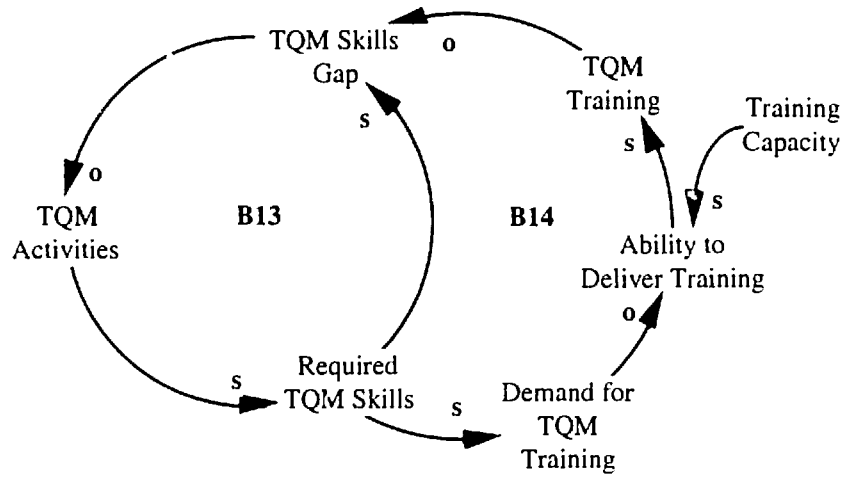
4.2.6 Step 6: Refine Fit of Archetype and Data

The use of systems archetypes at this stage is timely and appropriate. If it is used earlier in the process, it can stifle the “message” of the data by trying to impose a structure on the data before the data is fully understood. After the stories have been identified and all the obvious relationships have been mapped, however, the archetypes can play an important role in guiding *theoretical sampling*. According to Strauss (1987, pp. 38-39) theoretical sampling

is a means whereby the analyst decides on analytic grounds what data to collect next and where to find them...this process of data collection is controlled by the emerging theory.

Here, the archetypes can be used as the basis for theoretical sampling by guiding the data gathering process with a testable dynamic script. An initial assessment of fit is conducted between the CLD's drawn previously and the structure of the candidate archetype. Cycling between working up from the data and down from the archetype is the most productive approach for finding a good fit. The updated diagrams must always be reviewed for consistency with the data. Again, it is possible that none of the archetypes will fit the data.

In Figure 4.11, we see a causal diagram of a Training Capacity story (from Test, Inc. in Chapter 5) where a capacity constraint in training has a negative impact on TQM Activity. According to the story, as TQM Activities increase, the level of TQM skills required will increase as people tackle more challenging projects. This increases the Demand for Training which reduces the Ability to Deliver Training if Training Capacity is not increased to meet the rising demand. As the TQM Training that is delivered decreases, the TQM Skill Gap increases, which will have a downward effect on TQM Activities.



"Training Capacity" Story
(adapted from Brown & Tse, 1992)

Figure 4.11

The presence of a capacity constraint which contributes to a balancing loop alerts us to the possibility of a Limits-to-Success archetype. The training capacity should only become an issue if the demand for training outstrips the supply. Demand would outstrip supply only if there was something driving it up outside of the balancing loops. The Limits-to-Success dynamic script suggests that there should be a reinforcing loop hooked up with this diagram somehow.

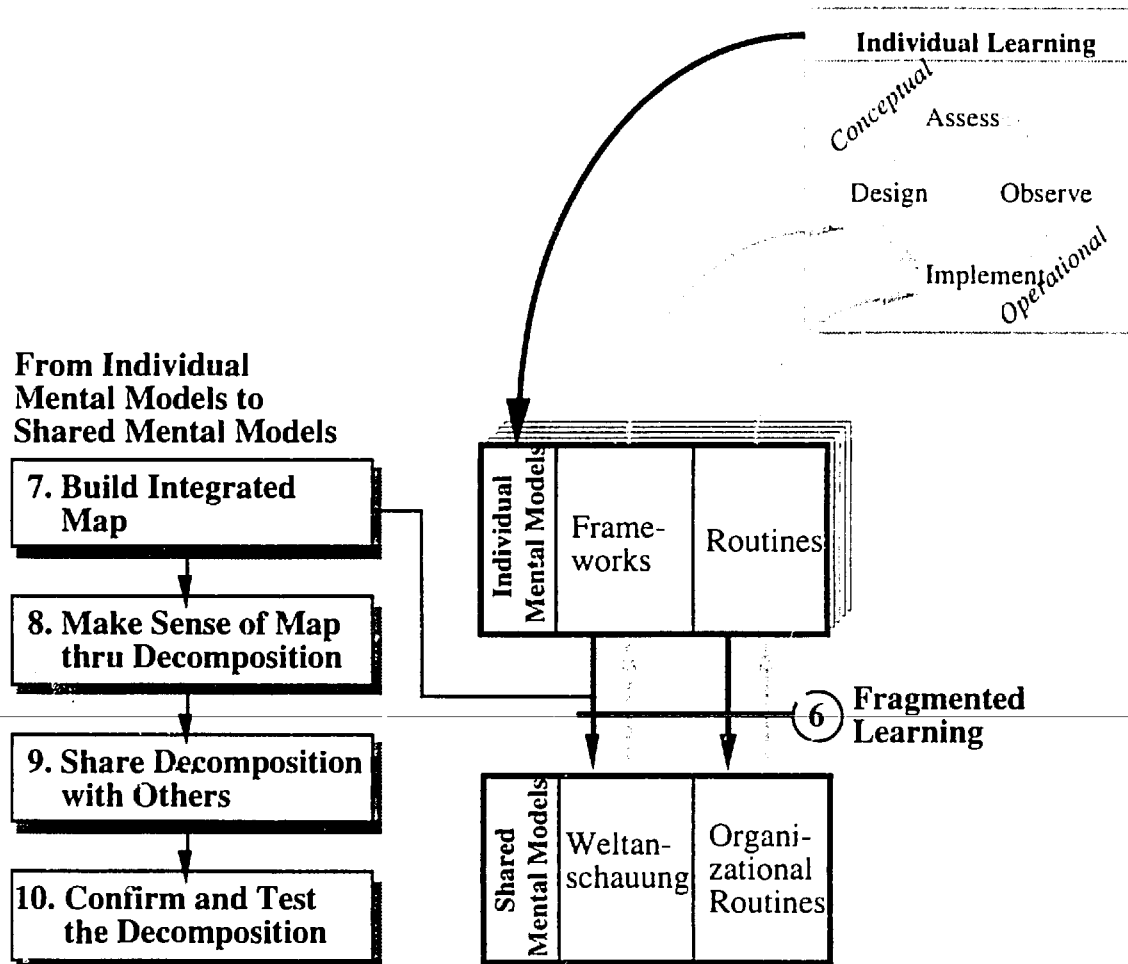
representation. A major distinguishing feature of this methodology relative to many other methods used for conceptualization is the heavy focus on an inductive process at the very front end (steps 1-4). We are not as concerned with obtaining a valid representation of the world "as it really is" as we are in accurately capturing what the individual's *belief* of the world is. In other words, we are trying to inductively build an accurate representation of a person's mental model. This does not mean that the mapping process is simply a recording of someone's regurgitation of what was already known. The mapping process may provide new insights to the individual and alter his or her mental model because of it as will be shown in the product development case in Chapter 6.

4.3 METHODOLOGY FOR TRANSFER OF INDIVIDUAL MENTAL MODELS TO SHARED MENTAL MODELS

As already mentioned in Chapter 3, making individual mental models explicit is necessary but not sufficient for advancing organizational learning. Since perceptions of reality can vary so widely among different people in the same setting, building shared mental models is crucial to organizational learning. Shared mental models are a product of individual mental models and vice versa—they are mutually influential. Thus, individual mental models play a pivotal role not only in individual learning but in organizational learning as well.

Building shared mental models addresses another incomplete learning cycle presented in Chapter 2—*fragmented learning*. To get beyond the fragmented learning of individuals and be able to spread the learning throughout the organization, we need a way of sharing and institutionalizing key insights that are important to a firm's future. The purpose of this section

shared mental models by building on the six steps covered in the previous section. The steps are outlined in Figure 4.13 and a step-by-step description follows.



From Individual Mental Models to Shared Mental Models

Figure 4.13

4.3.1 Step 7: Build Integrated Map

From the individual mental model maps developed above, we begin to develop a shared mental model first through integrating the separate stories to create a common map that contains everyone's stories (see Figures 5.5 to 5.10). In integration, the individual CLD's and archetypes developed in steps 1-6 are combined to tell a complete story. The first part of this step is to

develop a central story that clearly captures the dominant theme that runs through most of the maps (see Figure 5.11). Identifying a central theme is very tied to the main purpose—the *raison d'être*—of the whole mapping effort. In Chapter 5, for example, the central story was around the implementation plan for TQM so the main loop traced out a large reinforcing loop that represented the main purpose of that effort.

Once the main loop has been laid out, the other “minor” themes that had been identified in the previous steps are incorporated into the central story line. Variables are then combined or re-labeled at a higher level of abstraction. Consistency of the final diagram is tested for logic flow, abstraction level, and fit with the available data (see Figure 5.12 for an example of an integrated diagram).

4.3.2 Step 8: Make Sense of Map through Decomposition

A typical integrated diagram developed in Step 7 may have between 10-30 loops in it. Although it is a powerful representation of the current level of understanding of the system, at that level of complexity, most people find it extremely difficult to remember all the individual loops and their impact on the system as a whole. In this step, we try to make sense of the diagram by “decomposing” it into smaller, more manageable chunks.⁷ Again, the systems archetypes are applied here in a deductive process to spot structures in the integrated diagram that appear to fit the storyline of one or more of the archetypes. It is important to point out that the “groundedness” of the

⁷The traditional system dynamics alternative at this point is to proceed with the development of a computer simulation model. We will discuss this point further in section 4.4.

abstracted archetypes rests on the grounded theory building work in the earlier steps.⁸

The reasons for moving up a notch on the Ladder of Abstraction from the grounded diagram is for the purposes of sharing the insights gained from the individual mapping activities. It is unlikely that such a large and complex diagram can serve as a “shared” mental model if it can not be internalized in a coherent and accessible way. The archetypes, once internalized as generic dynamic scripts, can provide the mental model framework for sharing large “chunks” of the whole story.

4.3.2.1 Model Decomposition

The purpose of decomposition (or aggregation) is to reduce the complexity of a model without losing its defining characteristics (Eberlein, 1986; 1989; Richardson, 1986; Simon & Ando, 1961). Although the causal loop diagrams are not “computable” models, some of the methods used for simplifying mathematical models can be used as guides for simplifying CLD’s.

Simon & Ando (1961) address the issue of justifying the use of aggregate variables for economic analysis by providing a method for manipulating matrices into nearly decomposable forms. Under certain conditions, the solutions of the decomposed matrices can represent the original system. Although their purpose seems to be similar, their pure mathematical treatment of the problem makes it difficult to apply their methods to our conceptual models, i.e., the causal loop diagrams.

Eberlein (1989) outlines a theory of simplification that can be used for increasing the understanding of a system dynamics computer model. The

⁸Abstracting up from the “grounded” diagram raises a number of validity issues which will be discussed in section 4.4.

basic idea is to retain a selected subset of the feedback loops that are important in generating the selected behavior of interest. In his approach, “model simplification is based on behavior, with simplified models generating only selected behavior of the full model” (Eberlein, 1989, p. 55). The selection of modes is outlined in Eberlein (1986), but the method requires the manipulation of matrices of equations and is not applicable to our problem at hand. There are, however, some general guidelines that come out of the process that can be applied to the decomposition of CLD’s as well:

1. Identify and list the behavior modes of interest.
2. Seek a structure that generates a behavior mode of interest.
3. Simplify feedback structure by eliminating loops that do not have a significant effect on the behavior mode of interest.
4. Confirm that the behavior of the simplified model is consistent with the behavior mode of interest in the large model

Although we can’t do the kind of behavior testing described in the fourth guideline, we can translate the first three guidelines into a set of steps that can be use to simplify or decompose the integrated diagram created in step 7 into smaller chunks that are more comprehensible. We will address the behavior confirmation issue in section 4.4.2 where we discuss the role of computer models.

4.3.2.2 Causal Loop Diagram Map Decomposition

The sub-steps of the process of decomposing the causal loop diagram are listed below, followed by an example that revisits the “Training Capacity” story in section 4.2.6:

Step 8a: Identify a behavior mode of interest.

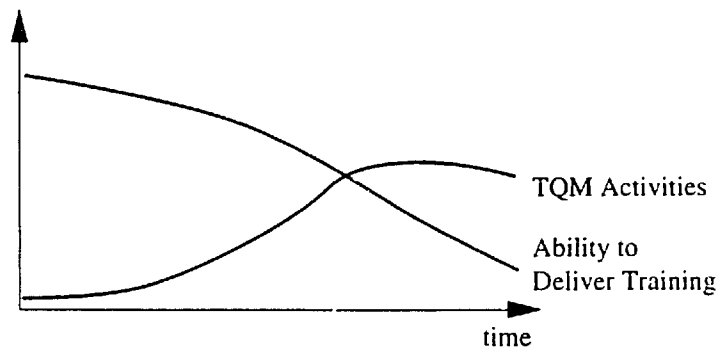
Step 8b: Find relevant loops in the integrated diagram

Step 8c: Simplify loops by collapsing multiple links into a single link

Step 8d: Check that loop polarity remains the same

Step 8a: Identify a behavior mode of interest.

We begin the decomposition process by identifying a behavior mode of interest based on direct observation or on reasoning based on prior theory (e.g., systems archetypes). In the training capacity story, there were two behaviors that stood out: the time behavior of TQM Activities and the Ability to Deliver Training. As TQM Activities increased, the Ability to Deliver Training decreased. The behavior modes of interest are shown in Figure 4.14.



Behavior Modes of Interest

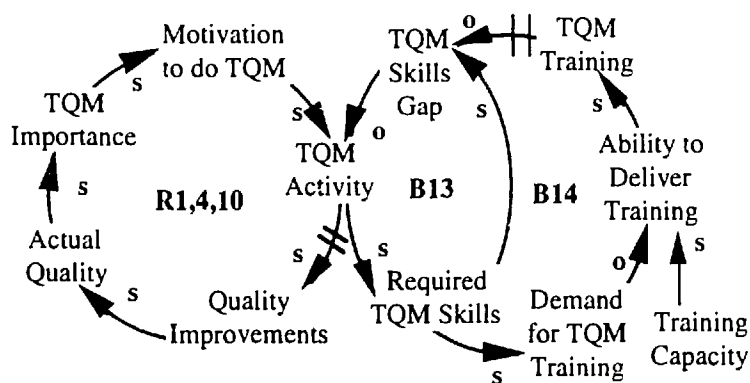
Figure 4.14

The archetype dynamic scripts can also serve as a guide for identifying behavior modes of interest. In particular, they can be used as a point of reference to see if the patterns of behavior suggested by an archetype, like Limits-to-Success, can help improve one's understanding of a behavior mode in the case. A generic behavior generally found in the Limits-to-Success archetype is the initial growth of a variable, followed by a leveling off or collapse of that variable due to a constraining factor. The behavior mode described in Figure 4.14 seems to follow this general pattern, as TQM activities

grow for a while, and then level off as the ability to deliver training decreases. In this case, that behavior pattern was already identified from the case history. But if it hadn't been already identified, the researcher could investigate whether such behavior was present, basing her reasoning on the Limits-to-Success archetype.

A second use of systems archetypes is to identify "latent" patterns of behavior. In this case, the systems archetypes may be used to posit patterns of potentially important behavior that have not *yet* been observed. For example, the Tragedy-of-the-Commons (ToC) archetype predicts an undesirable global outcome in the future while desirable local outcomes are being achieved in the present. When an organization recognizes that it is caught in a ToC archetypal situation, actions can be taken in advance to *prevent* the predicted dynamics from occurring (for an example, see the ToC case in Chapter 6).

Step 8b: Find relevant loops in the integrated diagram



"Training Capacity" Limits to Success—Revisited
(adapted from Brown & Tse, 1992)

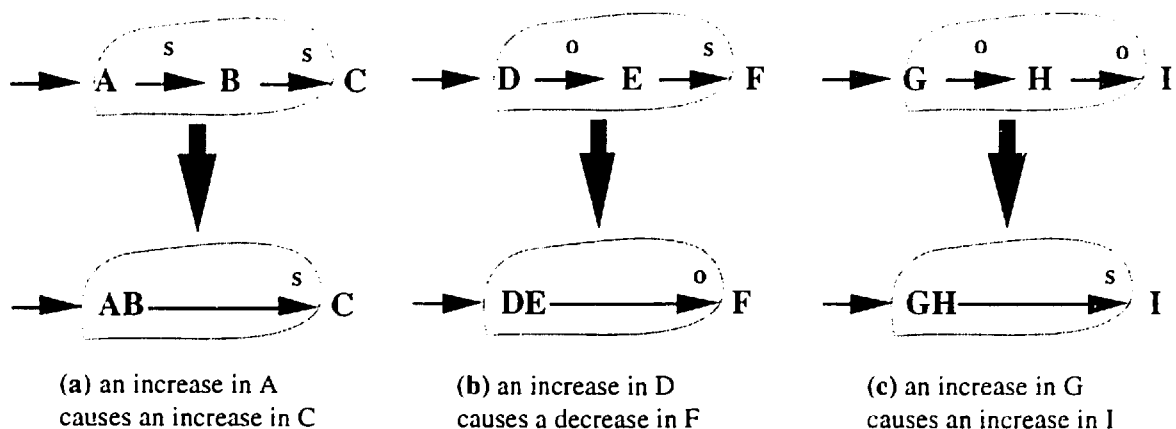
Figure 4.15

The relevant loops for the training capacity story were identified in Figure 4.12 and are redrawn here as Figure 4.15. As explained above, the diagram has been abstracted from the integrated diagram presented in Figure 5.12.

Step 8c: Simplify loops by collapsing multiple links into a single link

Although the CLD in Figure 4.15 only has three feedback loops, it contains 11 variables which add to its complexity. This simplification step can help reduce the number of variables, which in turn, reduces the number of links. This process can also reduce the number of loops, although this is not always the case. The archetypes may be useful for identifying the central dynamic that is of interest and to suggest a CLD structure that the simplified diagram can represent (although using an archetype is not necessary).

In reducing the number of variables and links in a diagram, the replacement variable needs to preserve the essential *meaning* of the individual variables, and the new link needs to preserve the *polarity* of the directed arc between the old set of variables and the variable to which it is causally linked. Figure 4.16 shows the three different combinations which

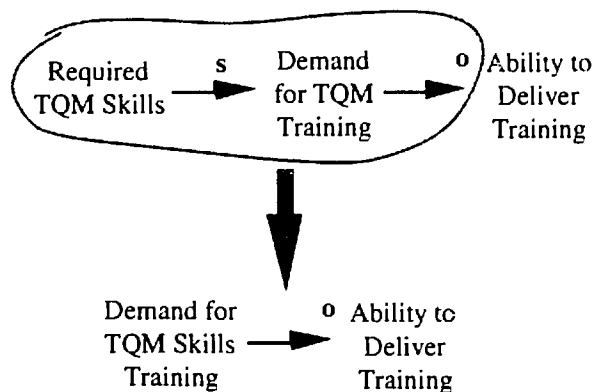


Preserving Directed Arc Polarity

Figure 4.16

will be encountered and what the collapsed link should look like. Notice that we start with a variable (e.g., A) and go “downstream” in the direction of the causal arrow and include all arcs and variables (e.g., $A \xrightarrow{s} B \xrightarrow{s} C$) up to but *not* including the “effect” variable (e.g., C) to which the collapsed variable will be linked. The link between the new variable (e.g., AB) and the “effect” variable (e.g., C) should be labeled with an “s” or an “o” to preserve the original causal effect on the final effect variable (e.g., $AB \xrightarrow{s} C$).

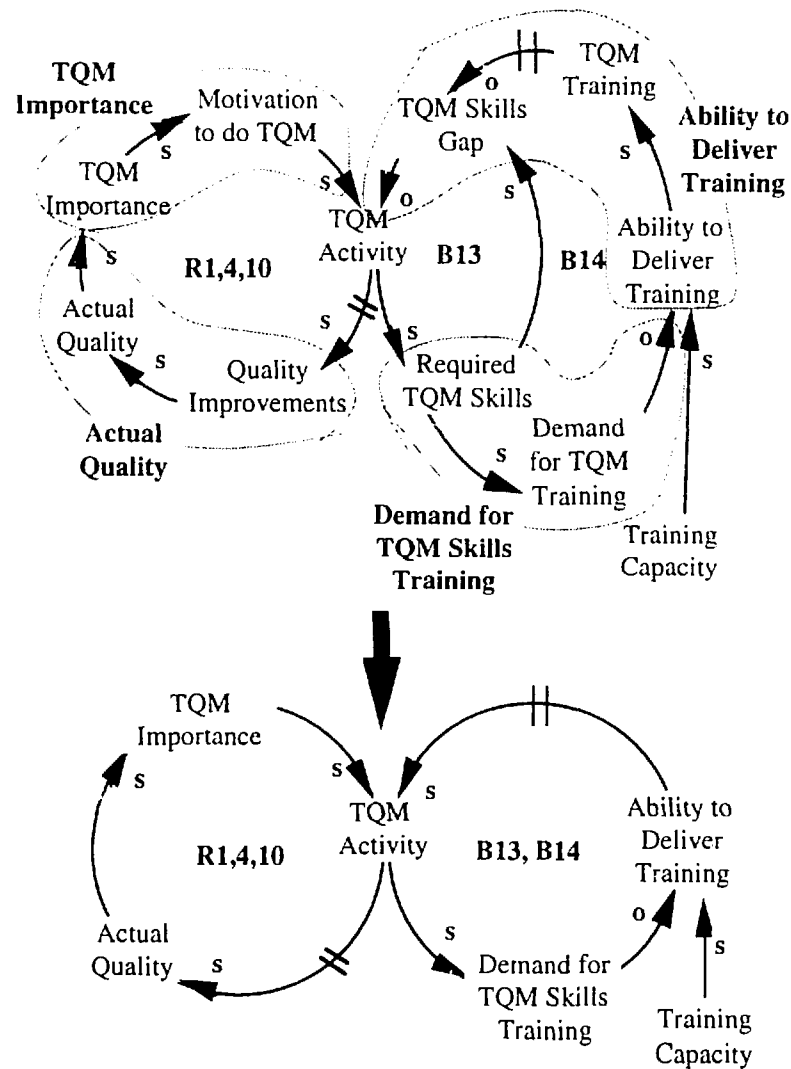
The three examples in Figure 4.16 show the three different combinations one will encounter in collapsing links. We’ve already covered case (a) where the links being collapse are all “s’s.” In case (b), one of the links is an “s” and one is an “o” which means that as D *increases*, E *decreases*, and F *decreases*. The collapsed version preserves the “net” polarity by signing the new arc with an “o” so that an *increase* in DE causes a *decrease* in F. In the case of (c), where an increase in G causes a decrease in H which causes an increase in I, we see that the net effect of an *increase* in G is an *increase* in I. The collapsed variable, GH, is linked with I and is signed with an “s” so that an *increase* in GH causes an *increase* in I.



Collapsing Links Example

Figure 4.17

A simple example of collapsing links is shown in Figure 4.17. We take a piece of a loop contained in Figure 4.15, and collapse two links into one. The two variables, Required TQM Skills and Demand for TQM Training are combined into a single variable, Demand for TQM Skills Training, because the TQM skill requirement is not as essential to the story as the demand that it generates for the training. The net effect to Ability to Deliver Training is preserved by signing the arc with an “o.” The entire simplified diagram is shown in Figure 4.18.



Simplified "Training Capacity" Story

Figure 4.18

In principle, the mechanics of this procedure can be generalized to collapse any number of variables and links into one variable and an arc whose polarity would be determined by the number of "o's" that had been collapsed. An odd number of "o's" would indicate signing the arc with an "o" while an even (or zero) number of "o's" would indicate signing the arc with an "s." In practice, however, the number of variables that can be collapsed together will be constrained by the first requirement of preserving the essential meaning of all the individual variables that are being combined.

Step 8d: Check that loop polarity remains the same

The final step is to double-check that the loop polarity has not been changed by the simplification process, i.e., balancing loops should still be balancing loops and reinforcing loops should still be reinforcing loops.

4.3.3 Step 9: Share Decomposition with Others

The diagrams are continually verified against the available data in the modeling process—checking for consistency between the storyline and the constructed CLD's. The diagrams should also be verified with *all* the people who were interviewed for "face validity"—that is, can they look at the diagram and agree that it reflects the dynamics of what they believe happened. If the individuals who had been interviewed as part of the data gathering process understand the decomposed diagrams and agree with the accuracy of the links and loops, it builds confidence that the diagram has preserved the essential structure that explains the behavior mode of interest.

4.3.4 Step 10: Confirm and Test the Decomposition

Once the relevant archetypes have been identified and agreed upon by the initial group of interviewees, they should be circulated to other relevant

players in the organization who had not been involved in the study. The validation at this stage consists of two parts. One, is whether there is a general agreement that the decomposition accurately describes what has happened. Second, is whether there is a general agreement on the prescribed actions suggested by the archetype. The results to both questions can be tallied and shared. How widely the mental models are shared can be measured by assessing to what extent there is general agreement about both the *framework* that the archetype provides for capturing the experience and the *routines* that the archetypes prescribe for addressing similar situations in the future.

There is one important caveat, however. This analysis does *not* imply that the prescribed actions should necessarily be taken. The CLD's and systems archetypes abstracted from the integrated diagram represent elements of a larger theory (i.e. that embodied by the integrated diagram). Individual prescriptive actions based upon individual sets of loops taken out of the whole diagram may conflict with one another, or interact in ways that do not lead to the overall desired behavior. In decomposing a complex computer model, individual insights can be checked against the larger model's behavior. This is not possible in decomposing a complex CLD. Therefore, we strongly recommend that (1) researchers and managers be extremely cautious in taking actions without considering such interaction effects and (2) computer simulation models be used to further test the policy implications of taking such actions.

4.3.5 Summary of Steps 7-10

The main purpose of steps 7-10 is to provide a way for a group of individuals to see their individual experiences mapped into a common map and see the richness of the inter-relationships among all the variables. The integrated

diagram gives them a “systemic view” by graphically showing how the individual stories and maps fit together as part of a larger system. On the other hand, the large size of the integrated diagram makes it very difficult for people to walk away from the diagram and remember all of it in a coherent way. That is, people may appreciate the richness of the connections, but the mental model representation they walk away with is akin to a bowl of spaghetti.

The decomposition process is meant to help simplify the diagram into manageable “chunks” while preserving all of the important structures that explain the behavior of interest. The process of sharing and testing the decompositions with others in the organization can help build a shared mental model of the structure and the behavior mode it is meant to represent.

It is important to note that the methodology proposed above is meant to accurately represent what people believe to be true based on their current understanding of the situation. It is a structured process for mapping people’s mental models into an explicit and sharable form. Although new insights may be discovered by the participants through the mapping process, the methodology does not (nor is it meant to) verify that the mapping is an accurate representation of reality. This kind of testing process is beyond the capability of simple pen and paper tools like CLD’s and systems archetypes; it requires the use of a computer model. This topic will be covered in the following section.

4.4 DISCUSSION

In any modeling effort, whether it be computer models or conceptual maps, the issue of validity needs to be addressed. In this section we will take a

critical look at the methodology outlined above and assess its level of validity. We will also look at its strengths and weaknesses relative to its stated purpose and compare them with the pros and cons of the traditional system dynamics approach of building computer models. We conclude with a brief discussion on possible future work to develop the methodology further.

4.4.1 Validity

In scientific research, there are three types of validity one needs to address: construct, internal, and external (Kidder & Judd, 1986). While these validity criteria were established for evaluating experimental research and are not fully achievable in “real world” research settings, they serve as a useful benchmark against which one can measure strengths and weaknesses of one’s research. It is with this intent that we will apply the validity criteria to the mental model mapping methodology described above. A fourth kind of validity is also relevant for the kind of work described in this thesis—face validity. We begin by establishing a brief definition of each type of validity criterion.

Construct validity is the degree to which the variables that have been defined accurately reflect or measure the construct of interest. That is, it tries to address the question:

To what extent are the constructs of theoretical interest successfully operationalized in the research? (Kidder & Judd, 1986, p. 28)

Having high construct validity means that all the constructs that the research intended to study have been successfully represented by the specific variables the researcher has selected. For example, if one were interested in studying the effect of poverty on educational achievement, one needs to operationalize the concept of poverty into a measurable variable, such as “income level.”

Internal validity is concerned with the extent to which one can draw conclusions about the causal effects of one variable on another. Internal validity means addressing the question:

To what extent does the research design permit us to reach causal conclusions about the effect of the independent variable on the dependent variable? (Kidder & Judd, 1986, p. 28)

Internal validity comes closest to the everyday use of the term validity, i.e., “does theory match reality?” If we have high internal validity, it means that we are more able to argue that the relationships we have identified are causal (not correlational) ones. We can have high construct validity in both the independent and the hypothesized dependent variable, but have low internal validity in the causal link between the two. For example, “income level” may be a very good indicator of poverty and “years in school” may be a good measure of educational attainment, but if internal validity is low, we can not say much about the causal connection between the two. Threats to internal validity come in the form of rival hypotheses which the experimental design has not addressed.

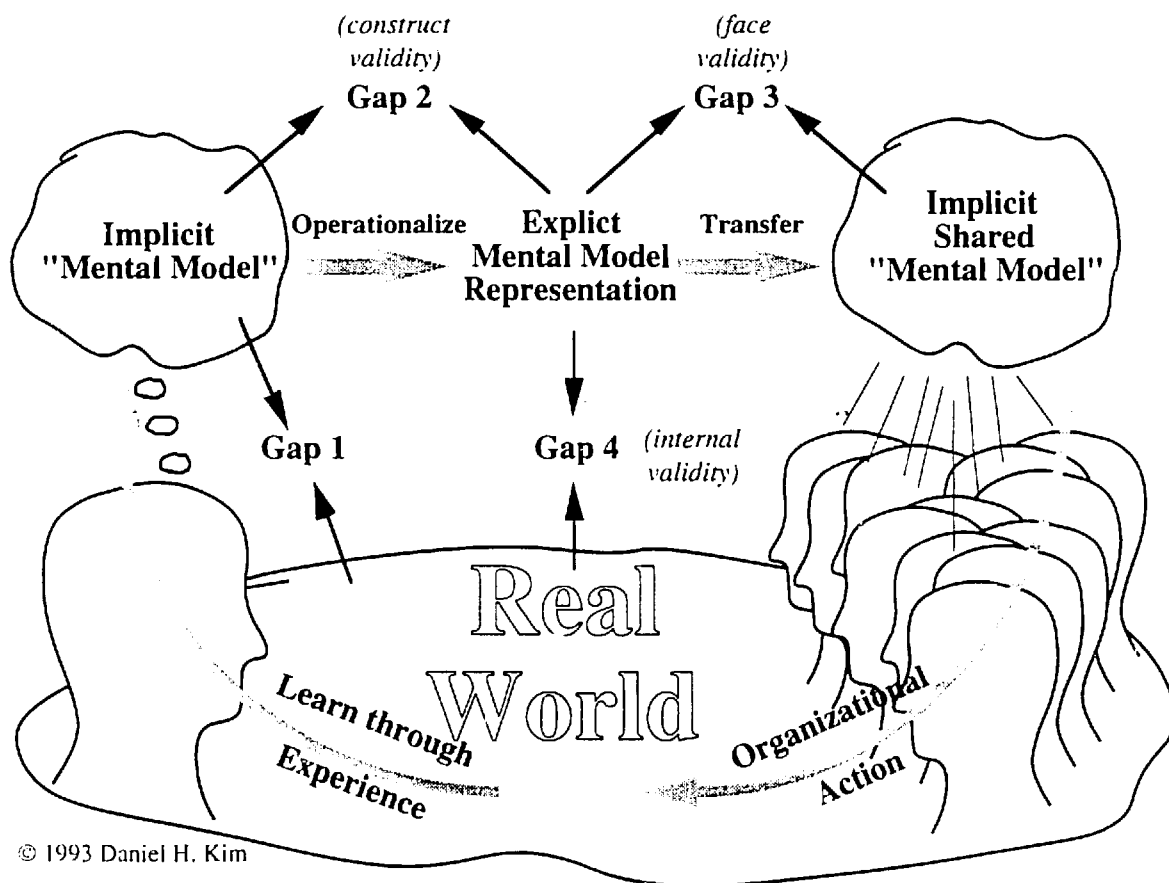
External validity is concerned with the generalizability of the research results to other similar settings of interest. Addressing external validity means answering the question:

To what extent can we generalize from the research sample and setting to the populations and settings specified in the research hypothesis? (Kidder & Judd, 1986, p. 28)

Suppose, for example, that we had high construct validity in our measures of poverty and educational achievement and that we could claim a reasonable causal connection between the two (high internal validity). How strong of a claim we can make that the results of the research are applicable beyond the study itself is dependent on the strength of the study’s external validity.

Face validity is evaluated by a group of experts or judges who assess whether a measuring technique measures what it claims to measure (Kidder & Judd, 1986). Although every instrument we use must pass a face validity test, it is inherently a subjective process. One way to quantify a measure of face validity would be to compute the amount of agreement on a measure by a panel of judges

We return to a picture we first presented in Chapter 3 to help clarify the discussion on the validity issue as well as on the purpose of the methodology outlined above. The revised picture is shown in Figure 4.19 with four distinct gaps identified in the organizational learning cycle.



Assessing the Gaps in Organizational Learning

Figure 4.19

The first gap is the one between an individual's implicit mental model and the real world (Gap 1) which is a measure of how well-grounded the individual's mental model is. That gap can never be addressed directly by an outsider because mental models *are* implicit until they are explicated through some formal representation. It is a matter of philosophical debate whether the individual herself can ever address that gap directly. In either case, this dissertation is not about addressing Gap 1 (thank God!).

The second gap is between what the individual has implicitly in her head and the explicit representation of it (Gap 2). The purpose of steps 1-6 in our methodology is to operationalize an implicit individual mental model into an explicit representation of that mental model while minimizing the size of Gap 2. The grounded theory building process is designed to ensure high *construct* validity. That is, the variable development process is intended to create a high level of confidence that the concepts represented by the constructed variables accurately captures the gathered data. The lower the size of Gap 2 is, the higher the construct validity, that is, our representation accurately "measures" the implicit mental model.

The third gap is between the explicit representation of the individual mental model and the "sharedness" of the implicit mental model of others in the organization (Gap 3). The variables identified in the decomposed diagram has less *construct* validity than the ones at the end of steps 1-6 because by collapsing two or more variables into one, we are reducing the likelihood of the new variable accurately representing the original, more grounded variables. At this point, however, we are more interested in *face* validity because the purpose of these steps is to build a shared understanding of the dynamic behavior and structures of interest. Thus, circulating the decomposed maps and getting a high degree of agreement among the people

surveyed increases face validity. To the extent that others in the organization who have shared experience in the context we are studying believe there is a good match between their implicit mental model and the explicit representation, we have high face validity. The main purpose of steps 7-10 is to help reduce the size of Gap 3.

The fourth gap is between the explicit representation of the mental model and the “real world” of experience in which the organizational members share (Gap 4). The more evidence we can provide that the causal relationships captured in the mental model representation is in fact what exists in the real world, we will be building *internal* validity. The size of Gap 4 depends on the size of Gap 1 and Gap 2. The methodology outlined in this chapter, however, is not intended to explicitly address either Gap 1 or Gap 4.

One way to increase internal validity would be to expand the data gathering to include more people and more sources of data. The causal connections that are constructed from the data could then be tested against other available data. The development of a system dynamics simulation model can also strengthen internal validity by testing for internal inconsistencies in the mapped relationships—a possibility we will discuss in the following section.

Since the focus of the methodology in steps 1-6 is on mapping individual understanding of a specific situation in which she was involved, the issue of *external* validity is not really relevant. However, in a broader context of organizational learning—that is, when shared mental models developed by a given group are eventually used to guide action by other groups—the concept of external validity will become relevant. Studies of external validity, while outside the scope of this dissertation, will be important future research as this methodology gets put into broader practice.

4.4.2 Role of System Dynamics Computer Models

The methodology described above is strong on construct validity and face validity, but weak on internal validity. The use of systems maps is not intended as a replacement for a computer model, but as a necessary and important step to help managers clarify their level of understanding of dynamically complex issues and provide a way to communicate it to others. Although there will be some cases where insight generated by the clarity of the causal maps can be translated directly into action (the product development team in Chapter 6, for example), there are many more instances where the diagrams alone will be inadequate. This is where the role of computer models become important. The model can be all the more powerful when the organizational members themselves conclude that the use of a computer model is necessary once they have gained clarity around their issue.

In Figure 4.20, we have placed a fork in the process after step 7. The traditional system dynamics approach is to proceed from the conceptual map of step 7 to building a computer model. Policy analyses are conducted through multiple simulation runs and the results are shared with others through various means, such as CLD's, management flight simulators, simulation runs, and learning labs. Policies are implemented and their impact are assessed.

Which way should one go when one reaches the fork? There is no absolute right choice for all settings. The choice depends on the purpose and circumstance of each individual case. As the above discussion showed, steps 8-10 can provide a high level of face validity for sharing a map that is of importance to the organization. The decomposition steps help to make the maps more accessible to more people, thereby increasing the size of the shared

base. The maps, however, are weak on internal validity and can lead to actions that are not based on a sound representation of the actual causal connections.

The computer models can clarify and add richness to the explicit representation of the causal connections of a complex system. By simulating and testing, one can help improve an individual's understanding of one's own mental model representations as well as the long term consequences of actions taken based on that understanding. However, the level of skill and the investment of time required to construct a good computer model is much more significant than what is required to create causal maps. Even though the additional payoff may be proportionately greater, it can be difficult for those who have not experienced the value of simulation models to see value in it. This is especially true of people who believe that they don't have a Gap 1—they have a lock on reality. It is a classic catch-22. Once people see the value, they may be glad to invest in it. But unless, they invest in it, they will never see the value.

The systems mapping work, because of the minimal investment it requires upfront, can provide a glimpse into the value of going further into more rigorous work with computers. The main point is that the choice does not have to be an either/or. As Figure 4.20 shows, the methodology proposed

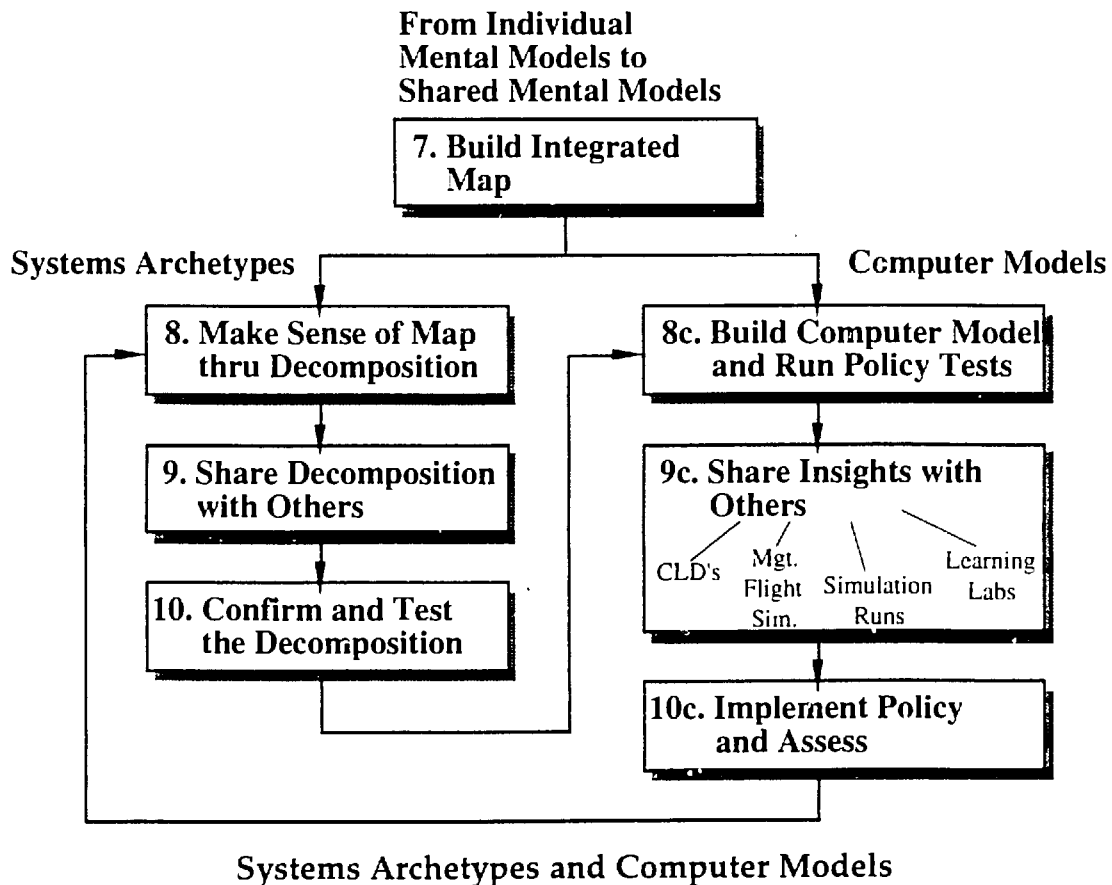


Figure 4.20

in this chapter is more like a detour than a substitution. People may end up going from sharing the decomposed diagrams, and based on that understanding go right to getting engaged in the process of building a computer model (with a professional modeler's help). One could also cycle through the computer modeling process and then go to Step 8 and follow the decomposition route in step 8 above for clarifying and sharing the learning from that experience.

4.4.3 A Final Note on Validity

Does this mean that the use of the computer model suddenly brings more validity to the party? The answer is no. The same validity issues apply equally to the whole computer modeling process. The question of validity must always be addressed relative to a stated purpose.

	Model-Building Operator's View	Model-Building Observer's View
Definition of Validity (<i>Webster's Third New International Dictionary</i>)	"well-rounded or justifiable: applicable to the matter at hand: pertinent, sound. Able to affect or accomplish what is designed or intended: effective, efficacious."	"validity of an inference: correctly derived from its premises; specifically: true in terms of the logical principles of the logistic system to which the inference belongs."
Definition of Model	Theory of relationships	Computational Result
Proof of Validity	Degree of confidence in assumptions; no absolute proof	A defined concept; certain statistical tests are met.
View of Model	A tool for decision-making	Collector's item; organizes and rearranges data
Use of Model	Seeks shared confidence	Seeks competitive debate

Operator vs. Observer View of Models and Validity

(adapted from Forrester, (1973))

Figure 4.21

Forrester (1973) provides an interesting comparison between two views on models and their validity: an operator view and a model-building observer's view (see Figure 4.21). Operators are people who make decisions to control action (e.g., manager). Observers, on the other hand, explain and criticize, but they do not act (e.g., staff advisor).

Their views of models and of validity, as expressed by Forrester, are very different. The operator's view is pragmatic and "use-oriented." The observer's view is philosophical and "theory-oriented." The methodology in this chapter was developed with a particular eye on the operator's view and the belief that operators need to better articulate the implicit theories guiding their actions. Although, Forrester was referring to computer models in his paper, the points apply equally well to any kind of model—including pen-and-paper models.

Chapter 4 References

- Argyris, Chris (1990). *Overcoming Organizational Defenses*. Wellesley, MA: Allyn-Bacon.
- Argyris, Chris, Putnam, R., & Smith, D.M. (1985). *Action Science*. San Francisco: Jossey-Bass.
- Balz, William F., & Garberding, Scott R. (1993) *A System Dynamics Analysis of Total Quality Management Implementation Issues*. Unpublished Master's thesis, MIT Sloan School of Management.
- Brown, Jim , & Tse, Scott S. F. (1992). *A System Dynamics Analysis of Total Quality Management Implementation False Starts*. Cambridge, MA: MIT Sloan School of Management.
- Burchill, Gary W. (1993). *Concept Engineering: An investigation of TIME vs. MARKET orientation in product concept development*. Unpublished dissertation, MIT Sloan School of Management.
- Cohen, Morris R., & Nagel, Ernest (1936). *An Introduction to Logic and the Scientific Method*. New York: Harcourt, Brace & Co.
- Crosby, Philip B. (1979). *Quality is Free: The Art of Making Quality Certain*. New York: McGraw-Hill.
- Dewey, John (1926). *Experience and Nature*. Chicago: Open Court Publishing.
- Eberlein, Robert L. (1986). Identifying an Displaying Important Feedback Paths. In *International Conference of the System Dynamics Society*, (pp. 159-177). Sevilla: System Dynamics Society.
- Eberlein, Robert L. (1989). Simplification and Understanding of Models. *System Dynamics Review*, 5(1, Winter), 51-68.
- Ehrler, Steven M., & Jansen, Eckart W. (1993) *System Dynamics as an Organizational Diagnostic for Total Quality Management Implementation*. Unpublished Master's thesis, MIT Sloan School of Management.
- Forrester, Jay W. (1961). *Industrial Dynamics*. Cambridge, MA: Productivity Press.
- Forrester, Jay W. (1973). Confidence in Models of Social Behavior--with Emphasis on System Dynamics Models. *MIT Sloan School of Management unpublished working paper*.
- Forrester, Jay W. (1979). *Information Sources for Modeling the National Economy*. Cambridge, MA: System Dynamics Group, MIT Sloan School of Management.
- Giancola, Paula (1992). *A Systems Thinking Perspective of the Product Development Process: A Study of Business and Academic Collaboration*. Cambridge, MA: MIT Sloan School of Management.
- Glaser, B.G. (1978). *Theoretical Sensitivity: Advances in the Methodology of Grounded Theory*. Mill Valley, CA: Sociology Press.
- Glaser, B.G., & Strauss, A.L. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Chicago: Aldine.
- Hempel, Carl Gustav (1965). *Aspects of Scientific Explanation, and other essays in the philosophy of science*. New York: The Free Press.
- Isaacs, William (1992). The Ladder of Inference. In Cambridge, MA: MIT Organizational Learning Center.
- Kendrick, John J. (1992). Companies Continue to Embrace Quality Programs, but Has Total Quality Generated More Enthusiasm than Results? *Quality*, 31(5 (May)), 13.
- Kidder, Louise H., & Judd, Charles M. (1986). *Research Methods in Social Relations* (Fifth ed.). New York: Holt, Rinehart and Winston.

- Kim, Daniel H., & Burchill, Gary (1992). Systems Archetypes as a Diagnostic Tool. In *Proceedings of the 1992 International System Dynamics Conference*. Utrecht, Netherlands: System Dynamics Society.
- Kuhn, Thomas (1962). *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press.
- Lewin, Kurt. (1951). *Field Theory in Social Science*. New York: Harpers & Brothers.
- Mizuno, Shigeru (1988). *Management for Quality Improvement: The 7 New QC Tools*. Cambridge, MA: Productivity Press.
- Peterson, John T., & Tok How, Lim (1993) *A System Dynamics Analysis of Total Quality Management Implementation False Starts*. Unpublished Master's Thesis, MIT Sloan School of Management.
- Randers, Jorgen (Ed.) (1980). *Elements of the System Dynamics Method*. Cambridge, MA: MIT Press.
- Richardson, George (1981). *Problems with Causal-Loop Diagrams*. Cambridge, MA: System Dynamics Group, MIT Sloan School of Management.
- Richardson, George, Andersen, David F., Rohrbaugh, John, & Steinhurst, William (1992). Group Model Building. In *Proceedings of the 1992 International System Dynamics Conference*. Utrecht, The Netherlands: System Dynamics Society.
- Richardson, George, & Senge, Peter (1989). Corporate and Statewide Perspectives on the Liability Insurance Crisis. In P. Milling & E. Zahn (Eds.), *Computer-Based Management of Complex Systems: Proceedings of the 1989 International Conference of the System Dynamics Society*. Berlin: Springer-Verlag.
- Richardson, George P. (1986). Dominant Structure. *System Dynamics Review*, 2(1, Winter), 68-75.
- Richardson, George P., & Pugh, Alexander L. (1981). *Introduction to System Dynamics Modeling with DYNAMO*. Cambridge, MA: MIT Press.
- Roberts, Roger (1992). *Implementing the Learning Organization: The Application of Systems Thinking Tools to Product Development in a Manufacturing Firm*. Cambridge, MA: MIT Sloan School of Management.
- Schein, Edgar (1987a). *The Clinical Perspective in Fieldwork*. Newbury Park, CA: Sage Publications.
- Schein, Edgar H. (1987b). *Organizational Culture and Leadership*. San Francisco: Jossey-Bass.
- Schein, Edgar H. (1987c). *Process Consultation Volume II: Lessons for Managers and Consultants*. Reading, MA: Addison-Wesley.
- Schein, Edgar H. (1988). *Process Consultation Volume I: Its Role in Organization Development*. Reading, MA: Addison-Wesley.
- Senge, Peter M. (1990b). *The Fifth Discipline*. New York: Doubleday.
- Shiba, Shoji (1992). *Four Years of Management Thinking: Planning & Implementation*. Cambridge, MA: Productivity Press.
- Shiba, Shoji, & Graham, Alan (1993). Total Quality Management as a Learning System. *Center for Quality Management Journal*, 2(1), 48-56.
- Simon, Herbert A., & Ando, Albert (1961). Aggregation of Variables in Dynamic Systems. *Econometrica*, 29(2), 111-138.
- Stone, Eugene F. (1978). *Research Methods in Organizational Behavior*. Glenview, IL: Scott, Foresman and Company.
- Strauss, Anselm L. (1987). *Qualitative Analysis for Social Scientist*. Cambridge: Cambridge University Press.

- Thornton, P.B. (1992). Teamwork: Focus, frame, facilitate. *Managerial Review*, 81(November), 46-7.
- Vennix, Jac A. M. (1990). *Mental Models and Computer Models: Design and Evaluation of a Computer-based Learning Environment for Policy-making*. Beuningen, Netherlands: University of Nijmegen.
- Vennix, Jac A.M., Gubbels, Jan W., Post, Doeke, & Poppen, Henk J. (1990). A Structured Approach to Knowledge Elicitation in Conceptual Model Building. *System Dynamics Review*, 6(2, Summer), 194-208.
- Womack, James P., Jones, Daniel T., & Roos, Daniel (1990). *The Machine that Changed the World*. New York: Rawson Associates.

Chapter 5

Mapping Mental Models: TQM Implementation False-Starts[†]

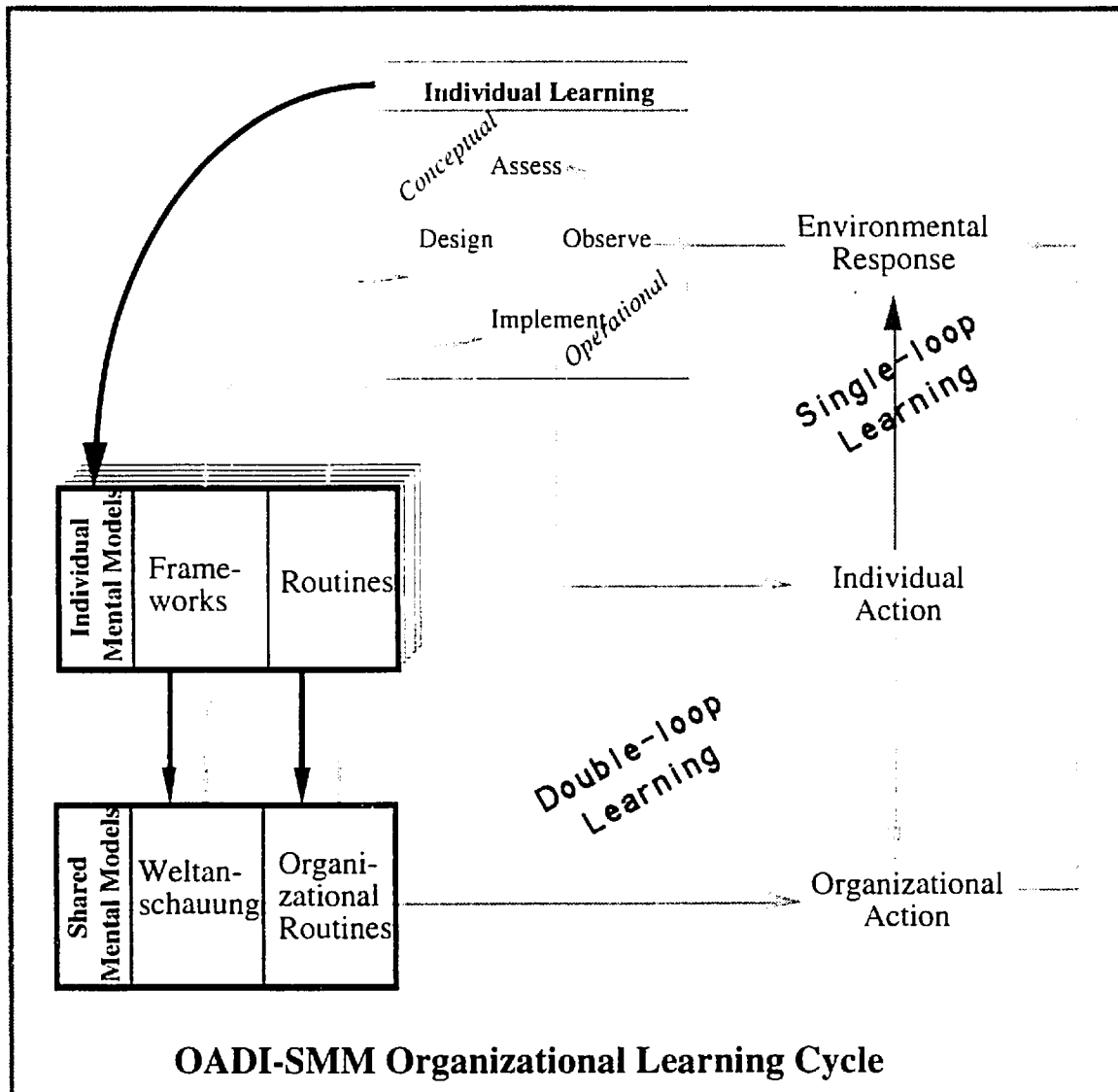
"The world is ruined by best efforts."

—Dr. W. Edwards Deming

5.0 INTRODUCTION

In Chapter 2, the role of mental models in organizational learning was made explicit and highlighted as a critical link. In Chapter 3, we looked more deeply into the nature of mental models and reviewed various methods used for mapping them. Chapter 4 presented a research process and methodology for making mental models explicit and for moving from individual to shared mental models. In this chapter, I present two case studies where the methodology is used to make systemic sense of TQM implementation efforts that proved unsuccessful. In terms of the organizational learning model developed in Chapter 2, the focus of these two cases is on the link between individual learning and individual mental models and the link between individual and shared mental models (see Figure 5.1).

[†] The work described in this chapter was supported by the MIT Organizational Learning Center. The case studies covered in this chapter are based on joint work with Gary Burchill and fieldwork presented in the MIT Sloan School of Management master's thesis (Brown & Tse, 1992) which the author and Burchill helped supervise.



Mapping Experience into Explicit Mental Models

Figure 5.1

This chapter begins with a brief history of total quality management and its implementation in the U.S., paying particular attention to the emerging problems organizations are experiencing in this country with TQM implementation and the need for a systemic approach.

TQM implementation false starts are studied in two companies. Focusing on each company, one at a time, we present the "stories" told by three interviews through causal loop systems maps. The stories are then integrated

into a single diagram which graphically shows the inter-relationships among all the stories. The integrated diagram is then “decomposed” into several systems archetypes, each of which provides a clearer way of seeing the systemic structure responsible for the company’s experience. The presentation of the first company case, Test, Inc., includes systems maps of the individual stories, an integrated map, and the systems archetypes. The second company presentation, Wafers, Inc., includes the integrated diagram only and the systems archetypes. In the final section, I present a summary of the findings on TQM implementation patterns, and offer some possible next steps.

5.1 TOTAL QUALITY MANAGEMENT

Total Quality Management¹ (TQM) embodies a management approach that is committed to satisfying customers by designing, producing, and delivering high quality products. TQM attempts to involve all employees to continuously improve the company. TQM is both an all-encompassing philosophy about managing a business and a set of specific statistical tools applied to specific problems. It is this blend of the micro and the macro which makes it such a potent discipline. Either element by itself would not be revolutionary. Without the philosophy, TQM is reduced to a bag of tools which is applied to problems only as they arise; in effect, simply helping to fight fires. Without the statistical tools, however, TQM is nothing more than a guiding light to a goal that offers no help for navigating the terrain. TQM’s

¹There are many different terms in common use at different companies, such as QIP (Quality Improvement Process), TQ (Total Quality), EI (Employee Involvement), CWQC (Company Wide Quality Control), Total Quality Control (TQC), etc. Although there are differences among some of these, they are usually variants of the same theme, namely, to improve the quality of product and services to customers through a company wide focus on quality, and will be collectively referred to as TQM or Total Quality Management.

success in helping companies improve the quality of their operations lies in linking the goals of top management with a set of tools that workers can use to achieve those goals.

Quality improvement efforts are carried out with the purpose of improving the product or service provided *as determined by the customer*, both internal and external. The application of the traditional TQM tools (see Chapter 3) to manufacturing has proven highly successful. Reduction in the number of defects, shortening of manufacturing and product development cycle times, and increasing throughput in the manufacturing process have all been accomplished steadily through the TQM process.

TQM is not just about improving production steps and reducing cycle times, however. *It is a thought revolution in management* (Ishikawa, 1985; Ozawa, 1988). In other words, TQM is about changing the mental models of management in order to enhance an organization's fundamental capability to determine its own future. This change requires more than a one-time shift in thinking; it means continually re-evaluating the way managers think. Sustaining this thought revolution requires not only engaging in the continual improvement activities already accepted by many firms, but also changing the conventional wisdom and mental models shared within an organization—it requires organizational learning.

Many American companies have begun applying the TQM approach in their organizations with mixed results. In this chapter we will look at the results of a study on TQM implementation false-starts and see how systems archetypes can help provide a better understanding of those false-starts. But first, we will begin with a brief history of TQM from its origins in Japan to its eventual adoption in America.

5.1.1 A Brief History of Modern TQM²

After the devastation of World War II, Japan had to rebuild its industrial base almost from scratch. With the help of U.S. occupation forces, they began to apply the modern techniques of quality control in rebuilding their industries. A group of engineers and scholars formed the Union of Japanese Scientists and Engineers (JUSE) to engage in research and to disseminate knowledge about quality control. The concept of quality control was introduced to Japan in 1950 when JUSE invited Dr. W. Edwards Deming, a recognized expert in the field of statistical sampling, to give a seminar on statistical quality control for managers and engineers.

Although the tools proved to be valuable for production problems, managing the process of getting workers to use them effectively was problematic. Dr. Joseph M. Juran's visit in 1954 shifted Japan's quality control emphasis from the factory floor to an overall concern for the entire management. The initial concept of Total Quality Management emerged from that shift in thinking.

Quality Control (QC) began with an emphasis on inspecting-out defects and evolved into the concept of Quality Assurance (QA) controlling manufacturing processes in order to keep defects from being produced in the first place. This idea was later extended to include the product development process—to *design-in* quality from the very beginning. Once the product development process was involved, it was clear that the entire company needed to be included in quality control activities. QC was no longer the province of inspectors performing an isolated function, but a company-wide activity which involved all divisions and all employees.

²For an in-depth history of quality management in America, see Bushe & Shani (1991).

QC Circles grew out of the important role workers played in the actual manufacture of products. The cornerstone of the QC Circle lay in the combination of education in statistical methods with on-the-job application of the tools that were learned. It was based on a strong belief in voluntarism which contributed to a slow start in the initial number of activities, but later mushroomed rapidly.

The results of Japanese TQM activities require little elaboration. Japanese companies have practically decimated many U.S. industries, such as integrated steel, dynamic memory semiconductors, motorcycles, televisions, 35mm cameras, and nearly the entire consumer electronics industry. They have penetrated virtually every market they have entered with superior quality products both in workmanship as well as design. Since the early 70's, Japanese car makers have increased their share of the U.S. automobile market almost every year by producing fuel efficient and higher quality cars. Japanese semiconductor manufacturers currently supply 85% of the worldwide memory chip market. The U.S. steel industry's share of the world market shrank dramatically between 1975 and 1985, and by 1986, imports had risen to 37% of domestic consumption. By 1989, the U. S. went from being the world leader in steelmaking capacity 25 years ago to being in third place behind the former Soviet Union and Japan (Dertouzos, Lester, & Solow, 1989).

5.1.2 TQM Implementation in America

Although it has taken some strong convincing, many U.S. manufacturing firms have begun to implement the TQM way of conducting business and have made significant strides towards improving quality. AT&T cut the development time for their model 4200 cordless telephone from two years to one year while improving quality and lowering costs (Dumaine, 1989). All

three U.S. auto makers have undertaken TQM activities, which have helped reduce defects, cut cycle times and improve customer satisfaction.

Computer and semiconductor manufacturers have also instituted TQM in their organizations. Analog Devices, a leading manufacturer of linear integrated circuits, has included Quality Improvement objectives in their strategic planning and bonus incentives (Stata, 1989). At Hewlett-Packard's Lake Stevens, Washington, facilities, they have cut failure rates for their 30 products by 84% and manufacturing time by 80% over the past three years (Dumaine, 1989). In recognition of their achievements in quality improvement, Motorola received the first annual Malcolm Baldrige National Quality Award, America's equivalent of Japan's prestigious Deming Prize. Florida Power and Light became the first non-Japanese company ever to win the coveted Deming Prize in Japan.

5.1.3 Emerging Problems with TQM Implementations

If the decade of the 80's can be characterized as the great boom years of quality awareness in America, the 90's are beginning to look like a hangover after the big party. After several years and millions of dollars of investment, a large number of companies are sobering up to the fact that their quality efforts have not produced much in terms of tangible results. A study by Arthur D. Little (Kendrick, 1992), which surveyed over 500 American companies, revealed that only a third of them felt that their TQM efforts produced any competitive impact. According to Graham Sharman, an expert on quality with McKinsey in Amsterdam, two-thirds of quality programs that have been in place in Western firms for more than two years "simply grind to a halt because of their failure to produce the hoped-for results."³

³"The Cracks in Quality," *The Economist*, April 18, 1992 (1992)

If TQM has been responsible for giving Japanese firms their competitive edge, why is it not producing results for the majority of Western firms? That's a question many managers are confronted with today, as TQM efforts at their companies falter. According to Boston Consulting Group Vice President Thomas M. Hout, "The majority of quality efforts fizzle out early, or give some improvements but never fulfill their initial promise."⁴ The TQM movement, viewed as the savior of Western industry when it was introduced in the 1980s, appears to be losing momentum in the U.S., as many companies have become disillusioned by their lack of significant progress.

The problem is not that TQM methods don't work, but that most Western companies do not fully implement the total package and sustain the effort. Companies implement pieces, not recognizing that TQM as a system requires *all* of the pieces. Many do not go much beyond implementing the statistical-based tools on the production floor (Brassard, 1989). Although such tools can help pinpoint problems in a machine or a particular process and bring them under control, they do not address the larger organizational issues that need to be overcome. In fact, Deming estimates that statistics-based tools address approximately 20% of the problem; the other 80% has to do with management and systems which govern an organization's policies and often determine its behavior.

5.1.4 The Need for a Systemic Approach to TQM

The Japanese began their TQM activities in the early 1950's and have taken nearly four decades to attain their current level of worldwide prominence. Viewed as a marathon race, the Japanese overtook the U.S. sometime in the '80s by running at a faster pace for the previous two decades—improving

⁴"Where Did They Go Wrong?" *Business Week*, October 25, 1991 (1991).

their manufacturing capabilities at a faster rate than the U.S. Stata (1989, p. 69) presents a provocative argument that, unlike “the Boston Consulting Group’s experience curve theory which says that learning is a function of cumulative volume, independent of time...learning, properly managed, occurs as a function of time, independent of cumulative volume.” Fine’s (1986) work on a quality-based learning model provides theoretical support to Stata’s statement. Fine showed how quality investments can drive learning curve rates faster and produce cost reductions more rapidly than normal experience curves.

If learning were solely volume-based, then the Japanese could never have caught up to the U.S. in industries such as autos. The fact that they did catch up provides tangible evidence that the volume-based learning hypothesis is not generally applicable. The implications are that the U.S. could leapfrog Japan just as Japan previously leapfrogged the U.S. The question remains as to how.

One possibility is to see TQM in the broader context of organizational learning and find a way to accelerate the learning process as a whole. Viewing TQM as a method for improving a product or process can largely ignore a potentially greater benefit—the learning effect that the activity itself has on those involved in improvement activities. Failure to see TQM in the context of learning and ignoring the benefits of learning effects may lead to underinvestment in TQM with negative consequences on both cost and quality competitiveness (Fine, 1988).

Many of the TQM methods and tools (such as, the seven QC tools and the seven management tools covered in Chapter 3, with the exception of the KJ method) are particularly well-suited to advance learning at the *operational* level. As discussed in Chapter 2, however, learning is required at both the

conceptual and operational levels. Compared to TQM's emphasis on operational learning, system dynamics' underpinnings are more conceptual in nature. System dynamics (SD) approaches problems from the basis of the whole, rather than first breaking up the whole into its individual pieces and trying to understand each part. Where TQM focuses on *analysis* of the separate parts that make up the whole, SD strives for *synthesis* of the constituent parts.

From a SD perspective, if a system is decomposed into its components and each component is optimized, the system as a whole will almost certainly not be optimal (Ackoff, 1981). A common characteristic of many complex systems is that they are often designed with the intention of optimizing the parts rather than the whole. In a typical company, the manufacturing function is expected to operate as efficiently as it can. The same goal holds for marketing, accounting, engineering, etc. When a leading manufacturer of linear integrated circuits initially attempted to deploy TQM to improve on-time delivery performance, each function jockeyed to improve its own performance measure, which resulted in no net improvement to the customer.

As shown in Chapter 3, system dynamics provides a methodology for thinking about the ways in which prevailing mental models may restrict learning, for gaining deeper insights into the nature of complex systems, for finding high leverage points in the system, and for testing one's assumptions about the efficacy of various policy choices. Systems archetypes, used to map the dynamics of TQM implementation efforts, can help identify the organizational structures that become barriers to success. The following sections describe an exploration using systems archetypes as a diagnostic tool

in making sense of two companies' experience with TQM implementations that stalled out.

5.2 DESIGN OF TQM IMPLEMENTATION FALSE-STARTS STUDY

Systems archetypes⁵ offer an innovative *diagnostic* tool for understanding complex organizational issues. As presented in Chapter 3, the archetypes provide a simple framework for casting vexing problems in a systemic context. Each archetype represents a dynamic structure that comes with a coherent story line which people can readily understand. The archetypes provide a way to identify a starting point and ending point, as well as help to decide what to include in the diagram. Archetypes help novices see the circular feedback structures that are producing the problematic behaviors they are trying to solve.

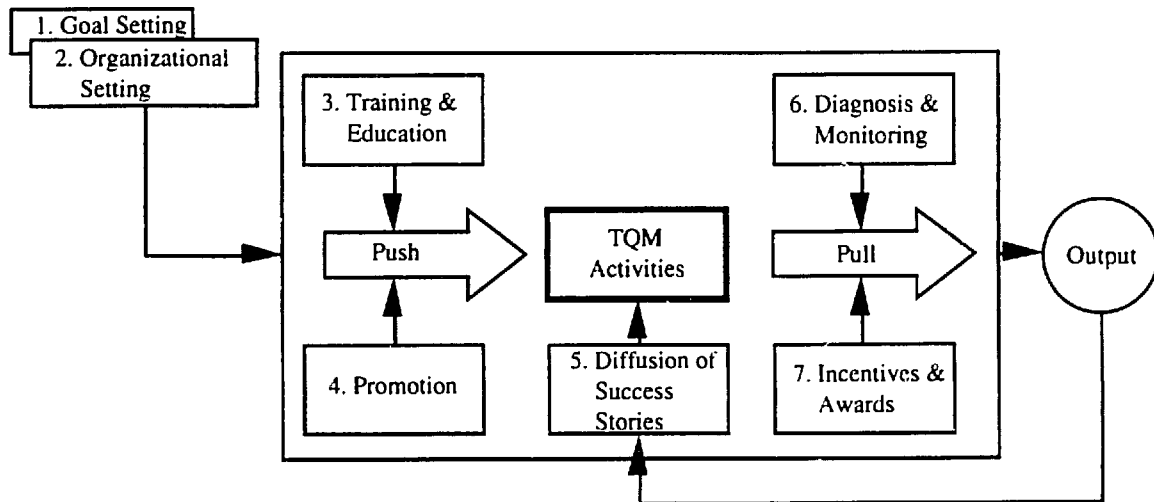
In this study, we used the systems archetypes as a diagnostic tool to better understand TQM implementation dynamics in two companies. More specifically, we wanted to see if the archetypes could reveal a coherent theory of how organizations can fall into "structural" patterns of failure that may be common across organizations.

5.2.1 A Systemic View of TQM Implementation

TQM implementation requires the development of an infrastructure to redesign relationships in the company and to mobilize mass participation. The infrastructure must be able to institutionalize TQM as a new way of doing business. Figure 5.2 shows a TQM implementation plan that identifies the necessary infrastructure for ensuring success. In this model, there are some "push" activities, such as training & education and promotion, some

⁵See Chapter 3 for a summary and description of systems archetypes.

“pull” activities, such as diagnosis & monitoring and incentives & awards, and sustaining activities, such as diffusion of success stories, that serve to reinforce the TQM activities. This system is established through appropriate goals and organizational structures.

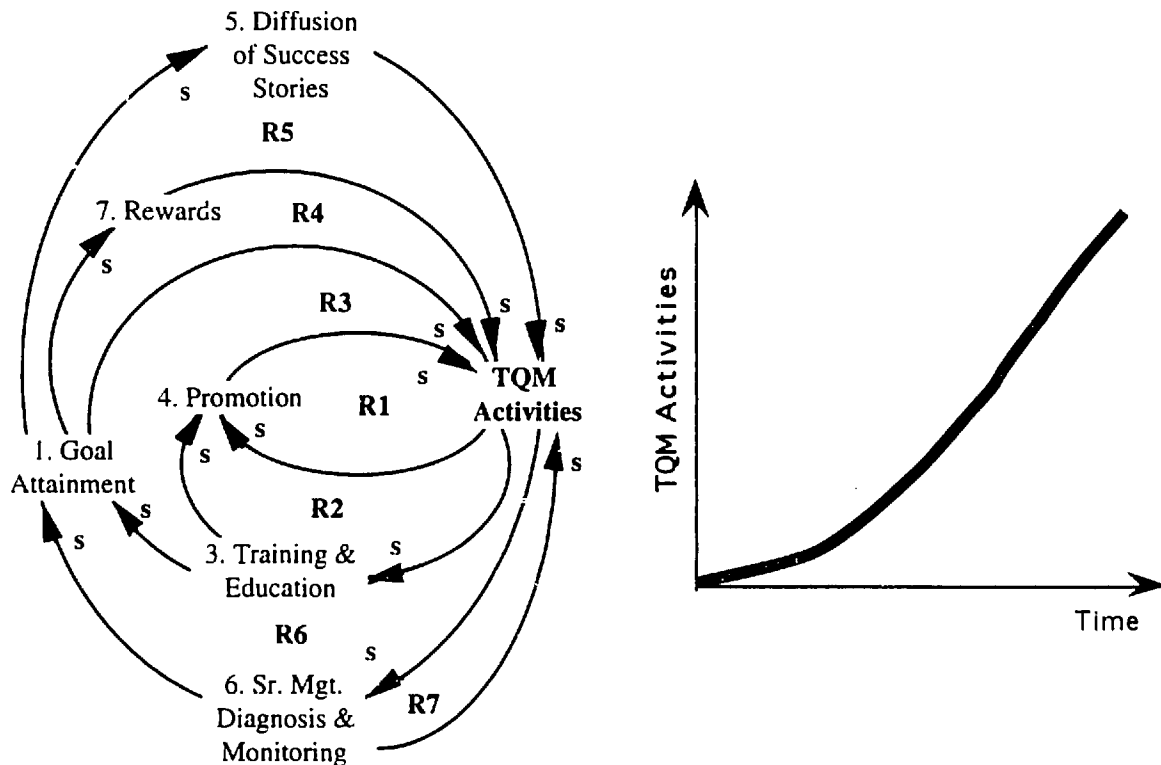


TQM Implementation Model

(source: (Shiba, 1993))

Figure 5.2

From a systemic perspective, each of the areas identified in the TQM implementation plan represents reinforcing processes that are designed to drive the successful growth in TQM activities (see Figure 5.3). As “TQM Activities” increase, “Promotion” activities increase, leading to more “TQM Activities” (R1). Similarly, “TQM Activities” leads to more “Training and Education” which leads to more “Promotion” and increased “Goal Attainment” which lead to more “TQM Activities” (R2 & R3), and so on. The implicit assumption is that if we work hard in pushing and pulling all the right things, we will achieve success. If we focus on the growth drivers, we should expect the pattern of behavior of TQM Activities to resemble exponential growth as shown in Figure 5.3.

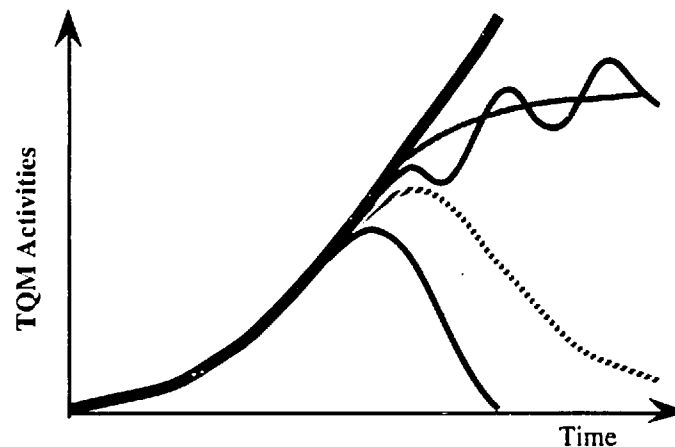


Reinforcing TQM Implementation Loops

Figure 5.3

The reality, at least in U.S. firms, is usually characterized by a behavior pattern that looks more like the one shown in Figure 5.4. As we saw in Chapter 3, that pattern of behavior suggests that a Limits-to-Success archetype may be at work. A Limits-to-Success archetype is characterized by a reinforcing loop and a balancing loop linked together by some performance criteria. When performance growth begins to slow (e.g., TQM Activities), the leverage in most Limits-to-Success cases lies in developing a better understanding of the balancing loop(s). The more obvious action that people take is to “do more of the same” that had worked before, namely, to keep pushing on the factors shown in the reinforcing loop(s). In the long run, this leads to diminishing returns from the reinforcing loops and to continue the shift in “loop dominance” to the balancing loops.

In the seven-point TQM implementation infrastructure above, we identified all of the factors to be a part of reinforcing feedback loops. What is missing in this TQM implementation model is an explicit coverage of all the balancing feedback loops that will resist the effects of the growth loops. The Limits-to-Success archetype is an appropriate starting point to begin looking at TQM implementation false-starts more systemically since the classic Limits-to-Success behavior pattern is consistent with real-life experiences of TQM false-starts.



Multiple Possible Outcomes

Figure 5.4

We conducted a separate study in each of the two companies.⁶ A summary of the findings are presented in sections 5.3 and 5.4. The results from both cases are summarized and discussed in the final section.

5.3 CASE 1: TESTING EQUIPMENT MANUFACTURER—TEST, INC.⁷

In this section, we begin with some background information on the company, Test, Inc., and a brief chronology of TQM efforts there. This is followed by a force-field diagram developed over three separate interviews which captures

⁶For a more detailed coverage of the studies, see Brown & Tse (1992).

some of the major factors identified to be enabling and inhibiting forces in the TQM implementation effort. Causal loop diagrams are developed around several "stories" drawn from the interviews which are then integrated into a single map that contains the interconnections among all the stories. We then identify systems archetypes embedded in the larger diagram, which help us understand the dynamics in terms of systemic "chunks" rather than as a complicated web of spaghetti.

5.3.1 Test, Inc.: Background

Test, Inc. manufactures electronic systems and software for the electronics and telecommunications industry. Most of their customers are component manufacturers who use Test, Inc.'s systems to design and test their products, as well as electronic equipment manufacturers who inspect incoming components and design and test whole circuit boards and other assemblies using Test, Inc.'s systems. The company sells through direct worldwide sales organizations. Their products are so complex, they require extensive support by both the customer and the company.

Two of the key divisions of the company, the Test Equipment Group (TEG) and the Central Manufacturing Group (CMG) were involved in our TQM study. One TEG division, Test Equipment East (TEE), performs final assembly and testing of test equipment that its engineers design, while subsystems and components are manufactured at the separate Central Manufacturing Group.

The story of the TQM false start began with interviews with three managers directly involved in its implementation, chosen to gain different perspectives that people in different organizational levels could give on the

same story. Interview subjects at Test, Inc. were the Director of Quality⁷ at CMG, the Division Manager of TEE, and an Engineering Manager at TEE.

CMG's successful quality program had been set up with the help of the Quality Director, who was helping the TEE division set up a similar quality program at the time of the false start. He belongs to the CMG, reports to the Senior Vice President of that group, and acted as the "quality facilitator" in the false start. The Division Manager, the general manager of the entire TEE division, directs about 250 people and reports to the Senior Vice President of the Test Equipment Group. The Engineering Manager reports to the Division Manager. Both he and the Division Manager are considered "quality implementors" in the story.

5.3.2 Test, Inc.: Chronology

Test, Inc. did not have a corporate TQM program at the time of the false start. Quality programs at Test, Inc. at that time occurred on a division basis. The false start story at Test, Inc. involves the effort by one of the divisions, TEE, to initiate a TQM program.

By the time of the false start at TEE, CMG itself had already started its own quality improvement effort and was well into the TQM process. Interviews with Test, Inc. managers, pinpoint 1983 when the quality program at CMG was initiated through the division's own desire to become a world class manufacturer. Patterned after the methodologies of the Crosby school (Cowan, 1991) the division's full-time quality coordinator, the Director of Quality, educated, planned, and helped implement improvement processes.

⁷Where specific references are necessary, each person is referred to by functional responsibility to preserve anonymity.

Corporate headquarters initiated the quality improvement effort at TEE in a 1989 directive. While the division was suffering a significant loss in market share, major customers were complaining about the poor quality of its products. The company CEO concluded that improving quality would be the key to their recovery. This decision was subsequently passed on to the group vice president of the Test Equipment Group, who in turn directed the Divisional Manager of TEE to begin a quality improvement program.

With the help of the Director of Quality of CMG, a Quality Improvement Team (QIT) was set up at TEE to improve the quality of Engineering Change Orders (ECO) as a start. The team consisted of managers from the different functional groups of TEE and was chaired by the Division Manager himself. The members of the QIT were sent to the Crosby quality college in Florida for training, and an effort was made to devise plans for setting up the quality program at TEE.

However, being a QIT member did not necessarily mean strong commitment to the process. For instance, not everyone on the QIT made it to the quality college. Members allowed phone calls and meetings to take priority and interrupt the weekly QIT meetings, sometimes missing meetings altogether. Accomplishment of action items was sporadic. When the team finally initiated an awareness seminar for the employees of TEE to learn about the quality program, it was prepared not by the QIT members, but by their subordinates. It was delivered not as an informational session as recommended by Crosby, but as a motivational session. In addition, the requisite follow-on employee training never took place, partly due to poor planning, and partly to the corporate-wide TQM program taking shape at the same time. This quality program finally ended and was folded into the corporate effort.

5.3.3 Test, Inc.: Systems Maps

Based on our interview data and the force field diagrams, we identified six stories and developed causal loop diagrams that captured the dynamics of each story. The individual stories were then integrated into a single diagram from which several systems archetypes are identified and summarized.

The following stories—Directive without Commitment, Customers Demand Quality, Cost of Poor Quality, Urgent vs. Important Work, Roll-out without Follow-up, and Training Capacity—were mapped based on the data gathered through interviewing three different people who were involved in the same false-start effort. The systems map of each story along with a brief description is presented below.

5.3.3.1 Directive without Commitment.

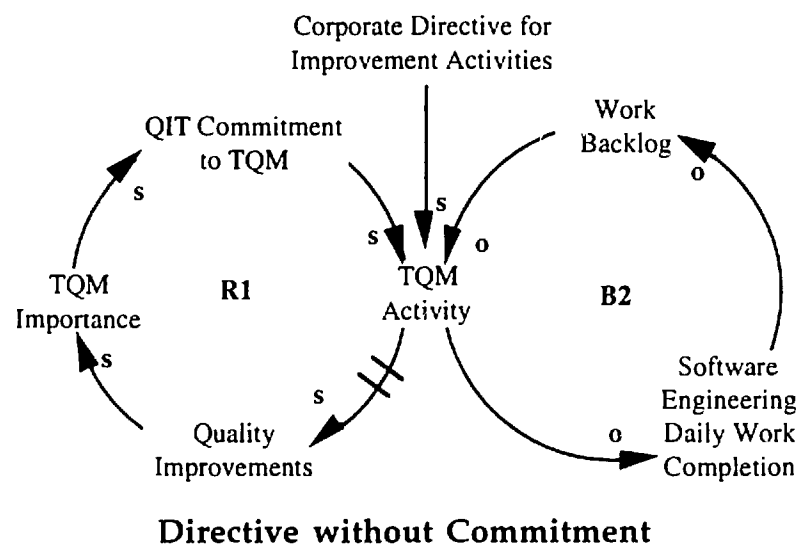
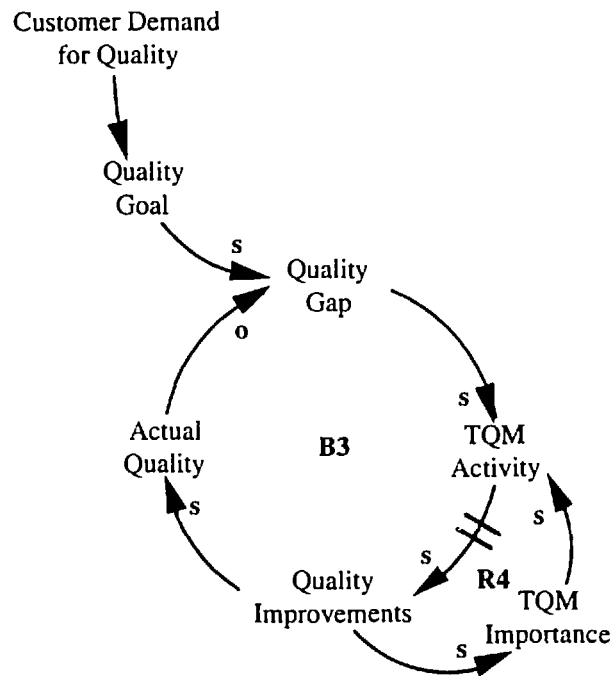


Figure 5.5

The causal loop diagram in Figure 5.5 shows the classic quality reinforcing loop that links TQM activities to quality improvements to commitment and finally to more TQM activities (R1). The directive from corporate to do TQM provides the initial thrust that is meant to start the reinforcing process.

Involvement in TQM activities, however, took time away from the engineers' daily work, resulting in an increase in their "Work Backlog." Reducing the backlog almost always took precedence over conducting TQM activities, thus it provided a balancing force that always took precedence over any TQM activities (B2). Given the continual pressure of Work Backlog, very little was invested in TQM Activity and, given the delay between action and results in Quality Improvements, the reinforcing loop did not have much of a chance to gain momentum. This lack of commitment to TQM activities, shown in the diagram, clearly exhibited itself in the team's choice of priorities.

5.3.3.2 Customers Demand Quality.



Customers Demand Quality

Figure 5.6

In this story, the demand for better quality by an important customer, one well-known for its quality products and one who purchased a lot of test

equipment every year, initiated the corporate directive for quality efforts at TEE in the first place. The division's products were not meeting the customer's expectations of high quality standards from its suppliers. As a result, TQM was implemented at TEE in an effort to close the gap in customer expectations and appease the customer.

As shown in Figure 5.6, "Customer Demand for Quality" raises Test, Inc's own "Quality Goal" which widens the "Quality Gap" between what they have and what is now desired. The gap can be narrowed by engaging in "TQM Activity" that will lead to "Quality Improvements" which lead to true quality improvements over time (B3). When that happens, TQM gains credibility and importance, which can lead to further increase in TQM activities (R4). As processes improve and if customers' demand for quality is stable, the quality gap decreases, making TQM activities seem less urgent and leading to the decline of improvement activities.

5.3.3.3 Cost of Poor Quality.

In the beginning, the true cost of poor quality was not known nor was it really seen to be an important topic. When the "Business Quality Goals" were translated into explicit measures of Acceptable Cost of Poor Quality metrics (at Test, Inc., measured as the cost of additional Engineering Change Orders, ECO's), the "Cost of Poor Quality Gap" was able to be quantified. Once the cost of poor quality (COPQ) was exposed, this heightened people's awareness of the cost of poor quality, as shown in Figure 5.7. As the importance of TQM was recognized, this led to "Improvement Activities" which, over time, reduced "Cost of Poor Quality" and brought the COPQ back to an acceptable level (B5). This is a balancing loop that acts to close the gap between acceptable and actual

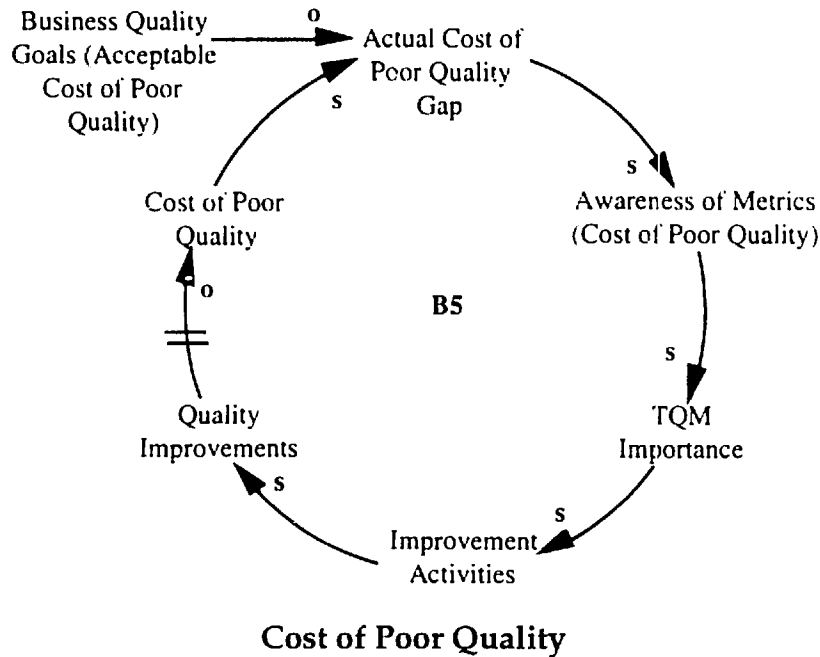


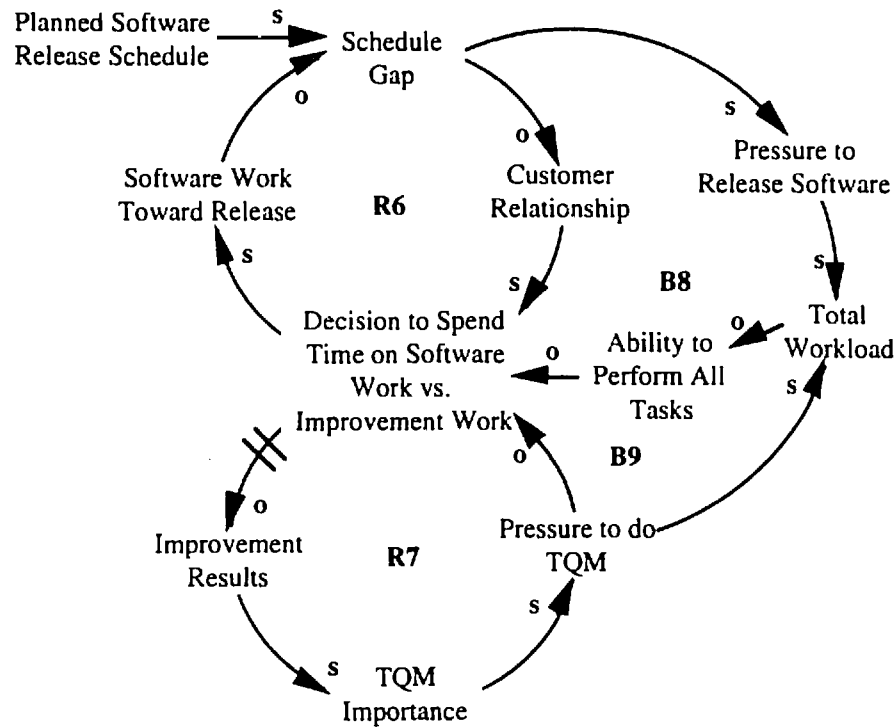
Figure 5.7

COPQ. Obtaining the actual cost figures usually involves a delay because of situations like the one described above and because the cost effects of the quality improvements cannot be assessed immediately.

5.3.3.4 Urgent vs. Important Work.

At Test, Inc., some software engineers were given improvement activities such as collecting data to measure the cost of quality ("important" work), while also being pressured to perform their daily software engineering tasks in order to meet the software release schedule ("urgent" work). But, missed release dates could be detrimental to the relationship with a customer which presented a conflict for the engineers when faced with the decision of choosing which tasks to give more priority to during a time crunch.

The story map in Fig. 5.8 describes a case of Success-to-the-Successful archetype where a common resource (time available) must be allocated between two different tasks (Software work and Improvement work). There



Urgent vs. Important Work

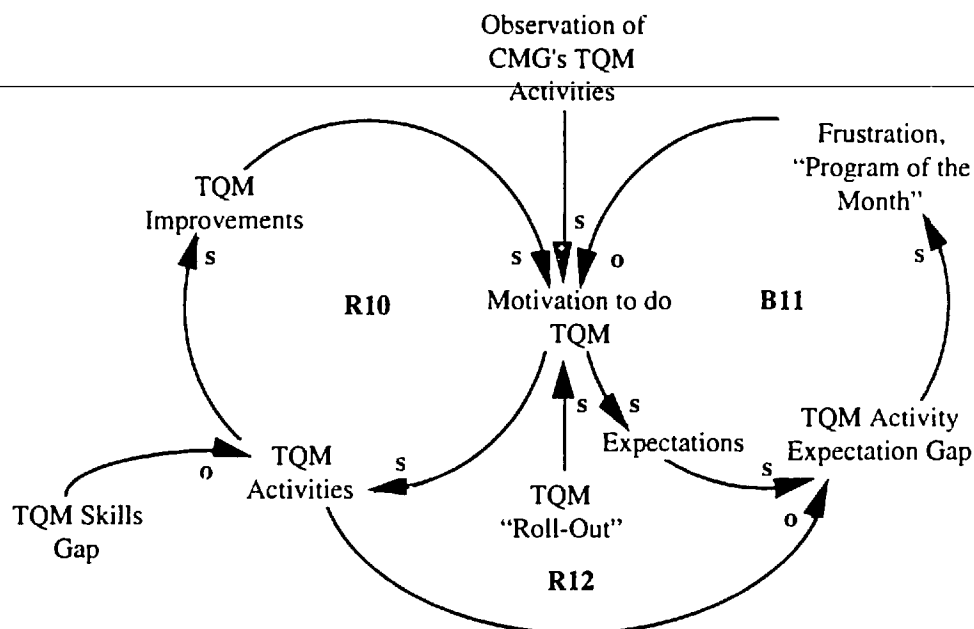
Figure 5.8

are clear and immediate benefits of working on the software release work that shows up in release schedule performance and improved customer relationship. This makes the decision to invest in software release work over improvement work easier and easier (R6), especially when total workload is high. The total workload on the engineers is raised by both the pressure to release the software on time and the pressure to perform TQM activities.

Performing software work does relieve the pressure to release, which increases the ability to perform both types of tasks (B8). On the other hand, TQM activities can lead to improved business results over time, boosting the importance of TQM and the internal pressure to do more. This process could cause people to choose to do more improvement-related work (R7). But the pressure to do more TQM is seen as an added workload by the engineers and the perception that they can't do all the tasks grows leading them to choose

software release work over improvement work (B9). A key point is that results of the improvement work occur only after a time lag, and short-term pressure such as a shortening of the planned software release date tend to siphon off the available resource because it delivers quicker results.

5.3.3.5 Roll-out without Follow-up.



Roll-out without Follow-up

Figure 5.9

This story describes what happened when the employees at TEE were motivated to get involved in TQM without any planned efforts to train them in the requisite skills to carry out improvement activities. The TEE people knew of the successful implementation in the CMG division (located in the same building) so expectations were high that a similar program would be initiated there as well, once a QIT was formed. But the QIT members delegated the planning for the training program to a group that did not fully understand the purpose of their task.

The general manager who had been given responsibility for implementing the TQM activities at his division by the vice president was caught in the middle. He didn't know whether he should wholeheartedly go into it or wait it out. Without adequate follow-up after the initial sessions, motivation turned into frustration as employees grew disillusioned with the lack of change and labeled the TQM effort as another "program-of-the-month." Motivation to engage in TQM activities dropped, and eventually the level of activity also dropped.

The dynamics of the TQM roll out are captured in Figure 5.9. The roll out effort got many people highly motivated to start engaging in TQM activities immediately (see B11). This set up a high level of expectations about what they would be able to engage in shortly after the informational session and quickly led to frustration when nothing happened. Motivation to do TQM dropped. The longer term plan, of course, was to motivate people to engage in "TQM Activities" which would lead to improvements and further motivate people to do TQM (R10). But with high motivation and no TQM Activities, people's expectations went unfilled. The wide gap between expected level of activity and actual activity led people to become frustrated and view the current effort as another program-of-the-month. This led to demotivation which reduced the likelihood of actually engaging in TQM activities even if the skills gap is later addressed (R12).

5.3.3.6 Training Capacity.

At Test, Inc. there was insufficient training of the workers after the initial TQM roll-out seminar. Essentially, the training capacity or program was not in place when it was needed most as described in the above section on "Rollout without Follow-up." As shown in Figure 5.10, the level of "TQM

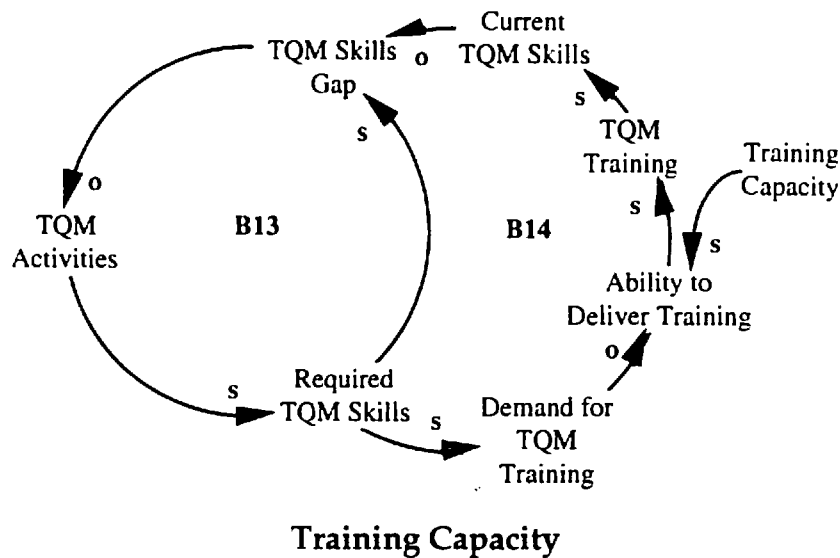


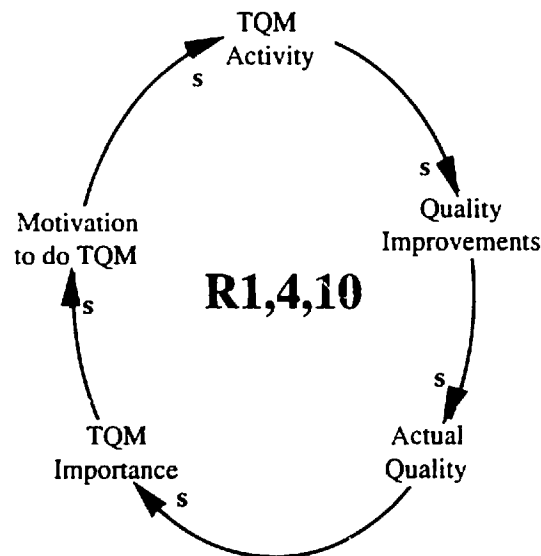
Figure 5.10

Activities” drives the need for higher levels of “Required TQM Skills” because the work will require increasing sophistication of tools and methods. The skill gap increases and TQM activities decrease since people do not have the proper level of training to go beyond what they are currently doing (B13).

One way to remedy this gap is to obtain more training. So, “Required TQM Skills” also drives the “Demand for TQM Training.” But, as training demand increases, the “Ability to Deliver Training” will fall if “Training Capacity” is not expanded to accommodate the additional demand (presuming that it was already operating at full capacity, which it was in this case). “TQM Training” declines since the training resources are not in place to meet the demand, and thus, skills are not upgraded and the “TQM Skills Gap” increases (B14). Both balancing loops act to counter any increases in TQM activities unless adequate training capacity is provided.

5.3.4 Test, Inc.: Integrated Map

In relation to our methodology outlined in Chapter 4, we are in step 7, Build Integrated Map. After mapping each of the individual stories (in section 5.3.3), we want to integrate all the diagrams into a large single map to see how the stories are inter-connected with each other. In general, the identification of such a main loop should be guided by the overall purpose of the activity being studied. Once the loop has been identified and drawn, it provides a framework for integrating the story maps.



Main Reinforcing Implementation Loop

Figure 5.11

In this study, the main objective was to implement TQM as a way of doing business as outlined by the implementation plan presented in section 5.2.1. So, we begin by trying to identify the main reinforcing loop that represents the implementation effort, such as the one shown in Figure 5.11. The launching of "TQM Activity" is expected to produce "Quality Improvements" which will result in higher levels of quality. As people see the results, the importance of TQM is recognized and motivates more people

to engage in "TQM Activities." This is the basic story line that was in loops R1, R4, and R10.

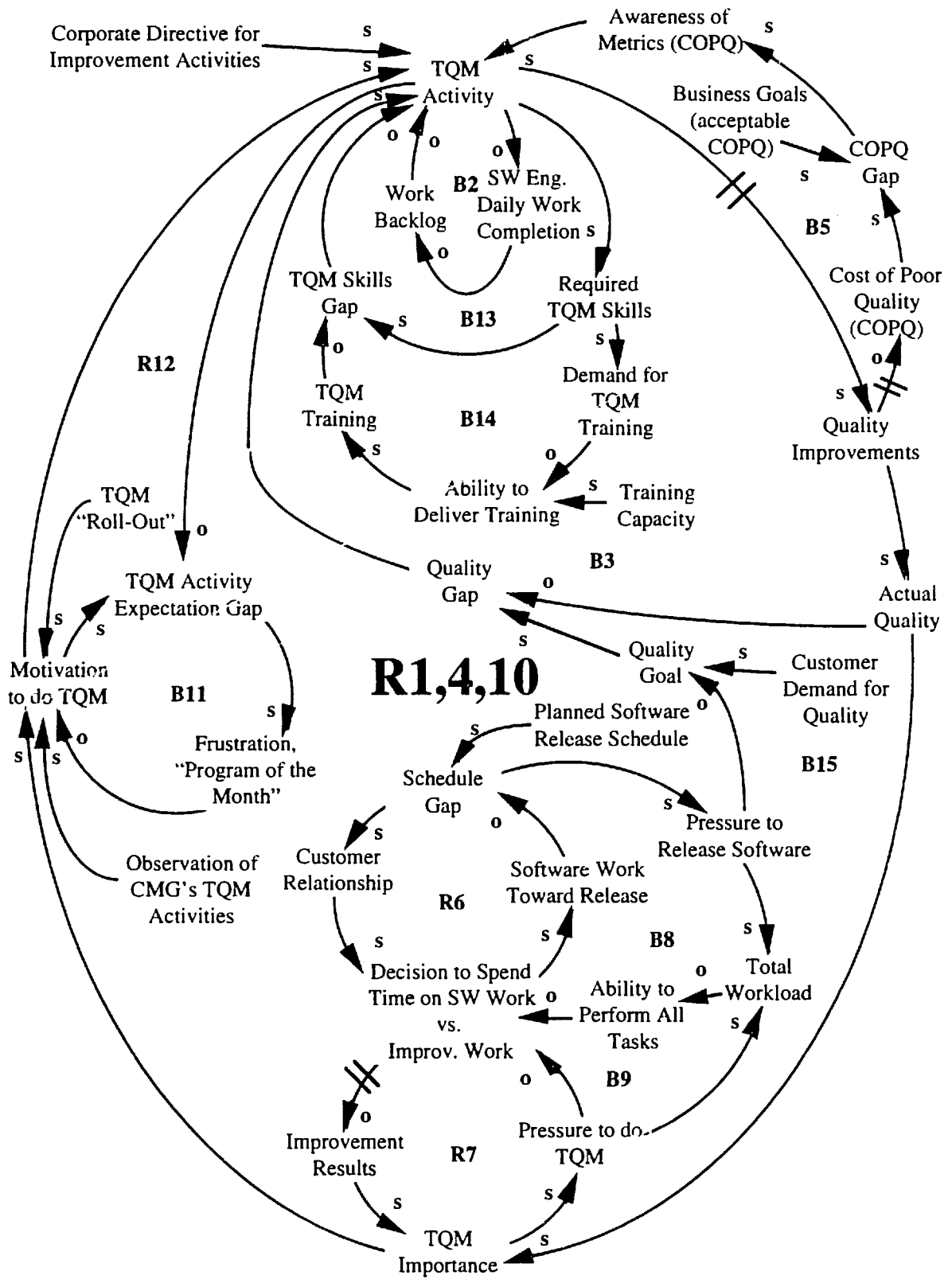
Once the main loop is laid out, the various stories can be "attached" to it. A complete map is shown in Figure 5.12.

5.3.5 Test, Inc.: Systems Archetypes

The integrated map (Figure 5.12) contains the essential features from all the stories identified in this case study and has been double checked with the interview stories and the interviewees. The integrated map, while extremely useful for displaying all the interconnections, however, can be difficult enough for those intimately involved in the study to comprehend, let alone communicate to those who are totally unfamiliar with the case. We found it useful to simplify our integrated maps by applying the concept of abstraction and abstracting upwards to find larger "chunks" of stories embedded in the integrated map that can be captured in terms of systems archetypes.

In the above discussions on the individual stories, there were cases where the presence of an archetype was pretty clear (for example, Success-to-the-Successful in Urgent vs. Important work). There may be additional archetypes that won't surface until all the stories have been integrated into a single map because the archetypal structures may be contained in the interlinkages with other story loops. It is also possible that no new archetypes will emerge.

The large reinforcing loop that is the focus of the implementation effort is coupled with many balancing loops which can either slow things down (as in the case of training) or completely derail it (as in the program-of-the-month frustration). Using the descriptions of systems archetypes as dynamic scripts in Chapter 3 (see Figure 3.29), we can decompose the integrated diagram into



Test, Inc. Integrated Map

Figure 5.12

<p>3. Limits-to-Success (Roll-out without Follow-up)</p>		<p>The Roll-out without Follow-up LtS structure can easily happen in many companies because it is so easy to get the “hype” ahead of the substance. Launching promotional campaigns and speeches can be done relatively quickly and get everyone pumped up. The time required to actually get things in motion and the investments that are required to build supporting structures are usually a lot longer.</p>
<p>4. Limits-to-Success (Training Capacity)</p>		<p>The Training Capacity LtS structure is a classic case of a system running up against a resource limit. The catch-22 for most organizations is that they don't want to have training capacity until there is a demand for it. But, unless there is training capacity available, it is likely that demand will not grow. A tough choice for organizations is deciding to put in training capacity ahead of current demand, and base it instead on future demand that would be generated by a successful reinforcing loop.</p>

<p>5. Success-to-the Successful (Urgent vs. Important Work)</p>		<p>This case of Success-to-the-Successful can occur whenever you are introducing something new into an established environment. Unless special measures are put in place as added reinforcement for the new work, the choice preferred by the existing system will generally be the one that the players are already accustomed to and are currently being rewarded for. This tendency is referred to as competency traps in the organizational literature (Levitt & March, 1988)</p>
<p>6. Drifting-Goals (Customers Demand Quality)</p>		<p>This Drifting-Goals structure that is drawn from the Customers Demand Quality story reflects a general vulnerability of goals to erode due to internal pressures. In this case, increased TQM activity takes time away from software release work, which creates schedule pressure that will have a negative effect on the quality goal. This is why having a goal that is set by an absolute standard (e.g., zero defects) or by competitors (e.g., benchmarking) or by the customers (e.g., voice of the customer) is important for TQM efforts to stay on track.</p>

5.4 CASE 2: SEMICONDUCTOR MANUFACTURER—WAFERS, INC.

In this section, we describe the background of a second company, Wafers, Inc., give a brief chronology of their TQ efforts. An integrated systems diagram of their story is presented from which we will draw out the systems archetypes

that captures the stories told by the interviewees of their company's experience.

5.4.1 Wafers, Inc.: Background

Wafers, Inc., a division of a multinational manufacturer of semiconductors and related components, is a major producer of high performance linear and mixed-signal integrated circuits (ICs) used in precision data acquisition systems with approximately 1000 employees and worldwide annual corporate sales in excess of \$500 million. Wafers, Inc. also manufactures digital ICs, signal processing components, board-level subsystems and systems. Original equipment manufacturers (OEMs) who incorporate the company's products into a wide variety of instruments and systems comprise a large portion of the company's sales. Among its many markets are military/aerospace, computers, consumer electronics, industrial automation, telecommunications, and automotive/transportation.

All seven manufacturing operations worldwide are under the responsibility of one Vice President of Manufacturing. Several factors drive this centralized manufacturing structure. In the past, technological innovation drove the company's success—introducing new proprietary products offering unique features. As long as the final product performed as promised, production quality was not seen as a big concern. When company's customers came under intense competition, they began to demand lower prices, higher quality and better service than they had in the past. In addition, Wafers, now entering new consumer markets where high volume manufacturing at low cost is becoming as important as technical innovation, must adjust to the fast-paced nature of their customers' markets, minimizing

delivery delays and meeting promised delivery dates, if they are to retain market leadership.

As done at Test Inc., our TQM false start study at Wafers, Inc. began with interviewing three people directly involved in that TQM implementation, again chosen to gain different perspectives on the same story from people in different organizational positions: the Vice President of Quality, the Operations Manager of the Semiconductor Division, and the General Manager, also of that division.

The Quality V.P. at Wafers, Inc., reporting directly to the company CEO, is responsible for facilitating quality implementation efforts within the company across all the divisions. The Quality V.P. was the "quality facilitator" in the TQM false start story. The Operations Manager in the Semiconductor Division, one of the "quality implementors" in the false start study, reports directly to the Vice President of Manufacturing at Wafers, Inc., but also answers indirectly to the General Manager of the division. The General Manager, formerly a product-line manager at one of Wafers, Inc.'s manufacturing facilities, now manages the company's largest division, the Semiconductor Division.

5.4.2 Wafers, Inc: Chronology

The company's quality improvement efforts began in 1984, when the emphasis throughout the company was on design and marketing. Manufacturing was seen as "a necessary evil in order to get customers to part with their money."⁸ Between 1982 and 1988, the company missed their five-year goals by a wide margin. The company CEO, rather than blaming the disappointing results on the malaise of the U.S. electronics industry at the

⁸Brown & Tse, 1992, p. 22.

time, thought they were due to a pervasive internal problem, and instituted a push for quality in both product and services aimed at satisfying the customer. This company's quality initiatives began with the "Crosby" approach in the early 80s, and shifted into the "Juran" approach in the mid 1980s. It is now embracing the broader philosophy of Total Quality Management.

The company instituted the office of Quality Manager (which has now become the Quality V.P.) and hired a veteran of quality programs to fill the position, someone who brought along with him a knowledge of and experience with quality improvement methodologies, including industry benchmarking. He helped the managers of Wafers, Inc. to organize their improvement efforts and set realistic goals as improvement targets.

Wafers, Inc. made efforts to improve the company's on-time delivery performance of products to customers at Wafer's Semiconductor Division (WSD). Management at WSD saw the basic problem as the unreliability of the factory's committed delivery dates. The factory often missed up to two or three committed delivery dates in a single order (Kim, 1986).

Managers at Wafers, Inc. were aware of the problem with on-time delivery performance before the Quality VP came into the organization. However, without a process methodology in place, efforts to improve performance only led to varying degrees of finger-pointing among the managers.⁹

Moreover, the incentive scheme apparently made matters worse. Managers' performance evaluations were directly related to the amount of

⁹This information was taken from interviews with the Operations Manager, as cited in Brown & Tse, 1992, p. 2 3.

shipments delivered during the quarter, resulting in a big boost in shipping effort near the end of each quarter to push products out the door.

TQM soon gained high visibility, thanks to the efforts of the Quality V.P. The company's upper management responded well to the process improvement methodologies he had suggested. Results of the quality improvement efforts were the first item on the agenda at quarterly senior management meetings. The Quality V.P. developed a quarterly scorecard, making visible comparative ranking between divisions and pressuring the non-performing divisions to institute TQM into their operations. Initial resistance, especially among divisions at a greater distance from corporate headquarters, waned over time, and before long, TQM was adopted among all the divisions. Indicators that quality was improving included on-time delivery performance which increased from 60% in 1986 to 97% in 1990.

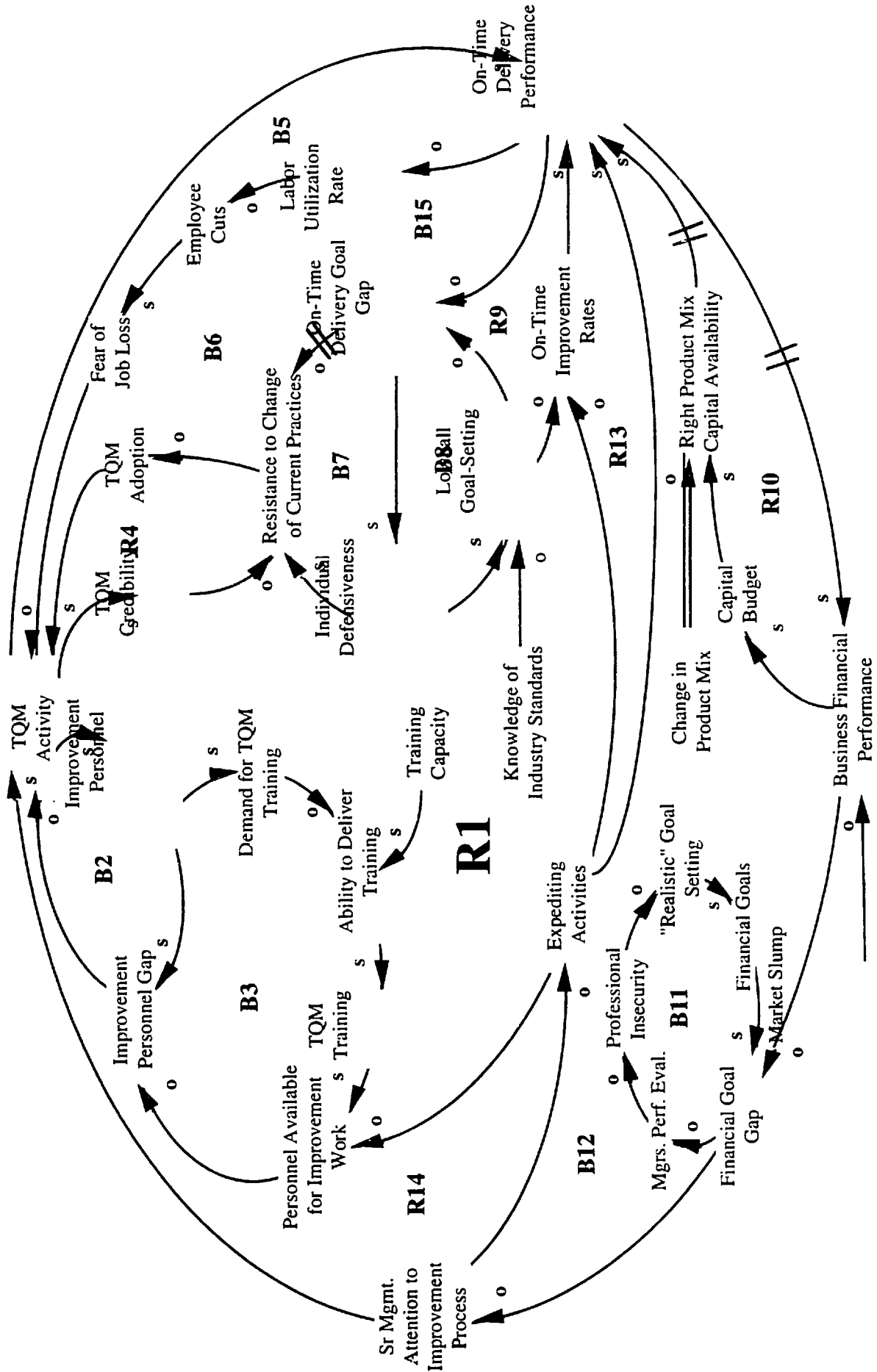
Then in 1990, the business of Wafers, Inc. underwent several significant changes. Internally, the acquisition of another company in a distant location caused some confusion, and required organizational changes. Externally, the electronics industry was in a downturn reflected in their fall from favor on Wall Street. The stock price of Wafers, Inc. dropped precipitously, and management was concerned about takeover threats. As a result, management attention was focused away from quality and other long-term programs and more on the short-term financial survival of the company. Quality improvement was now last on the agenda of quarterly management meetings and then fell off the agenda completely, a clear sign that it was no longer a priority item in the company. Improvement activities slowed as short-term shipping activities became more important. On-time delivery performance deteriorated from its peak of 97% in 1990 to the low 90's as of mid-1991.

We conducted a case study of their shipping performance activities based on their assessment that it constituted a quality improvement “false start.” In other words, improvements were made and then subsequently lost. In this analysis, the Division’s General Manager, Quality Manager and Operations Manager were interviewed for approximately 2 hours each.

5.4.3 Wafers, Inc.: Integrated Map

The integrated diagram presents a rich map of the pattern of events which still preserves each mini-story without losing the details.¹⁰

¹⁰See Company A in Brown & Tse (1992) for a description of the individual stories.



Wafers, Inc. Integrated Loop Diagram (adapted from Brown & Tse, 1992)
Figure 5.13

5.4.4 Wafers, Inc.: Systems Archetypes

Archetype Name (and Story Source)	Archetype Structure	General Lessons
<p>7. Limits-to-Success (Dimming the Spotlight)</p>		<p>This Limits-to-Success (LtS) structure is likely to be encountered in any company setting where TQM activity is dependent on improvement personnel to support that activity.</p>
<p>8. Limits-to-Success (Train the Troops)</p>		<p>This is very similar to the "Training Capacity" story from Test. In both cases, the inability to meet early demand for training that was generated by the TQM Activity served to dampen TQM activity. Note that the "logic" of the loop would suggest that the decrease in TQM Activity would erode delivery performance, create financial problems which would take senior management attention away from TQM and lead to a downward spiral. However, this should be interpreted as "the loop has the potential to do that if the R1 loop was not hooked up to any other loops. In reality, things are never that cleanly divided and so we must always remember the larger systemic context from which we 'decomposed' the archetype."</p>

<p>9. Limits-to-Success (Productivity Dilemma)</p>		<p>The Productivity Dilemma presents a real problem for any organization that is improving its productivity without a commensurate growth in its customer demand. If productivity grows faster than demand, then labor utilization decreases and the traditional view of a financially sound business decision would be let the unneeded workers go. Thus, according to loop B5, fear of job loss will tend to keep TQM activity at a level that is not job-threatening.</p>
<p>10. Limits-to-Success (Resistance to Change)</p>		<p>Any change effort is likely to encounter resistance from some members of the targeted organization which is a natural and expected reaction ((Schein, 1987c) The resistance dynamic captured in this set of loops is a tricky one to overcome because the resistance is in the form of a mindset or mental model that is predisposed to not value TQM. When TQM activity succeeds in improving on-time delivery performance, for example, the goal gap decreases, which <i>increases</i> resistance to change because the improved delivery performance is perceived as a signal that the old ways are working just fine. Thus, TQM adoption decreases and TQM Activity is also decreased.</p>

<p>11. Success-to-the-Successful (Ship or Measure)</p>		<p>This story of Ship or Measure is similar to the Urgent vs. Important Work story in Test., Inc. In Wafer's case, it is Senior management's attention that gets pulled in one direction over another. If there is constant attention given to TQM, delivery performance will improve with better financial results. This reinforces Sr. management's attention to TQM (R1). If, on the other hand, attention is given to Expediting, due to a dip in financial performance, expediting goes up which has a negative effect on On-Time Improvement Rates which undermines delivery performance in the long run and leads to more financial pressure to focus on short-term expediting (R13).</p>
<p>12. Drifting-Goals (Resistance to Change)</p>		<p>The Resistance to Change story is also part of a Drifting-Goals archetype. The dynamics in this case revolve around people's insecurity about achieving delivery goals which leads to defensiveness and "low balling" goal setting. By setting goals that are achievable with current practices, the resistance to change increases. This stifles TQM activity and lowers delivery performance. The resulting gap creates more defensiveness, reinforcing the tendency to low-ball goal setting.</p>

5.5 DISCUSSION

Based on these two case studies of TQM implementation false starts, we can draw some tentative conclusions about the common pitfalls that all companies may be susceptible to. We started this chapter by pointing out that TQM implementation plans tend to focus on launching and managing the reinforces forces, such as training and promotion, without an explicit focus on identifying the balancing forces that are important for a successful implementation. On the basis of the two case studies, a tentative recommendation can be mad that the Limits-to-Success archetype may be a good point from which to start an investigation of TQM false-starts. Our preliminary results indicate that this archetype comes up repeatedly in the company stories.

5.5.1 Common Failure Modes

The most common archetype identified was Limits-to-Success with eight occurrences. There were two Success-to-the-Successful and two Drifting-Goals archetypes identified. A surprising result was that none of the other archetypes appeared in the stories. In other organizational settings, Tragedy-of-the-Commons (ToC) or a Shifting-the-Burden (StB) archetype frequently surface as one of the organizational stories (see Chapter 6, for example). In retrospect, however, it makes sense that these are not likely to show up in cases where the implementation never really takes off because they either explain the entrenchment of behaviors over a long period of time (StB) or the dynamics of multiple demands on a fixed resource (ToC).

The archetypes and stories identified in this study highlight three major issues that may be relevant to other company settings: (1) Constancy of Purpose, (2) Training Capacity, and (3) Old vs. New. The lessons and

implications of these issues will be discussed as well as how they fit with some of the existing quality literature. A fourth issue, Productivity Dilemma, was not common to both companies, but it still has potential implications for all companies engaging in TQM and other productivity enhancing work.

Company	Constancy of Purpose	Training	Old vs. New
Test, Inc.	<ul style="list-style-type: none"> • Cost of Poor Quality (LtS) • Roll Out without Follow Up (LtS) 	<ul style="list-style-type: none"> • Training Capacity (LtS) 	<ul style="list-style-type: none"> • Urgent vs. Important Work (LtS) • Customers Demand Quality (DG)
Wafers, Inc.	<ul style="list-style-type: none"> • Dimming the Spotlight (LtS) • Ship or Measure (StS) 	<ul style="list-style-type: none"> • Train the Troops (LtS) 	<ul style="list-style-type: none"> • Resistance to Change (StS) • Ship or Measure (StS) • Resistance to Change (DG)

Three Major Systemic Issues in TQM Implementation

Figure 5.14

5.5.1.1 Constancy of Purpose

One of Deming's fourteen points is creating constancy of purpose for improvement. He states:

There are two problems: (i) problems of today; (ii) problems of tomorrow, for the company that hopes to stay in business...

Problems of the future command first and foremost constancy of purpose and dedication to improve the competitive position to keep the company alive and to provide jobs for their employees. (Ernst & Young, 1991, p. 24-25)

Deming admonishes people that fixating on solving the problems of today is not enough. You can get better at solving current problems as you slowly go out of business. A constancy of purpose is required, however, to keep focused

on the problems of tomorrow because of the constant demands of today's problems.

Juran (1982) also stresses the importance of the role senior management plays in a successful implementation. His prescription includes having a formal infrastructure in place from the very top of the organization to the individuals on the factory floor. TQM implementation should be a well-planned out process with full commitment (not just involvement) by top management.¹¹ The experiences of these two companies as represented by the archetypes shows why this is such an important point.

The Cost of Poor Quality (LtS) and Roll-out without Follow-up (LtS) stories from Test and Dimming the Spotlight (LtS) and Ship or Measure (StS) stories from Wafers all indicate a lack of constancy of purpose. In the Cost of Poor Quality story, the TQM activity would be affected by the ebb and flow of focus on the Cost of Poor Quality" (COPQ) measures. If there was enough of a gap that it couldn't be ignored, efforts to make improvements would increase. When COPQ and the COPQ gap fell, so would the focus on TQM activities.

In the Roll-out without Follow-up, there was an inconsistency between people's expectations and the actual level of activity that followed some initial meetings. The expectations were based on people's observations of what was happening at another division at the same location, but the efforts begun with them were not the same. This lack of consistency of action was seen as a lack of senior management commitment and led to a reinforcement of a prevalent belief that this was yet another program-of-the-month.

¹¹At a seminar for top management at Wafers, Inc. Juran explained the distinction between involvement and commitment. He said that when you have ham and eggs for breakfast, the chicken was involved in producing it, but the pig was committed.

Dimming the Spotlight points out the dynamic where inadequate attention is paid to making sure there are enough 'improvement' personnel available to *sustain* a TQM effort. If senior management pays attention to TQM in the beginning of the effort and then gradually dims their spotlight on it, the result is likely to be another Limits-to-Success situation.

The Ship or Measure story points to the important issue of performance measures which govern an organization's actions. If the business financial performance indicators that senior management pays attention to are based solely on the results of short term activities, like quarterly shipments, then the incentive for workers to invest in TQM activities is low. People know how to "win" by the old measures.

All four stories emphasize the need for a clearly articulated constancy of purpose. The window of opportunity for establishing credibility may be relatively short in organizations who have had many programs come and go.

5.5.1.2 *Training Capacity*

One of the difficulties that is probably common to a lot of organizations is having just enough training capacity to handle current demand without having more than is required. The two stories are virtually identical.

The Training Capacity story focuses more on the skills gap while the Train the Troops story focuses more on the personnel gap. Either for lack of skilled people to help the workers or the lack of skills of the workers, the TQM activities will suffer. The Training Capacity (LtS) and Train the Troops (LtS) stories suggest that the downside of having inadequate capacity may be greater than the costs of having too much capacity, especially in the early stages of implementation. Given the level of skepticism with which most

new programs are received, the success or failure of the first phase of an implementation can have far greater effect than any later efforts.

5.5.1.3 *Old vs. New*

Introducing a new way of doing things into an organization raises the issue of how to balance the new with the old. Five of the stories were related to this issue: Urgent vs. Important Work (StS), Customers Demand Quality (DG), Resistance to Change (StS), Ship or Measure (StS), and Resistance to Change (DG).

The Urgent¹ vs. Important Work and Ship or Measure stories point to the same issue of when faced with a choice under a lot of work pressure, the day-to-day work needs almost always wins out. The visibility of the results, both in terms of the formal measures as well as the immediate experience of getting tasks accomplished (engineering work or expediting shipments), for the short term work makes this a classic case of the Success-to-the-Successful archetype. From a management perspective, the archetype highlights the need to explicitly address the issue of work overload when implementing something new that will require time and effort for people to learn. It is the principle of short term pain vs. long term gain. Unless senior management changes the system to accommodate that tradeoff, the short term pain is likely to govern the workers' actions.

Resistance to Change is also a Success-to-the-Successful archetype but is slightly different in nature. This one is based more on a mindset or worldview that is not open to new ideas. Providing enough time and training will not be enough to overcome this hurdle of ingrained cynicism, which is why Ishikawa (1985) says it requires a thought revolution in

management. Senior management commitment and constancy of purpose is critical for overcoming this kind of resistance.

A TQM implementation can face an Old vs. New issue through a Drifting-Goals structure, as in the stories of Customers Demand Quality and Resistance to Change. Without having an absolute measure that is not measured relative to how things are currently being done, the goals are susceptible to drifting depending on daily work pressures and entrenched mindsets.

5.5.1.4 *Productivity Dilemma*

Although this story was represented in only one of the stories, it raises an important general issue—as you increase productivity, what do you do when you need fewer people to do less work? This question is very relevant to companies who are working to accelerate organizational learning as well as those engaged in TQM. It also relates directly to Deming's point about driving out fear in the workplace.

It is popular to refer to people as a company's most important asset and to view productivity improvements as increasing one's asset base. This, however, is more rhetoric than substance because people only show up on the expense side of company income statements. Hence, when productivity improvements are gained, financial analysis suggests reducing the "unnecessary" people to make performance look even stronger.

But what if we *really* treated employees as assets—not just in words but on our financial statements? For starters, we would add another category on the asset side of the balance sheet and devise a way to assess the value of the intellectual capital of the organization. Unlike physical assets, people assets could *appreciate* over time. We would still account for people's salaries,

benefits, and other employee-related costs as expenses, but we would also have a corresponding valuation for the people-capacity of the organization. The people-asset column would provide corporate visibility that the people-capacity was being enhanced.

Putting employees on the balance sheet as assets could also change the way we think about cost-cutting. Training would be viewed as an investment and would not be automatically cut when times get tough. Vacation days would be considered vital investments that help our most important assets become even more productive. Employees would not be seen as expenses to be cut out, but assets which we could invest in and expect to get a return in terms of higher productivity, new products, better quality, and a myriad of other possibilities that we have not yet begun to identify.

5.5.2 Next Steps

In this study, we used the methodology developed in Chapter 4 to understand better what the organizational pitfalls are when implementing TQM. The stories and systems maps were grounded in the data provided by the interviewees. In terms of organizational learning, the study helped the managers take the first step by making their individual experiences explicit in a form that captured the dynamic richness of their organization's experiences with TQM implementation.

Both the integrated map and the systems archetypes can provide a means to begin sharing the current understanding of their TQM experience with others in the organization. Although this study did not include taking the next step of trying to make the systems maps widely shared, that step could be done as a follow up to this study. We could develop a measurement of how widely the structures identified in the model are shared in the organization

before and after being introduced to the diagram. This would be an assessment at the level of conceptual learning and frameworks. Another measurement that would need to be designed is assessing the different actions that people took on the basis of the new insights—that is, measuring operational learning and assessing its link to conceptual learning.

A useful result would be the development of a whole typology of implementation patterns of failures and successes represented in the framework of archetypes and causal maps. To build such a typology we need to replicate this study with about a dozen more field sites. The results of this initial two-company study suggests that there may be some common patterns that companies can fall into (such as the Limits-to-Success structures summarized in the charts above). Having such a typology could help companies understand how their implementation plans went awry so that they can design ways not to fall into the same traps. Perhaps more importantly, new companies venturing into a TQM implementation can learn from the experience of others and avoid falling into the same organizational traps.

5.5.3 Final Note

In a recent article, Drucker (1990) writes about the “postmodern factory” whose essence will be defined by the synthesis of four principles and practices—Statistical Quality Control (SQC), new manufacturing accounting, “flotilla” or module organization of manufacturing processes, and a systems approach. According to Drucker, the integration of these four concepts will build a new theory of manufacturing where every manufacturing manager “will have to learn and practice a discipline that integrates engineering, management of people, and business economics into the manufacturing

process" (p. 102) Implicit in Drucker's theory is the underlying goal for creating an environment conducive to continual learning.

Being TQM-driven means creating such an environment by advancing continuous improvement at every level of the organization. From the factory worker to the CEO, the goal is to become better learners. But, to be lasting and significant, organizational learning must advance on both the operational and conceptual level. Learning at one level without the other is like trying to run a marathon with one foot nailed to the starting line; you can be off to a quick start, but you won't get very far.

Total Quality Management and system dynamics have complementary strengths that can greatly enhance an organization's ability to improve its performance through a more balanced learning process. The integration of the two approaches can provide the synergistic boost that will help U.S. firms reassert their competitiveness and build the foundations for a new type of organization—a *learning organization*, where front line people work in self-managed groups, managers develop their research skills and take on the role of theory-builders, and leaders become more like philosophers who inspire the human spirit. At the core of these learning organizations will be learning systems and processes that are firmly rooted in the two disciplines of TQM and system dynamics. Systems archetypes can play an important role in helping companies see their organizations more systemically.

Chapter 5 References

- Ackoff, Russell L. (1981). *Creating the Corporate Future: Plan or Be Planned For*. New York: Wiley & Sons.
- Brassard, Michael (1989). *Memory Jogger Plus+*. Methuen, MA: Goal/QPC.
- Bushe, Gervase R., & Shani, A.B.(Rami) (1991). *Parallel Learning Structures*. Reading, MA: Addison-Wesley Publishing Co.
- Business Week* (October 25, 1991). Where Did They Go Wrong?

- Cowan, D.A. (1991). Effect of Decision Making Styles and Contextual Experience on Executive Descriptions of Organizational Problem Formulation. *Journal of Management Studies*, 28(September), 463-83.
- Dertouzos, Michael L., Lester, Richard K., & Solow, Robert M. (1989). *Made in America: Regaining the Productive Edge*. Cambridge, MA: MIT Press.
- Drucker, Peter F. (1990). Emerging Theory of Manufacturing. *Harvard Business Review*(May-June), 94-102.
- Dumaine, Brian (1989). How Managers Can Succeed through Speed. *Fortune*, February 13, 54-59.
- The Economist* (April 18, 1992). The Cracks in Quality.
- Ernst & Young (1991). *International Quality Study*.
- Fine, Charles H. (1986). Quality Improvement and Learning in Productive Systems. *Management Science*, 32 (10), 1301-1315.
- Fine, Charles H. (1988). A Quality Control Model with Learning Effects. *Operations Research*, 36(3, May-June), 437-444.
- Ishikawa, Kaoru (1985). *What is Total Quality Control?* Englewood Cliffs, NJ: Prentice-Hall.
- Juran, Joe M. (1982). *Upper Management and Quality*. Wilton, CT: Juran Institute, Inc.
- Kendrick, John J. (1992). Companies Continue to Embrace Quality Programs, But Has Total Quality Generated More Enthusiasm than Results? *Quality*, 31(5 - May), 13.
- Kim, Daniel H. (1986). *Identifying Multi-Loop Feedback Structures Relevant to Delivery Delays and Customer Satisfaction in a Major Semiconductor Company*. Cambridge, MA: MIT Electrical Science and Engineering.
- Levitt, Barbara, & March, James G. (1988). Organizational Learning. *Annual Review of Sociology*, 14, 319-340.
- Ozawa, Masayoshi (1988). *Total Quality Control and Management*. Tokyo: JUSE Press.
- Schein, Edgar H. (1987c). *Process Consultation Volume II: Lessons for Managers and Consultants*. Reading, MA: Addison-Wesley.
- Shiba, Shoji, Graham, Alan, and Walden, David (1993). *A New American TQM: Four Practical Revolutions in Management*. Cambridge, MA: Productivity Press.
- Stata, Ray (1989). Organizational Learning—The Key to Management Innovation. *Sloan Management Review*, 12(1), 63-74.

Chapter 6

Mental Models and Organizational Learning: Product Development Management[†]

*"We have the technology, the people, the skills, all the pieces—
why can't we put it all together?"*

—product development manager

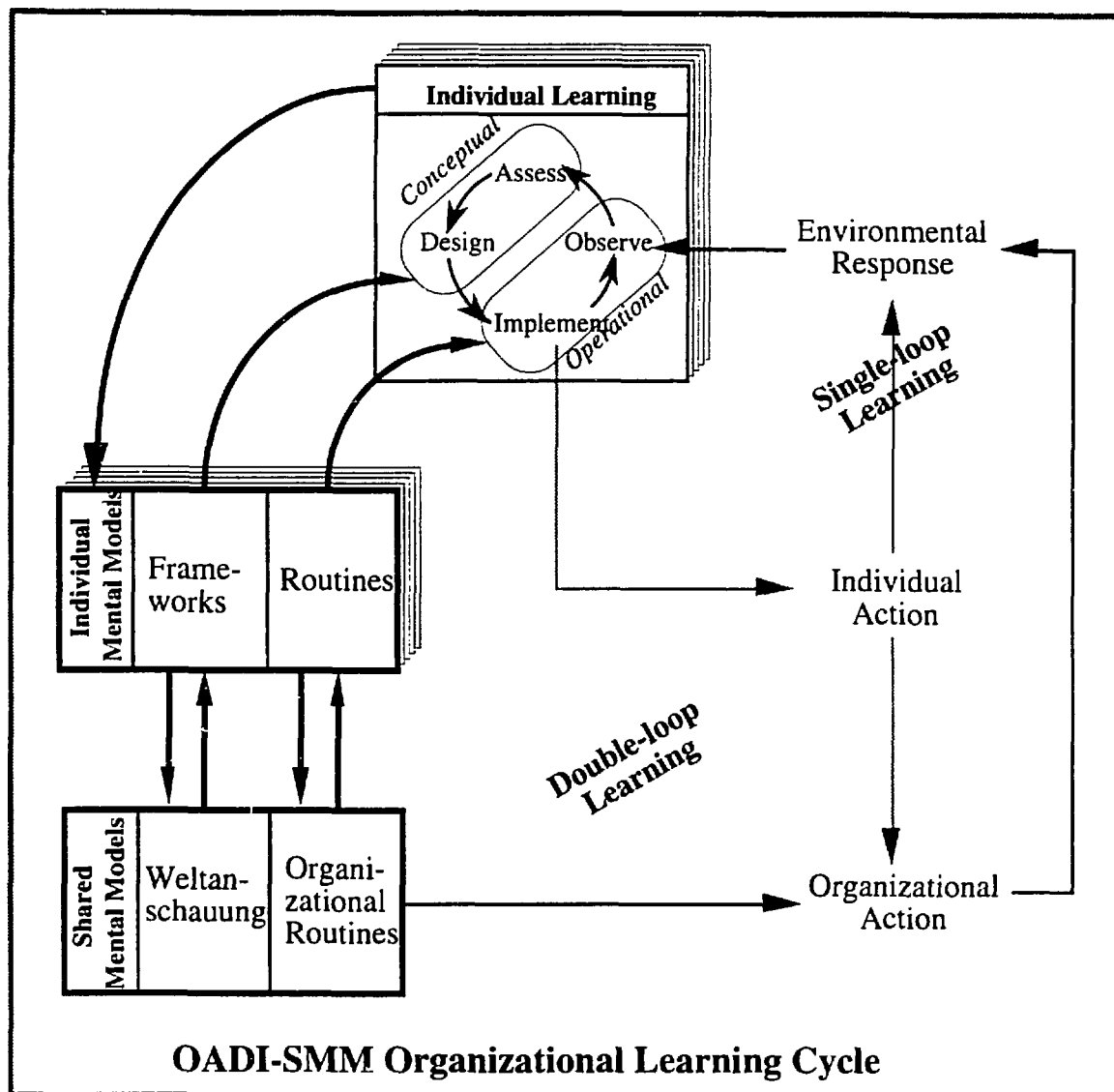
6.0 INTRODUCTION

In Chapter 5, the TQM implementation study focused on helping managers map their experiences into a systemic framework, make their understanding of that experience explicit onto the systemic maps, and then put it into a form that could be explicitly shared with other people. Given the nature of that study, it provided an example of the ability to take individual learning, make it explicit into the mental models, and put it into a form which clarified the experience and made it more easily sharable, much in the same way that K-J diagrams in total quality methodology allow one to construct a representation of one's process and experience in a form that's easily shared.

The purpose of this chapter's study on product development management is to go through the entire organizational learning cycle as

[†]The work described in this chapter was supported by the MIT Organizational Learning Center. Parts of the case work and product development literature review are drawn from two MIT Sloan School of Management masters' theses which the author helped supervise (Giancola, 1992; Roberts, 1992)

described in Chapter 2 (Figure 6.1), especially as implemented in the methodology developed in Chapter 4. This study details the outcome of a new product development team's going through the process described in Chapter 4. The frameworks and the tools enabled team members to look at their product development process differently, and to do that by making explicit their individual mental models.



Double-Loop Organizational Learning

Figure 6.1

In section 6.4, for example, there will be a detailed extended example of how the core product development team went through a process of building a shared mental model as a group, and we'll see the map that resulted from that process. Making individual mental models explicit in a group session began to illuminate a shared mental model. This helped them to clarify potential points of high leverage action, which they could and did take. Implementing these actions resulted in organizational changes which led to apparent improvement in particular performance indices.

In section 6.5, there will be an example which takes the experience of the core team into a "learning lab" environment. Once again, building the systems archetypes helped other component teams to build a shared understanding of issues similar to the ones addressed by the core team. In particular, understanding the Tragedy-of-the-Commons helped to resolve an issue that team members had been wrestling with for many weeks and which they expected to struggle with for many more weeks without resolution.

Two important insights were generated from the field work that were translated into organizational action. One insight, which was discovered during a systems mapping session, was the recognition of the ways in which the organization's own system for keeping the program on time was directly causing it to miss its timing schedule. The second insight was a direct result of working with the Tragedy-of-the-Commons archetype and recognizing the important role it plays in managing the product development process. The Tragedy-of-the-Commons archetype provided a systemic understanding of why a heavyweight program manager with wide authority over the entire product program makes sense in circumstances where individual component groups can destroy a "common" upon which all depend.

We begin the chapter with an overview of product development in general, and a review of the research literature on product development, followed by a brief chronology of the project work at the company site. This is followed by the extended case examples in section 6.4 and 6.5. At the end of the chapter, we examine some conclusions about managing product development (section 6.6) and implications gained from the research findings for further efforts to improve organizational learning (section 6.7).

6.1 BACKGROUND ON PRODUCT DEVELOPMENT MANAGEMENT

Product development refers to the process by which an organization designs and markets a new product or redesigns and improves an old one. Three functional groups, broadly defined, are involved in this process: design engineering, manufacturing, and marketing. The scope and extent of the process vary widely across and within industries, depending on the complexity of the product involved.

6.1.1 Importance of Product Development

A key concern for companies in a wide range of industries is redesigning the product development process to increase the rate of new product introductions, improve the quality of products marketed, and reduce cost throughout a product's lifecycle. As Clark & Fujimoto (1990, p.1) have written, "Developing high-quality products faster, more efficiently, and more effectively tops the competitive agenda for senior managers around the world." Product development has also been highlighted by several studies of national competitiveness (Dertouzos, et al., 1989; Hayes, et. al., 1988). Survey research by Gupta & Wilemon (1990) has shown that 87% of the managers they surveyed felt increased pressure to develop new products faster. Companies who are able to get new products to market sooner and orders

from customers faster may well hold the principal tool for achieving competitive preeminence in coming years (Dumaine, 1989). Thus, product development performance constitutes a key strategic variable for management.

A number of forces are driving the need to improve the product development process. First, product lifecycles are shrinking across a broad spectrum of industries (Roseneau, 1990), which means that new designs become stale faster as product functionality is rapidly surpassed by each round of competitive introductions. Second, as competition becomes more global, there is the dual challenge of serving worldwide markets and fighting worldwide competitors. Ohmae (1990) states that, in the "interlinked economy" of the world's industrialized and newly industrializing countries, products must be introduced almost simultaneously across the globe. Companies no longer have the luxury of extending product lives by pushing old products into new geographic markets, because customers everywhere are demanding similar quality and functionality.

Third, the rate of technological change has accelerated, and the increasing speed of dissemination of technological advances has compounded this effect. As a result, more companies can access new technologies faster, and innovations can be rapidly emulated. This means that advantages based on (currently) superior technology can be short-lived (Gomory, 1989). Finally, markets are becoming more fragmented as marketers identify finer segmentations of customers based on behavior and needs. Such niche markets place greater demands on firms to develop a higher variety of differentiated products in order to hit these smaller targets.

The implications of these trends are extremely important because the opportunity costs of being late to market can be very high. One analysis.

showed that products coming to market within budget but six months late earned 33% less profit over five years versus 4% for those that came out on time but were 50% over budget (Gupta & Wilemon, 1990). Globalization means that firms must be ready to roll out new products on a wide scale and also monitor a larger, more disparate group of competitors and markets.

Peters (1991) sees several imperatives for success in such an environment. "Sluggish" functional units must be broken into self-contained teams focused "on the fragile, fleeting task at hand." Information must flow freely both within the firm and in the company's "extended family" of suppliers, distributors, and customers. Experimentation and rapid decision making must be encouraged while working to reduce blaming and risk aversion. This is the kind of rewiring that an organization must do in order to cope with the new, chaotic environment.

The above changes place great pressures on product development organizations to restructure themselves to become more nimble. For those that meet the challenge even greater rewards are available. Stalk & Hout (1990) cite a number of advantages of shorter development cycles, including higher margins due to premium pricing, lower development costs due to reduced rework, and improved quality due to better design integrity.

Stalk, Evans, & Shulman (1992) suggest that companies that can transform product development into a competitive weapon by iterating rapidly and consistently through the design cycle will have a competitive advantage. In light of the environmental shifts discussed above, how a firm competes may be more strategically significant than where it competes. Thus, a strong argument can be made for the link between superior performance in product development and superior corporate performance.

6.1.2 Product Development and Organizational Learning

As discussed in earlier chapters, in the rapidly shifting economic environment faced by firms today, organizational learning is crucial both to cope with day-to-day fluctuations and to adapt to significant environmental changes. Senge (1990a) defined a “learning organization” not as one increasing its capacity “merely to survive,” but as one “that is continually expanding its ability to create its own future.” This ability requires what he refers to as “generative learning”—learning that “enhances our capacity to create.”

A number of authors have described product development as a learning process or discussed it in the language of learning. For example, Adler, Riggs, & Wheelwright (1989) cite support for learning as a top management priority for enhancing product development capabilities. Maidique & Zierger (1985) developed a model of the “new product learning cycle” that involves three types of learning, each with a different locus. “Learning by using” takes place in the customer base as buyers gain experience with a product. “Learning by doing” occurs as the firm manufactures greater volumes of the product and is based on the theory of the “experience curve” (Henderson, 1972). “Learning by failure” takes place as managers launch successive generations and variations of products into the market and gain a better understanding of the development process and market responses by identifying failure patterns and weak links in the organization.

Meyers & Wilemon (1989), in contrast, direct their attention toward “intra-team learning” and the mechanisms by which it can be reinforced, documented, and transferred. Imai, Nonaka, & Takeuchi (1985) argue that learning is central to Japanese firms’ product development success where the interplay between “learning” and “unlearning.” Learning occurs within and

between “self-organizing product teams” in a highly adaptive and interactive manner. Unlearning, on the other hand, prevents the development process from becoming overly rigid, and provides a mechanism for continuous improvement.

Based on these varied perspectives on the product development process, it is clear that learning can be an important “by-product” of the product development effort, enhancing understanding of both the firm’s markets and its internal processes. Exploring the intersection between product development and learning should provide an extremely powerful leverage point for focused effort to improve organizational performance.

6.1.3 Review of Studies on Product Development¹

This section provides a brief, cross-sectional review of the many perspectives and different research methods used in product development process research. These studies range from best practices and how-to’s that offer advice on how world class firms have organized and managed their product development efforts to empirical studies that are based on well-defined research methodologies and involve the study of a large sample of firms or managers.

6.1.3.1 Best Practices

A number of recent books aimed at managerial audiences offer guidance on how to organize and manage product development efforts (Roseneau, 1990; Rosenthal, 1992; Smith & Reinertsen, 1991). Roussel, Saad, & Erickson (1991) and Stalk & Hout (1990), while not focusing exclusively on the product development process, devote significant time to new product issues. Almost

¹I acknowledge the help of Roger Roberts, a research assistant, who pulled together much of the literature review in this section. For a fuller review, see Roberts (1992)

all of these authors appear to draw heavily on concepts discussed earlier by Imai, Nonaka, and Takeuchi (1985; Takeuchi & Nonaka, 1986).

House & Price (1991) offer a set of metrics for tracking product development projects in order to reduce the development cycle. These measures, used at Hewlett-Packard, include “break-even time” (BET), time until the project generates a positive cash flow, and “time-to-market,” the time from the start of commercial development until the product is released in the market.

While other authors advocate one best way to achieve development success, Krubasik (1988) cautions managers to *customize* their product development approach based on factors related to their specific product and competitive environment. He points to two key variables, the opportunity cost of not reaching the market within a certain time window and the development risk of the project, as determining factors in how resources are allocated and projects are structured and managed. This is in contrast to most of the authors cited above.

Though many of these studies offer helpful models and insights, they have inherent limitations. Best practices examples and anecdotes about how the Japanese do it, for example, typically focus heavily on the “what” of product development and give little attention to detailed analysis of how new methods can be introduced into the complex realities of functioning organizations. Similarly, benchmarking can be a valuable for targeting an achievable level of performance but is not, in and of itself, a viable method for implementing such performance. According to Senge (1990a, p. 11), “[benchmarking] can do more harm than good, leading to piecemeal copying and playing catch-up. I do not believe great organizations have ever been built by trying to emulate another, any more than individual greatness is

achieved by trying to copy another great person.” There is a need for a deeper understanding of the realities of product development in real organizational contexts which is the goal of many of the empirical studies covered in the next section.

6.1.3.2 *Total Quality Management*

Feigenbaum (1991, p.6), one of the founding fathers of the TQM movement, writes that “the most demanding task of managers and engineers will be to change a company’s philosophy from ‘make it cheaper and quicker’ to ‘make it better.’ New product development is one of the critical areas requiring excellence in work process emphasis.” He advocates applying quality principles within the development process.

Rosenthal (1992) also emphasizes the relationship between new product development process improvement and TQM. Focusing on processes that build customer satisfaction, a key tenet of TQM, accustoms the organization to viewing the firm as a network of processes (one of which is product development) instead of as a confederation of functional fiefdoms. Rosenthal also encourages application of the tools and techniques of TQM to the analysis of the development system.

6.1.3.3 *Marketing*

From a marketing perspective, product development can be seen as a process of scanning for unmet market needs, translating them into a design, and producing new products. Traditionally, marketing placed a heavy emphasis on product *concept* development and testing. This can be seen in texts by Urban & Hauser (1980), Crawford (1983), and Pessemier (1986) which devote little attention to the questions of managing the information flows across the firm or of resolving conflicts between functional perspectives.

Fine, Hauser, Clausing, & Sachs (1989) advocate the use of quality function deployment or QFD (also known as the “House of Quality”) for improving the marketing-design interface in product development. QFD addresses some of the organizational issues mentioned above by facilitating interfunctional planning and communications by starting from the premise that interfunctional teams can improve the development process (Hauser & Clausing, 1988). The mapping methodology facilitates communication between marketing, design, and manufacturing because it serves as a mechanism for transmitting the “voice of the customer,” recorded in terms of desired product attributes, and translating these attributes into the quantifiable language of engineering characteristics. Empirical studies on the effectiveness of QFD appear to show improved inter-functional communication (Griffin & Hauser, 1990).

6.1.3.4 Organizational Behavior

Product development can also be viewed as an organizational process that involves the resolution of conflict between representatives of various functional cultures or constituencies. The focus here is on how effective teamwork can be fostered in a process that cuts across the various turfs that delineate the political power structure of an organization.

One of the earliest attempts to examine product development from an organizational behavior perspective is based on the research of Lorsch & Lawrence (1965). Their study of the processes of specialization and coordination required to generate and market innovations showed that while specialists are needed to increase the efficiency of development tasks, coordinating mechanisms (of which there can be many forms) are crucial to integrate the views and work of specialists. Dougherty’s (1987) study of the

interaction of functional units as they attempt to comprehend the market for a new product emphasized the different ways in which functional units think about the market—each seeking disparate information and interpreting the information differently. She coined the term “*thought world*” to describe the distinctive views of different organizational constituencies, who institutionalized their own routines and reinforced the distinctions that would keep them separate and prohibit creative learning.

In a similar vein, McDonough & Leifer’s (1986) analysis of multiple cultures within firms engaged in product development projects revealed that, although a degree of creative tension is valuable among those responsible for innovation, strong project leadership is needed to reconcile the cultural clashes that result. Rubenstein, et. al., (1976, p.20), in their study to understand the factors influencing innovation success at the project level, concluded that one of the two areas on which organizational re-design efforts should focus is in improving “communication in terms of frequency, openness, timing, quality, context, and content between many pairs of company functions involved in the [product development] process.” This emphasis on communication is similar to the need to develop shared understanding between Dougherty’s multiple thought worlds.

A common prescription for improving the product development process is the use of cross-functional teams. Ancona & Caldwell’s (1992) assessment of the impact of diversity in both tenure and function on cross functional team performance produced mixed results. They found that diversity helps in setting goals and priorities and maintaining external communications, but diversity impairs the internal processes of teams in a manner that outweighs the external benefits. They suggest that changes in team training and

evaluation as well as in organizational norms are required to realize the potential benefits of teamwork.

6.1.3.5 *Operations Management*

Studies that focus on the mapping of flows and the examination of processing stages are classified under the heading of *operations management*. Rosenthal (1992, p. 7) sees the development process as a “series of decisions that combine to transform an initial [conceptual] product into a physical reality.” Larson & Gobeli (1988) found that a project-biased matrix product development structure yielded the best results for the firms in their sample. Allen’s (1977; 1980) pioneering studies of communication patterns in R&D groups showed that the level of communication drops dramatically if team members do not work in close proximity, providing strong support for the concept of co-located teams.

Others have studied the use of Design-for-manufacturing (DFM) guidelines on product development and production system performance for high-volume products in time-critical markets. By reducing part complexity and shortening tooling procurement times, deviating from DFM guidelines can actually improve performance (Ulrich, Sartorius, Pearson, & Jadiela, 1993). Eppinger et al. (1990) describe a process for reducing the overall complexity of a project by reorganizing the relatively few tasks that are critical to the project. They use a design structure matrix method for strategically decoupling major design tasks that require more manageable design team sizes (Gebala & Eppinger, 1991).

Perhaps the most extensive and detailed research on product development in the automotive industry is contained in Clark & Fujimoto (1991). The authors studied projects in 20 firms over a six year period, basing

their work on a view of product development as a “process by which an organization transforms data on market opportunities and technical possibilities into information assets for commercial production” (Clark & Fujimoto, 1991, p. 20) .

6.2 PRODUCT DEVELOPMENT MANAGEMENT CHALLENGE

Despite all the research done in the areas mentioned above, product development managers have a lot more questions than they do answers. As one product development manager put it:

“We have the technology, the people, the skills, all the pieces—why can’t we put it all together?”

If one agrees with the premise that they indeed have all the pieces, then the problem lies in how those pieces are managed. Clark and Fujimoto (1991) underscore this point repeatedly:

1. Introducing cross-functional communication tools in such an [adversarial] atmosphere (which we found in many other companies as well) might easily reinforce the traditional separation of functions. For example, requiring product engineers to release preliminary information to process engineers will make design fluctuations more apparent to process engineers. If process engineers react by blaming product engineers for the changes, the product engineers will become increasingly defensive. From their perspective, early communication to process engineers only opens them to earlier attack and gives process people opportunities to unilaterally impose design constraints in the name of manufacturability. This may reinforce a “don’t-tell-them-early” attitude on the product side. On the process side, the notion that early design information, because it is likely to change, does not deserve serious consideration may reinforce a wait-and-see attitude. (Clark & Fujimoto, p. 125)

2. We have observed that technical expertise in a variety of disciplines is essential to developing an outstanding product rapidly and efficiently, but that *even more important* [emphasis added] is the way expertise is applied and integrated. Firms face a variety of choices about structure, procedures, assignments, and communication. Effectiveness appears to be a function of consistency and balance in managing the critical linkages within and across the stages of development. (Clark & Fujimoto, p. 127)

3. Engineers, tending to be perfectionists, are often reluctant to release work that is incomplete. The upstream group will be even less willing to release information early if the environment is hostile, with design changes triggering accusations of sloth or incompetence. If the attitude of product

engineers is “I won’t give you anything now because I know I’ll have to change it later and I know that I’ll take the blame for it,” management may have to effect a fundamental change of attitude throughout the engineering organization, both upstream and downstream, a very difficult task. (Clark & Fujimoto, p. 213)

4. Introducing overlapping without requisite changes in communication, organization, and management is more likely to reduce product quality, incur unintended schedule delays, and lower morale in the engineering organization than to improve performance. (Clark & Fujimoto, p. 238)

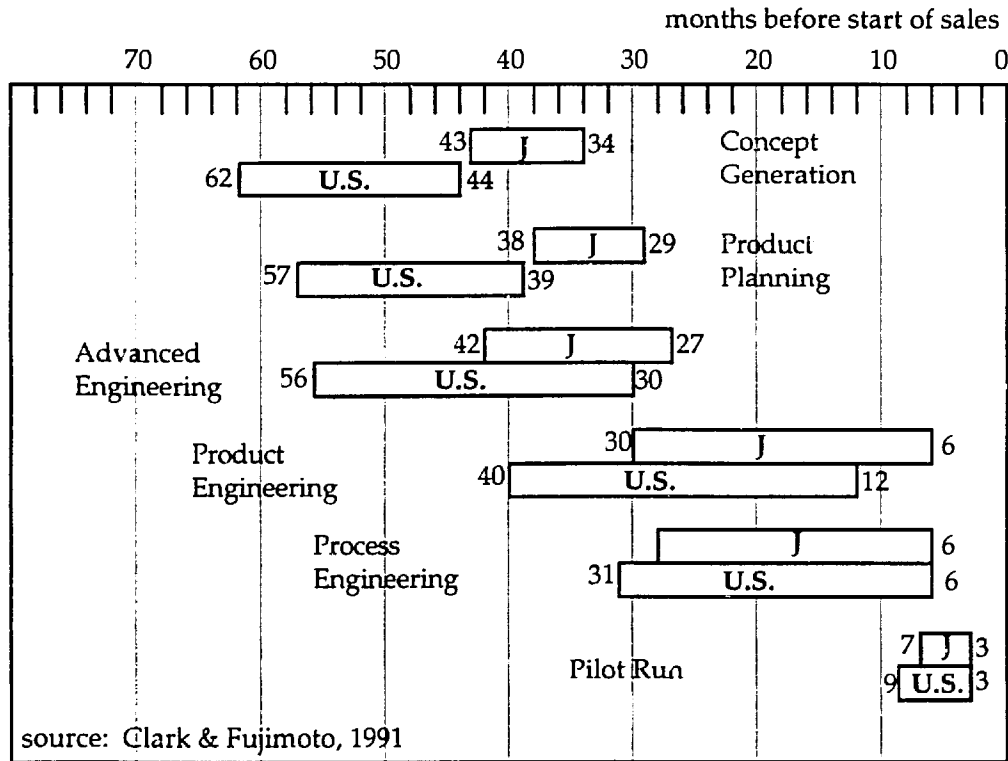
The above statements suggest that one of the greatest barriers to better product development performance lie in the *organizational structure and systems* in which the process must be managed. Even when an organization has all the component technologies and expertise necessary, performance can be greatly diminished by the way it is managed in concert. It is not enough to have the product development manager understand how all the pieces are to be managed; the whole engineering organization needs to understand and agree to work together effectively. That is, they all need to be thinking globally and acting locally (Sterman & Senge, 1991) .

To achieve the “fundamental change of attitude throughout the engineering organization,” that Clark and Fujimoto write about will require building a shared understanding (or shared mental models) of what the product development process is about in a way that allows people to connect their individual activities to the whole.

6.2.1 Focusing on Product Engineering and Process Engineering

Product development is an incredibly complex process which requires the managing of multiple constraints and demands spanning a relatively long time horizon with multiple phases. Figure 6.2 shows Clark and Fujimoto’s (1991) summary results of “average project schedule by stages.” Compared to the U.S., Japanese automakers have a substantial advantage in their ability to get products more quickly to market. The Japanese appear to undertake more

of their stages in parallel than their US counterparts. In particular, the product engineering and process engineering stages are conducted almost completely in parallel resulting in a 10 month advantage over the US in those two stages alone, and the pattern is similar in the other stages as well.



Average Project Schedule by Stages

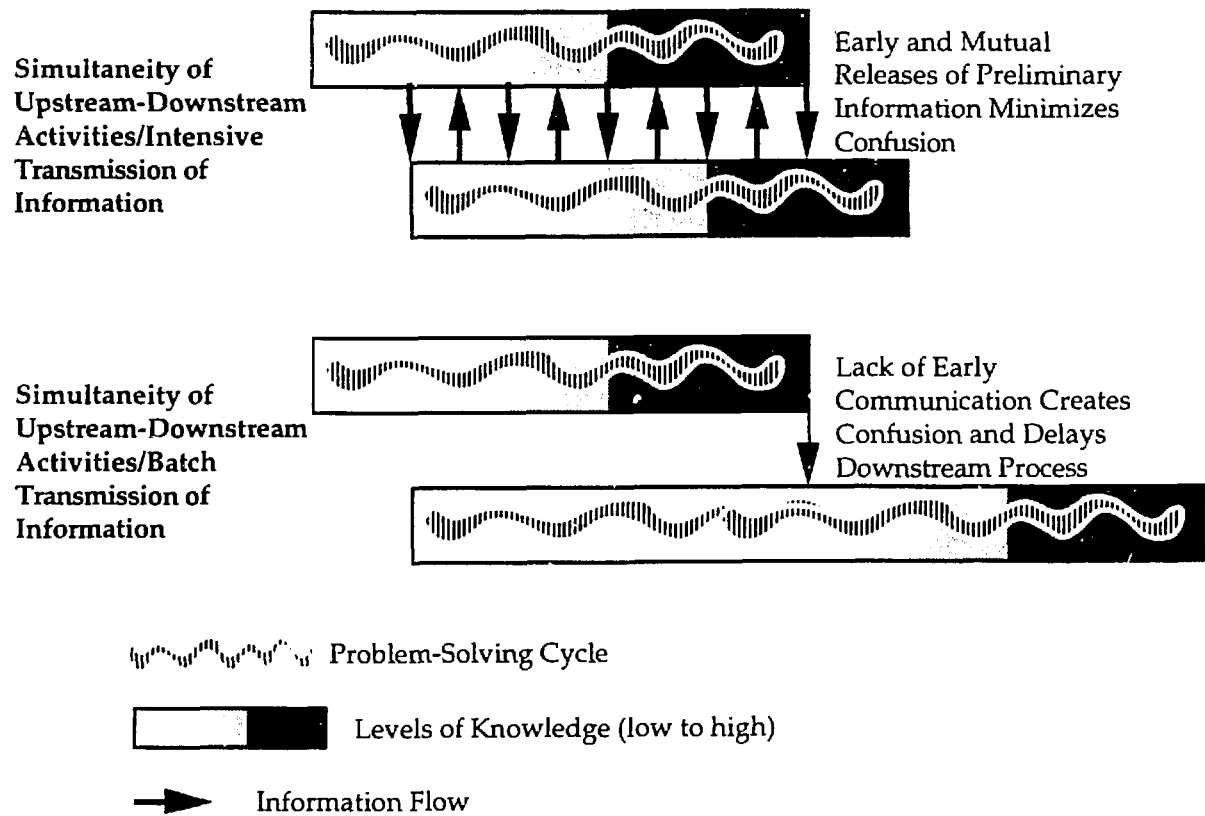
(source: Clark & Fujimoto, 1991)

Figure 6.2

As Figure 6.2 shows, the Concept Generation and Product Planning stages have virtually no overlap with the Product Engineering and Process Engineering stages. The Advanced Engineering stage does have some overlap, but we may be able to assume that the requisite advanced engineering is available to us when the product engineering effort gets underway. The product engineering and process engineering stages are likely to exist in many organizational settings outside of the car industry. Success in

these stages is also more likely to depend less on technical expertise and more on managerial competence than the earlier phases.

6.2.2 Focusing on Overlap/Coordination



Overlapping with and without Intensive Communication

(source: Clark & Fujimoto, 1991)

Figure 6.3

Clark and Fujimoto also cited the importance of simultaneity of upstream-downstream activity in *conjunction* with continuous sharing of information. One can conduct upstream-downstream activities at the same time but achieve very different results depending on how one went about sharing information throughout the process (see Figure 6.3). As quoted earlier, encouraging overlapping without changing the way communication is handled can potentially be less productive and more costly than doing things more serially.

What allows one organization to be able to implement and manage according to the top model while others struggle on in the bottom model? What keeps organizations who understand the value of the more productive model from following suit? These are the kinds of questions that are raised by the reported results. One approach to addressing such questions is to begin looking at how the current system has a natural tendency to reinforce its current status.

In this project, the maps and methods described in Chapters 3 and 4 were used to help line managers understand the reasons why problems recur when, in principle, team members already “know what to do.” Evidence shows that many firms are in this situation. Gupta & Wilemon’s (1990) survey data indicate that while 88% of managers face increasing pressure to accelerate development cycle, 87% report that, despite increased attention to product development, the problems of the past persist in their firms. As Meyers & Wilemon (1989) point out, this persistence of error is an indicator that learning is not taking place.

As a manager in the study said, “There is a lot of stuff out there that will tell you *what* to do, but not much to tell you *how* to do it.” Improving the rate of learning is a key to shrinking product development cycles and to improving the quality of the development process. Our focus in this study is thus placed on helping managers to “learn to learn,” and thus to identify, understand, and remove the obstacles to learning and change

6.2.3 Product Development and the OADI-SMM Learning Cycle

The OADI-SMM cycle in Figure 2.9 can serve as a framework for advancing organizational learning in a complex organizational setting like product

development. Managers at Cars² agreed that they traditionally have operated in the Assess-Implement mode and have not invested much effort in the Observe and Design aspects of the learning loop. At Cars, there was little learning that was transferred from one product program to the next. In terms of the organizational learning model, there was no formal process for capturing individual learning into explicit individual and shared mental models that would lead to new organizational actions. In effect, they did not have a product development process that was continually accumulating learning from each product development experience; they had a long series of isolated individual product development efforts.

In our work with Cars, we focused on building the managers' skills to tease apart assessment from observation. By building capabilities for reflection and inquiry, it is possible to encourage managers to balance inquiry (their new role) with advocacy (their traditional role).³ Making managers more conscious of their "bias toward action" helped them to be more aware of when they were jumping from assessment to implementation, and decrease their tendency to short-circuit the loop.

By recognizing the influence of mental models on what it is that people observe, and by bringing to light the ways in which "leaps of inference" and hidden assumptions can shape their assessments, it is possible to use the OADI cycle to reason from directly observable data, to assess the nature of systemic structures, and to identify and design high leverage improvement actions, leading to further observations of the effects of implementing such actions. The end result is the creation of an organizational capability to make

²This is a pseudonym used for the company under study.

³For a discussion of "Balancing inquiry and advocacy," see Senge (1990b), pp. 198-202.

positive changes in systemic structures and thus advance organizational learning in product development.

6.3 CONSUMER CARS MANUFACTURER—CARS, INC.⁴

6.3.1 Company Background

Cars, Inc. is one of the largest manufacturing companies in world. With customers' more sophisticated demands for quality and styling, Cars re-examined its product development process in an effort to better meet the market's new challenges.

Various parts of Cars experimented with different approaches, including world-class timing, to improve the product development process. One product development team, the APDP (A Product Development Program) team, chose to participate in a MIT Organizational Learning Center (OLC) pilot project in order to understand their product development better and use it as a means to build an ongoing learning process within their product development organization (see Appendix C for a brief description of the OLC research objectives).

6.3.2 Project Chronology

In late 1989 and early 1990, Cars' program for its top 2000 worldwide executives addressed crucial issues on the Chairman's desk. When the program ended, it became clear that another round of development programs would be needed. The series left executives intrigued but unclear as to how the new ideas might be applied. The Executive Development office was challenged to find a way to operationalize systems thinking throughout Cars.

⁴For a more detailed description of the process of conducting the research partnership with Cars, Inc. as well as the details of the meetings held, see Giancola (1992).

In 1991 a diverse group of Cars managers attended a series of five two-day core competency courses on systems thinking and related learning disciplines at the OLC. Seeds were planted, and some took root. Engineers in product development wanted a “deeper dive” into the learning, envisioning a difference in the way they made decisions in their new product development cycle if they could become familiar with new learning tools such as feedback loops and causal diagrams.

Interviews conducted separately with the Program Manager and Planning Manager in charge of APDP revealed differences in how they saw the source of problems, and their ideas about how to best effect solutions. The Planning Manager, on the one hand, wanted the OLC project to focus on improvements in the Evaluation Prototype (EP) in the belief that if the EP works, the new product is likely to work. Although Cars has a very specific process, they have never followed it in developing a product.

The Program Manager, however, believed that concentrating on EP would be too short-term a solution, and would create backlash problems in the future. He felt what was needed was to take a deeper look at Cars’ entire manufacturing paradigm, currently based on resource control and restraint and the engineer design validation process, which he sees as over-involving senior management, treating suppliers with indifference or hostility, and greatly reducing the authority of operating personnel to meet their quality, cost, and time objectives.

The APDP core team had its first 1-1/2 day planning session in January 1992, with the Program Manager, Planning Manager, Engineering Manager, Manufacturing Manager, Finance Manager, Purchasing Manager and Internal

Consultant from Cars and the OLC research team⁵. These sessions, held roughly every four to six weeks, were intended to help the team “learn about learning.” Over the course of several sessions, the team members were exposed to the problem articulation approach and mapping methodology described in Chapter 4.

The team members worked with ladder of inference, left-hand/right-hand column tool, action maps, systems archetypes, and affinity (or KJ) diagrams. The team members were trained to conduct interviews of others in the APDP program in order to gather additional directly observable data from those outside of the core team. The team constructed numerous systems archetypes based on their own experiences and the data gathered from the interviews.

6.4 MAPPING SHARED MENTAL MODELS⁶

In Chapter 4 we laid out a 10-step methodology for mapping individual mental models and for transferring them to others. In this section we will go through an extended example of a systems mapping session as an illustration of how the core team was able to make their individual mental models explicit and integrate them into a collectively constructed map around a specific issue. Although the Cars core team followed the general steps outlined in Chapter 4, we also made some variations. First, since this team was composed of the line managers who were actually responsible for the car program, the independent researcher coding process was not used. Instead, we created the variable names in real time with the managers. Second,

⁵The OLC team consisted of Daniel Kim and Paula Giancola, except for the first meeting which also included Bill Isaacs. Don Seville joined the OLC in August, 1992. Paula Giancola left the team in May 1992.

⁶The mapping session contained in this session are drawn from a transcript of a meeting held in August 1992.

interviews of others on the car program were conducted by the core team members and analyzed as a group. Third, the integrated diagrams were also created in real time by cycling through steps 1-4 in rapid succession. The map that resulted from this process produced new insights for the group and led to identification of high leverage actions that produced much better results for the APDP program.⁷

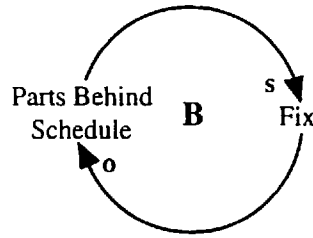
The core team members have been identified as managers A, B, etc. to help preserve anonymity. The author is identified as MR (MIT Researcher). Portions of the conversation that are underlined indicate variables that are relevant to the diagram being constructed. Portions that appear in **bold** are insights or process observations made by the managers. The presence of “•••” means that portions of the conversation have been deleted. My commentary is in *italics*.

6.4.1 Focusing on an Issue: Parts Behind Schedule

The session began with a discussion about what issue to focus our efforts on, given where the APDP program was in its schedule. The stated goal was to try to map what was intuitively known in various pieces by different people and make explicit the interconnections between the different pieces and how different actions that we take can create differing short-term and long-term consequences. The starting point was with a Fixes-that-Fail systems archetype.

[Mgr. A]: ...I agree with [Mgr. B] on that late parts being one that we can still influence. I took the one that adds in addition, a different version, they're basically the same, but then I took another version of it that's different, that adds in the supplier impact, which is exactly what we're doing today, putting the heat on the suppliers for 7-day weeks, 24-hour days...

⁷Throughout the APDP pilot project, the core team met on a monthly basis, more or less. During these sessions, a number of causal maps and systems archetypes were constructed. The following is a detailed account of one of the sessions.



Parts Behind Schedule

Figure 6.4

6.4.1.1 *Supplier Timing*

It is interesting to note that the suppliers are the first thing the managers think of when we focused on the problem of late parts. Later, they revise the sequence and say that the first thing they try to do is get more engineering resources. Both are considered to be typical “knee-jerk” responses to parts being late.

•••

[Mgr. A]: Well, [the] supplier will tell you what compromises he’s making.

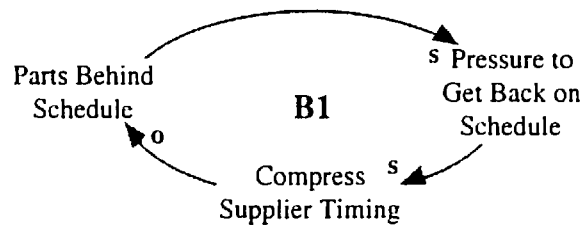
[Mgr. B]: He’s saying you’re pushing it off on me, and then manufacturing is going to say you’re pushing it off on me, which means I have less capability to evaluate the process. We’re shifting the burden from us who make decisions to the engineers who release the parts late, to the suppliers who have to...

[Mgr. A]: But the supplier tells you, we sat at a meeting where we were told [B] said they could ship the first shot without this data, and we said we don’t want you do that. So, they tell you what those timing shortcuts are. They’re observables.

[MR]: So, for instance, if we said, parts behind schedule and there’s this pressure--what do we do with that pressure? If we mapped one of them...

•••

[Mgr. B]: We compress the supplier, and the next thing that’s likely to happen is we’re gonna compress the builds somehow, we re-adjust the sequence.



Compress Supplier Timing

Figure 6.5

6.4.1.2 Build Timing

In this segment, we begin to see that there is some confusion about what “compress build timing” really means. Section 6.4.2 contains an example of another point of confusion around compressing development timing. In both cases, having the causal loop diagram and explicitly drawing the links appeared to keep the group focused on clarifying each variable being added to the map.

[Mgr. A]: Or we delay the build and reduce the time to function part.

[Mgr. F]: Does compress mean overtime?

[Mgr. A]: It either means overtime or it means to take the units that you were going to build if these parts are late for and put it later in the production schedule. If the moon roof is late you try to build cars without moonroofs first. We just did that. We delayed the first moon roof by five weeks and pulled other cars up in the lineup so we could handle the late part.

...

[Mgr. F]: Short cut, don't do testing,

[Mgr. A]: That's where you can compress the actual functioning.

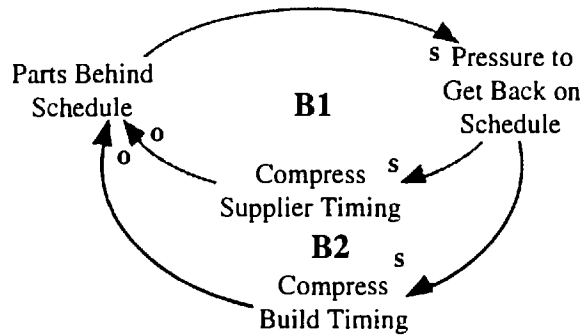
[Mgr. F]: I would say compress means do it faster, this is...

[Mgr. A]: Compress build or function, if you can't compress the build anymore you reduce the amount of time you have to function. Effectively to delay the prototype.

[Mgr. F]: To me, compress build means do the same thing faster.

[MR]: Well, compress build timing says you shrink the time, but the question is, and your intention is to get back on schedule, but if you point out with the supplier, one of the things is that if you have to take anything out, they'll gonna take short cuts. They'll tell you what...

[Mgr. A]: Well, why don't we make it compressed build or development.



Compress Build Timing

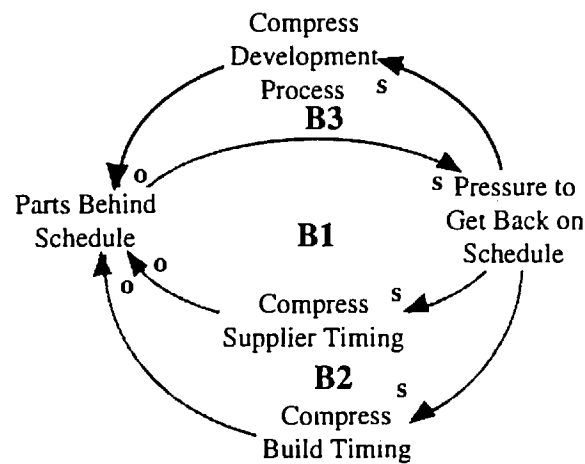
Figure 6.6

6.4.1.3 Development Process

[Mgr. D]: Or make it a separate thing.

[MR]: They're very different effects.

[Mgr. D]: Compress development timing is another way to get back on schedule.



Compress Development Timing

Figure 6.7

6.4.1.4 Delay Program

[MR]: What other actions do you take to respond to the pressures to get back on schedule?

[Mgr. A]: Well, the obvious last one is you delay the Job l.

[Mgr. F]: Would working over time be one?

[Mgr. A]: That's compress. Compress means all-out overtime.

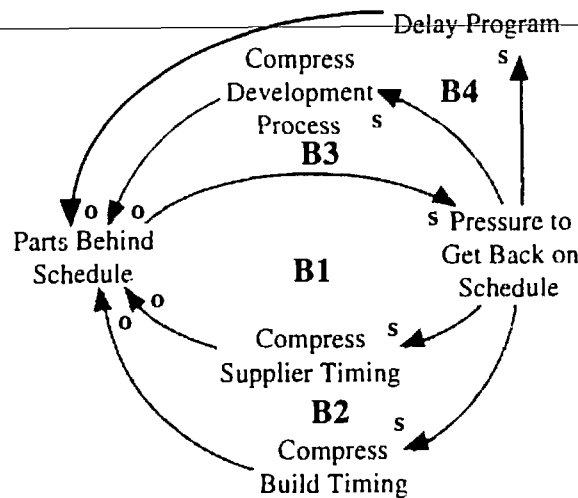
[Mgr. B]: What else we're going to do is we're going to throw more engineers to do the job faster, compress the engineers. If you find you're behind schedule because of parts, because engineering's not getting done fast enough, you're gonna throw more engineers in there to complete it.

[Mgr. A]: But that isn't helping.

[Mgr. B]: That's what we're doing on [program Y]. We allocated resources on a dramatic basis, and that could happen to us if we find out we're in trouble. Delaying the program is the easy solution. But the consequences...

[Mgr. A]: But it belongs on the loop.

[Mgr. B]: You're right. It belongs there.



Delay Program

Figure 6.8

6.4.1.5 Engineering Resources

[MR]: Any other ?

[Mgr. B]: Increase engineering resources to get the parts out faster. Or get the engineering done faster.

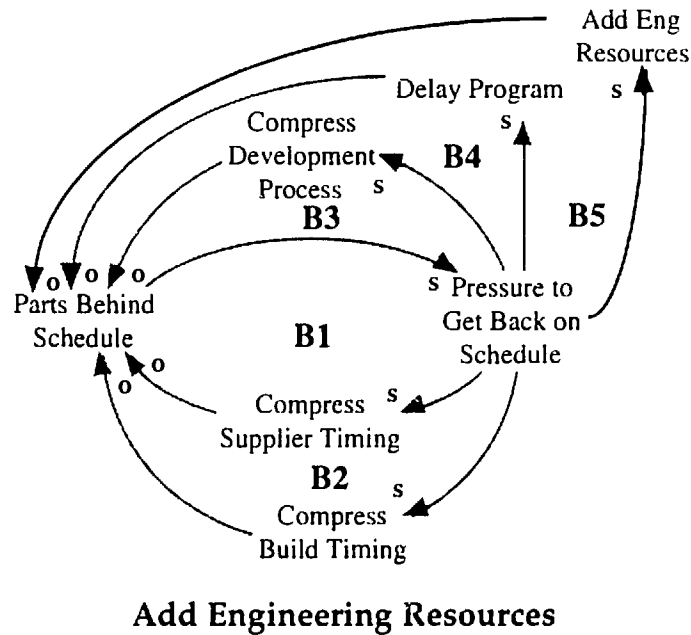


Figure 6.9

...

[Mgr. B]: That's probably the first thing you say, when parts are getting done late, they're going to be late, so can we catch up by adding engineering resources?

[Mgr. C]: And if that doesn't get to you, then you do the next thing, which is compress.

[Mgr. D]: Another thing you could do is reduce content.

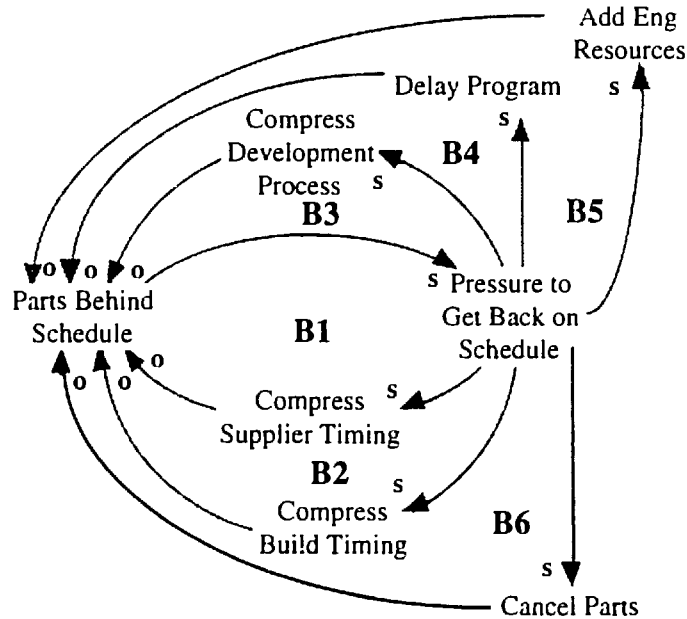
[Mgr. A]: Actually, the first thing we do is compress the supplier time. That's the way we react.

6.4.1.6 Cancel Late Parts

...

[Mgr. F]: [Mgr. D] said, reduce content is another option.

[Mgr. A]: In this specific map, it's not reduce content, it's cancel the late parts. That's basically what we do by reducing the content. The only reason you're reducing the content is cause you can't get it there in time, so what you're doing is canceling the late parts.



Cancel Late Parts

Figure 6.10

...

[Mgr. C]: That works with some parts. You can say, I'm 17 weeks late with the wood, I'm not anymore, but I was, for the console, instead of kicking prints and beating up the surface layout people, which is what we did, I canceled the wood, and say I'm going to go...

[Mgr. C]: But if it was door trim that was late, you wouldn't cancel the change in your door trim, because you really can't. You still have to do the work.

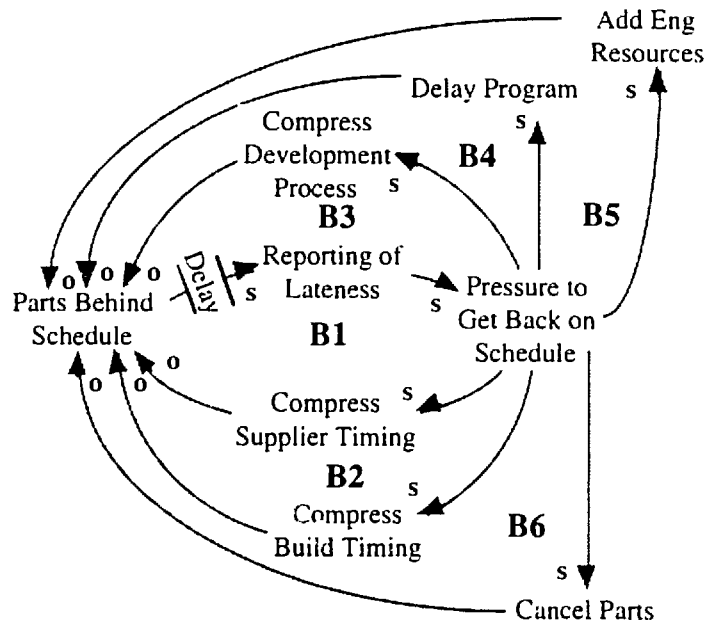
[Mgr. A]: It only works with some parts. They don't have to be optional, but it doesn't work with everything. You've got to have an engine. You've got to have wheels.

6.4.1.7 Information Delay

...

[Mgr. C]: There's one thing that doesn't come out there, and that's when you first admit that the parts are behind schedule. The thing is, you get to the point where you have to compress the product timing an everything because I don't know if there's actions you could do beforehand, but the fact that the parts are behind schedule, does not necessarily come out on a timely basis.

[Mgr. A]: [Mgr. C]'s dead right. If you put a delay right there between parts behind schedule and the first time you do something about it, that delay is actually going to be called here.



Information Delay

Figure 6.11

- [Mgr. C]: Or even right before the part's behind schedule there's a delay, there's something that...
- [Mgr. F]: There's a delay but it's also a loop, something that's going on, that you might want to explain that ultimately creates the pressure.
- [Mgr. A]: There is an unwillingness to admit that you blew it which then further reduces the amount of time you have to take action on it. She's right. That is a critical leverage point.
- [Mgr. C]: Because there may be some things that you can do, I don't think we've ever found what those things are, because by the time we finally get to this point, those are the only things we tend to do.
- [Mgr. D]: Recognition of parts behind schedule is what you don't have.
- [Mgr. A]: So there's a delay before you compress.
- ...
- [Mgr. A]: ...Two [meetings] ago, the fashe team came in and said, the fashes are one week late. One week later, last week, fashes are six and a half weeks on the front and eight weeks late on the rear, and I said, wait a minute, in one week, with no changes, nothing happened in that week to the fashes, I lost five and a half weeks on the front, and 7 weeks on the rear. I don't believe it!
- [Mgr. C]: [A person] did the same thing on his door trim panel. He went from one week to fifteen weeks.
- [Mgr. A]: That's the delay. They knew they were really late, and didn't want to say anything about it.

•••

[Mgr. A]: One of the reasons they delay, is the hope that somebody else is going to come in and report they're even later, and reschedule the build.

•••

[Mgr. C]: ...it's kind of like why go into all this stuff if maybe the first thing that we have to do is figure out why there's this delay in the reporting of lateness? And if there's some way that you can attack that. That is our leverage point.

[Mgr. A]: There are other leverage points.

[Mgr. F]: But if you want one that you could have an impact on,

[Mgr. A]: This points out even when you know you think you know what that system is, I could practically draw that in my sleep, still sitting here today we came up with a new insight.

6.4.1.8 Summary

The insight that Manager A is referring to is about the identification of the delay in getting lateness information from the engineers. The significance of this is not the fact that the delay was identified because that had already been known from the very beginning of the project. That data had come out in the preliminary interviews as well as in the interviews with those outside of the core team. *The significant insight is that they now saw it as a leverage point.* That little piece of information which had been an undistinguished part of their ocean of information suddenly glistened and caught their eye because of the implications *made explicit in the causal loop diagram.*

This realization led them to see more clearly the reinforcing dynamic of how people play the waiting game when they are late with their part. As one engineer we interviewed said:

If you make a mistake, people will jump all over the mistake, highlight it. People I work with are motivated to make it look like they don't make mistakes. Important to look like you know what you're doing, whether you do or not.

As the data from earlier interviews support, there is a tremendous reluctance to be the first bearer of bad news, therefore everyone waits for some one else

to be the first to be “discovered.” This *delay* has enormous consequences for the program because everyone else has to continue working under the assumption that everything is on schedule, rather than making the necessary adjustments as soon as they are required.

6.4.2 Clarifying a Point through Mapping.

There were times when people were confused about what was meant by one thing or another. The following segment shows how the mapping process helped to surface and clarify what is meant by “compressing development.” One reason it does this is by having a common picture of the conversation that is continually accumulating throughout the process so that inconsistencies and unclear points do not simply “disappear” once you have discussed them. Another reason is that the map also makes explicit the assumed linkages in an unambiguous way, making it easier to spot inconsistencies or lack of clarity.

[Mgr. A]: That’s compressed development, because effectively you build a [APDP] not *the* [APDP] you wanted developed.

[Mgr. B]: Compressed development I thought meant that instead of having three months to evaluate your prototypes you’re going to have two months.

[Mgr. A]: Well, for the part that’s late, compressing development could mean I’m going to delay it’s availability on the APDP, could mean I’m going to use something that’s not representative.

[Mgr. B]: That’s not the way I see it. We have to clarify that. Compressed development means you’re going to have less time to evaluate the parts, it doesn’t say anything about the quality of the parts you’re evaluating.

[Mgr. A]: What about I could have reduced time, or I could have no time.

[Mgr. B]: Yeah, *you* make that assumption, *I* didn’t. You see what I am saying?

[Mgr. A]: Yeah.

[Mgr. B]: I guess what we’re saying is instead of getting the part that’s supposed to be fully representative of Job 1 levels at [EP, we’ll say, build that with a surrogate process that’s not fully production representative, and the part isn’t of the materials that you would have at Job 1, so therefore, okay it’s a compromise part. See? We’re

fighting the battle now. And we're going to fight that battle at EP, and we're going to fight that battle at functioning when the engineer says let me give you some sort of a part so you can build the car, but...

•••

[Mgr. B]: Yeah, but we're in two months gonna be faced with an issue. Do you build on time, or do you accept compromise parts in certain areas? We will accept compromise parts in certain areas, which means, I don't care what we say, it's not the same quality as we wanted it. How do we capture that?

Because what the engineers are thinking at the back of their minds is "sh-, I'm not going to get my part there, I just know I'm not going to get that part there, I've got to find a substitute part."

•••

[Mgr. D]: And the manufacturer, the supplier, is not developing his process, is not becoming capable, is not reducing variability, because he's not even into the process yet. He's using a substitute.

[Mgr. B]: He's given you a substitute part in order for us to build.

[Mgr. A]: So what we're compressing is both product development and manufacturing development.

•••

[Mgr. A]: If we're going to go about it that way, I would change compress development, and I would call it instead delay prototype build.

[Mgr. D]: They think their job is to take care of scheduling problems. They try to do it themselves before they get overwhelmed and then all of a sudden, they're overwhelmed, they're late, and then they report.

[Mgr. B]: Then we get to a stage where building the prototype is more important than the quality of the prototype, meaning, "I've got to build a damn car. I've got to tell management we've built the car. I've got to tell management it's on the road, okay? They'll never know what I put into the car. I can fix this."

[Mgr. A]: That isn't the logic flow. Yeah, that's true, but it's not the logic flow. The logic flow is "I got thousands [of] tests I can be running if I built the prototype, this is just one of them, so I'll leave all the tests associated with this out. It's non-representative."

[Mgr. B]: He says, I'll be testing my component separately anyway, so I'll just give them something to put in the car. What happens is on the prototypes (a) we don't know what the level of the parts are; (b) we don't know what the design of the parts are, and (c) we don't know whether the parts are any good. Okay, you try to ride a crcr against a part, and the engineer suddenly says, you can't do that because it's not a representative part.

[Mgr. A]: That's use of substitute parts,

[Mgr. F]: Is delaying the prototype the same as delaying the program?

[Mgr. A]: You can delay prototype building, compress the amount of time you have to develop.

[Mgr. F]: So you're saying the only place you're talking about the quality if you will of development is in the substitute part. That loop takes care of all the concerns.

[Mgr. A]: No, if I delay the prototype parts, then I risk the quality of the program, because I no longer have as much time to develop the vehicle. The reason I put the cars on an EP is so I can find problems with the whole car. These parts interact in a total vehicle, so by using a substitute part, I reduce the amount of time I have for developing that part, but by delaying the prototype, I've got a representative prototype, but I've reduced the amount of time I have for development of all parts. So usually, we go the other way, we use a substitute part, but we build. Unless everything is late, or something real basic is late, like missing an engine.

...

[Mgr. B]: That's compressed build timing. That's where we revise [the] build schedule. This is an example of our being real clear on what we mean.

The value of having the causal map as it was being built became particularly evident through the above discussion around what was meant by "compress development." Without the diagram, there is no common, coherent structure to refer to as the discussion takes place. With the diagram, in front of them, they saw the balancing and reinforcing loops and how the various pieces were logical when viewed in within the context of the system. They realized that the loop structures made sense for the players and that's why the dynamics continue to occur.

In continuing to map, they saw how much of their efforts were expended *responding* to the late parts problem and how the late part problem worsened as a result of those actions. For example, when the suppliers' timing is compressed, the suppliers decrease the time they spend helping Cars, Inc. engineers, which slows their progress and leads to more parts falling behind schedule (see R9 in Figure 6.13).

6.4.3 The Completed Systems Map

The mapping session followed a general set of steps. First, we started with a problem symptom of interest (parts behind schedule), which then led us to identify all the actions that are taken (e.g., compress supplier timing, revise build schedule) to solve the parts behind schedule problem. But, there are

some side effects of taking any one of these actions which can make other parts late and increase the pressures to take more actions (e.g., “Supplier Time to Help Engineering”). Next, we identified the loops that create the problem symptom by looking upstream of the symptom by asking “what are the sources of the reasons why the symptom is coming up in the first place?” We then looked for the interconnections between the side effects and the problem creating loops. Next, we tried to identify leverage points that will have high impact for relatively low effort (e.g., reporting of lateness) which is covered in the following section.

Excerpts from the session that are linked to the additional pieces of the causal map are included below. The full diagram is shown in Figure 6.13.

•••

[Mgr. B]: Another thing is engineering error. That has been the cause of some of our biggest concerns on the [APDP] is just plain errors.

[Mgr. A]: And I would argue that changing the part you're probably introducing more errors. If you increase changes, you increase errors,

•••

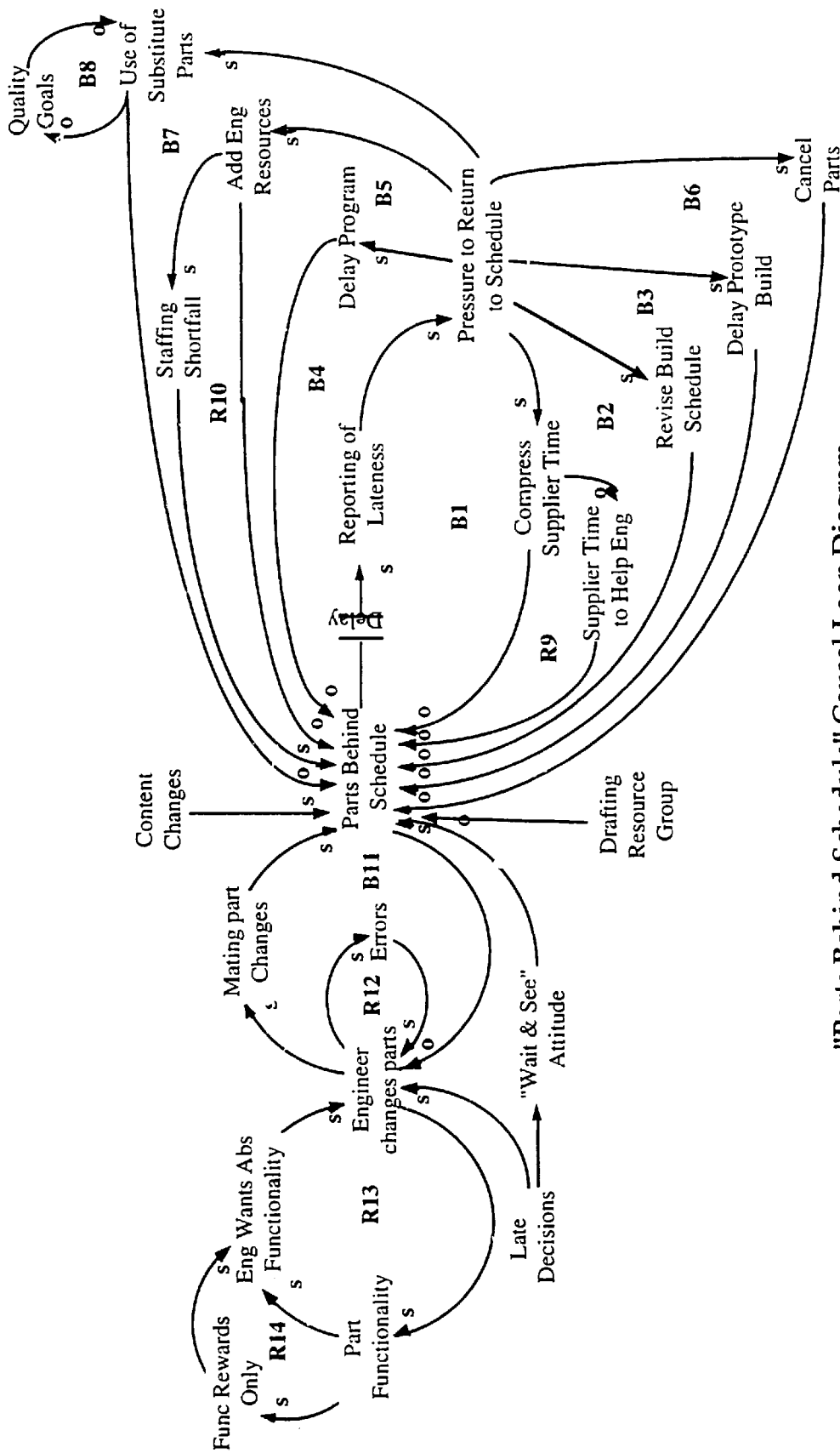
[Mgr. A]: That in addition to content changes is another one called late decision timing. We change the ground that it's on. There ought to be something...

•••

[Mgr. A]: Functional rewards, in other words, they don't care about timing, they're not going to worry about timing, they're aware of function. Timing is [a] program manager's problem. Shouldn't we add to the parts time schedule another thing is “Wait to start till we're sure that this is what they really want?”

•••

[Mgr. F]: Now this helps, when you stand back and learn, even what we just gone through...



"Parts Behind Schedule" Causal Loop Diagram
Figure 6.12

6.4.4 Leverage Points

Finding leverage points is not an obvious thing when dealing with complex social systems, such as a product development organization. The counter-intuitive nature of such systems has a way of directing one's attention to the place of least leverage, furthest removed from the fundamental source (Forrester, 1971). The causal loop diagram kept the types of interventions explicit and helped the team to picture the kinds of consequences they expected, as well as other unintended consequences.

[Mgr. A]: That's the leverage point even now, if we took action to reward or punish the engineer, not just for function, but for timing and for quality of EP parts, you'd get a different reaction, even now to know what's coming in the EP that we don't know about.

...

[Mgr. A]: It seems to me we want to work as much on the left side of [the diagram] as we can. Look for the leverage points and box them and either talk about an action plan or get together again and talk about an action plan. What actions it would take to have an impact on the leverage point?

...

[Mgr. B]: **This is really interesting because I'll tell you right now where we spend the bulk of our effort. To the right of parts [behind] schedule.** We do all of that stuff, some combination, every program does that, some worse than others, but every program is that, and we focus there, we throw so much energy into that....

Having all the actions that they typically take mapped out as responses to the delayed information about the lateness made it clear that those actions were all reactive responses to a deeper problem.

...

[Mgr. D]: There are reasons that some of these things happen, and one of the ones that's hurting us is that little circled delay there. And part of that is we don't walk our talk. We don't want to know the truth. We like people who come to us with good news.

[Mgr. B]: If we can project somehow, way in advance, that a part's going to be late because of information we have about it, then we can do something about it. For example, you know, if we could project this part's going to change, because we know another part's going to change, or because there's not resources working on this part, or a multitude of reasons, if we could project the lateness of the parts. Right now, we don't know the parts are late until they're late.

...

[Mgr. B]: ...I think we have a lot of leverage, potential leverage in that thing called delay. We're going to go back, and we're going to work on that. I want to book that for the moment.

It became apparent through the discussion that there was probably more than one leverage point. The diagram helped the managers to distinguish between leverage points that are currently available, given where they are in the program today, and ones whose relevant time was already past.

•••

[Mgr. A]: If we thought about the left hand side from the engineer that has the late part and we took his actions and split them into things outside of his control that affect him and therefore could make things late, and things within his control that affect whether things are late or not, think about it that way for a minute, I would conclude that late decisions is a high leverage point but it's too much to handle right now. It's another whole set of maps. But I'd say functional rewards only is a high leverage point that will affect how the engineer thinks. Timing could become very important to him, which would make him do different things even from the outside stuff that happens to him.

[Mgr. B]: In retrospect, there's a couple of really high leverage points there. But, we're beyond some of them, there is nothing we can do, that's basically done. But in the future, what are the high leverage points? And I agree with you, [Mgr. A]. That is a high leverage, because we still have to go through the verification program, we still have to go through the final design, the tweaking, and there's going to be incredible temptation to make a lot of change. So I would circle that "functionality," the whole issue of when do you stop engineering?

[Mgr. A]: If we were at a different point in a different program, I would box content changes as a high leverage point because it's discretionary and it really screws up the system, but we put in a change management control, we've said no more discretionary changes, so for our program, I don't think it's a high leverage point any more. I guess what you box depends on where you are.

•••

[Mgr. A]: I wouldn't box anything over on the revise build schedule and compress supplier timing, delay prototype build. Those are the last resorts that we have. We know what to do when we get to there, and they're not leverage points.

The completed map helped the managers see where they were in the context of the larger system which they were managing. As one manager observed:

[Mgr. B]: **...I'm now beginning to think how time-dependent we are in what we're doing, and as a result, this has been very beneficial to me. I'm really beginning to appreciate it, I'm beginning to understand what an incredibly complex thing we've got here...**

6.4.5 Outcomes

As a result of the mapping session described above and the identification of the delay in reporting lateness as a high leverage point, the Team Manager

and the Planning Manager focused their efforts on minimizing that delay. They began emphasizing the importance of informing as early as possible whenever a delay was known. They consciously worked on their behaviors to receive the “bad” news with encouragement rather than the more common and expected behavior of “shooting the messenger.” They began to hear about problems many weeks earlier than they would have typically, and thus, were able to minimize the use of the many “fixes” that were a normal part of the accepted way of managing a product program.

A critical test of the positive benefits of working on that leverage point (as well as other actions that were directly and indirectly related to the project) was the readiness of the product at the prototype phase. The APDP team achieved a part availability of 87%. As a basis of comparison, other programs usually come in at around 50% with no program ever exceeding 60%. The APDP team broke the company record by a large margin. Although, there is no way to directly attribute the result to the actions described above, the Team Manager and Planning Manager feel that they were an important factor.

Of course, the mapping session above was not the only thing that the core team worked on. By meeting together as a cross functional team, the project work helped improve the communication lines that Clark & Fujimoto (1991) identified as critical. The ability of the team to get information about the lateness of parts earlier is an important part of that communication. Another important piece of the communication channel is the frequency, bandwidth, and quality of the communication among the engineers in the various subsystem teams. As Figure 6.3 indicated, the ongoing intensive communication between upstream and downstream activities is an important factor in compressing the product development cycle time.

The systems mapping process that produced the insight and the chain of events described above was based on a Fixes-that-Fail archetype. In the next section, we illustrate how another systems archetypes translated into systemic insights that led to actions involving many players on the product development team.

6.5 FROM SHARED MENTAL MODELS TO ORGANIZATIONAL ACTION

The account described above illustrates how the process of making mental models explicit can help build shared understanding, identify high leverage actions, and catalyze individuals to act. Those actions, when taken by the top management of a product development program, can and did alter the behavior of many people which, in turn, reinforced the value of those actions in the minds of management. In addition, the changes in management action and policy about handling news about problems has had a cascade effect on others. There is a growing indication (based on field interviews and anecdotes) that the effects are diffusing throughout the larger APDP team.

Another contributor to the diffusion process is the use of learning laboratories which introduced a larger group (50 people to date) to the same process and tools that the core team had experienced. Through a two-day immersion, participants were introduced to new principles that helped change their individual *frameworks* and provided new tools that allowed them to produce new *routines* or responses to old problems.

In relation to our methodology of Chapter 4, the work described in this section is focused more on the use of archetypes described in the latter steps (8-10). The systems archetypes are presented as useful ways to view the organizational issues with examples from Cars developed by the core team members shared with other APDP members. The transferability of the

archetypal lessons appeared to be relatively high as the following account will illustrate.

The following story illustrates how systems archetypes can play an important role in building shared understanding among a group of people about an issue and lead to new actions. In effect, the principles underlying the archetypes can help to change the *Weltanschauung* of the organizational members and suggests new *operating procedures* that may become standard over time. To illustrate, we will take a closer look at the Tragedy-of-the-Commons archetype and how it was used in one of the subsystems teams of the APDP program at Cars, Inc.

6.5.1 Tragedy of the “Power Supply” Commons Story

As you recall from Chapter 3, in a Tragedy-of-the-Commons structure, each individual pursues actions which are individually beneficial, but which, over time, result in a worse situation for everyone involved because the collective action is depleting a “common” resource which is limited. At Cars, Inc., individual component teams in the APDP program were all drawing power from the limited power supply. It made sense for each component team to draw as much power as they required to maximize the functionality of their part. The collective result was an impasse in the design process, since no team could concede what they saw as in the interest of their own component. Individual attempts to resolve the issue were unproductive according to the power supply manager whose responsibility it was to work with the other teams to resolve the issue:⁸

⁸Again, to help preserve anonymity, the managers will be referred to as SSM A, SSM B, etc. SSM stands for Sub System Manager. Wherever there were specific references made that would reveal the nature of the company’s identity, I have substituted more general terms in [brackets].

[SMM A] ...There just wasn't enough power to [supply all the components]...So, we did all the normal Cars, Inc. things and we established a little [action team] and we brought all the members in and we sat around and we talked about problems and the weeks went by and we had these margins and we were, say, [x] amps from where we needed to be. And then after six or eight weeks we were [x-5%] amps from where we needed to be. And we were just sitting there floundering, looking at each other, saying, "Well, you know . . .", going through the same issues: "No, can't do anything with that, can't do anything with that."

Given the culture of Cars, the admission of their impasse was not an acceptable option. This relates to the discussion in the previous section about how bad news was usually received. When this manager attempted to go to management about this issue earlier, the response was a typical Cars response:

[SSM A] And I actually got some feedback from some management. I went to our manager and said, "We're struggling. I need some help."

"Well, are you telling me your team has failed?"

And I'm like, "well, no, our team hasn't failed. We just need more time."

"Well, take your time."

What typically happens in situations like this is that the teams will continue to struggle among themselves until at some point, the program timing is jeopardized and a decision has to be made. This usually means that the program manager has to step in and dictate what each team can have. This makes the teams unhappy because they were not able to make the decision themselves and because some of them did not get what they wanted. The manager is not happy because he had to intervene when he had expected the team to make the decision. The conclusion is that this was just an instance of a poorly-aligned team or a heavy-handed manager. There is no general framework for them to learn from, so it is seen as situation-specific.

Given the mindset that either you solve it yourself or you are a failure, it is not surprising that the team is reluctant to go to the Team Manager (TM) without having the issue resolved:

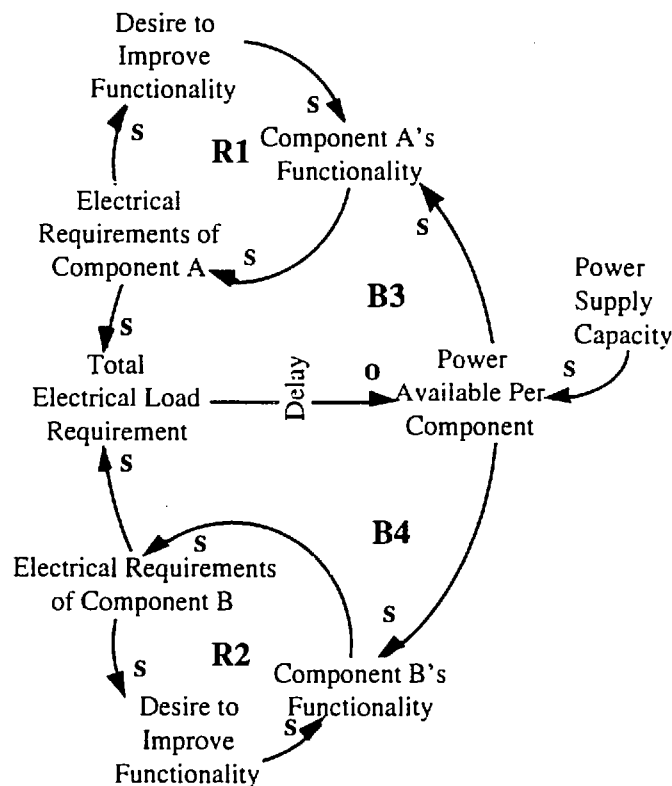
[SSM A] There just wasn't enough power to go around. And the solution to the Tragedy-of-the-Commons [as I see it now], if you can't resolve it within your peer level—which

we couldn't—you've got to go to the king and say, "King, they're misusing the resource. You've gotta implement some policy."

[But] we tried doing that before [without the Tragedy-of-the-Commons archetype], in the team saying, "Hey man, we've gotta go up to [the Team Manager] with this. We're just not getting anywhere."

"Oh, well, we can't—let's keep working at it."

Cause everybody felt on the team that if we do that, this team failed. Nobody wants to fail. Everybody wants to succeed.



Tragedy of the Power Supply

Figure 6.13

From a systems perspective, the team was caught in a Tragedy-of-the-Commons structure (see Figure 6.13) which meant that the problem cannot be solved at the individual level. Only a collective governing body or an individual with the authority to impose constraints on all the teams could resolve the situation. Once the team mapped out their situation and saw it as

a Tragedy-of-the-Commons archetype, the course of action and the *reasons for why the action was necessary* was clear..

[SSM A] When...we came back from [the learning lab] and there were a couple of us...who had been through the training...we went to the [power supply] team and said, "Hey, we've been eight weeks on this and we're not getting anywhere. We're convinced we're not going to get anywhere at our level because we're in a Tragedy-of-the-Commons. And that's where we got the funny looks...And the only solution to the Tragedy-of-the-Commons is to go to the King, so we gotta go to [the Team Manager].

Their first inclination was that [the Team Manager] would, you know, rip our head off. And I said, "You know, I'm pretty confident we'll go into [the Team Manager's office] and we're gonna say,

"[Team Manager], we're caught in a Tragedy-of-the-Commons and we need your help. And here's our solution. Here's our shopping list. You pick the ones you want. You know, you're the [APDP] manager."

And everybody on the team thought, "Yea, right."...But we did. We went in saying "we have a Tragedy-of-the-Commons. Can you help us out?"

The program manager was given all the component designs and was asked to make the decision on how much power to allocate to each of the components. Those who had to give up some functionality did not like it, but they understood *why*, given the systemic structure. A critical feature of this story is that the Team Manager also understood the principle of the Tragedy-of-the-Commons archetype. It provided a common language and framework that helped both parties rise above the common view that individuals (or teams) are responsible for either succeeding or failing and see that there are larger systemic structural forces that govern certain situations. The message of the Tragedy-of-the-Commons is not that individuals should not focus solely on their own actions, but rather to view those actions in a systemic context that reveals how the rewards and incentives conspire to make solutions at the individual level ineffective.

The following account from the Team Manager shows his understanding of this Tragedy-of-the-Commons case from his perspective:

[TM] [The power supply team was] fooling around and fooling around, trying to solve this problem. They didn't want to bring it to me. Months were going by. We were getting to real cut-off points where it was going to be too late to influence design. They couldn't solve the problem and they regarded that as a failure and so they didn't want to tell me they couldn't solve the problem. Okay, I didn't know there was a problem to solve because they wouldn't tell me—which is the normal Cars, Inc. environment.

And, finally they mapped it. They used the [systems mapping] technique. They mapped it. They said, "This is a case of 'Tragedy-of-the-Commons. There is absolutely nothing we can do that's gonna get us out of this dilemma. And they came to me and I made the decisions and I said, "Okay, here's what we're gonna do. You're gonna do this. You're gonna do that. And you're gonna do this.

And they took those as orders, which they accepted because while they knew, while it was against their chimney's objective, it was what it took to solve the Tragedy-of-the-Commons. And they viewed it as an insoluble problem within their constraints. So, they were happy to accept somebody saying, "Here's what you're going to do.

Whereas, if they didn't map it, they couldn't reach that conclusion. And they would have fooled around for another couple of months and it would have been too late. We would [then] have had to do something really drastic at the time of the prototype . . . because the prototypes wouldn't run.

The Team Manager went on to explain how the company's system was set up to make this a case of Tragedy-of-the-Commons:

[TM] Well, they laid out all their individual problems and their incentives: What their organization wanted, what its goals were and how it was going to be measured, and what the effect of what it would have to do to solve this problem would be on that chimney and its goals and how it measured the performance of its people on the team. And what they were doing was absolutely consistent and right with how they were going to be judged in their chimney.

And, when they laid out the whole map, it was clear that they were all going to be right and the end result was going to be the [APDP] would fail. So, it was absolutely clear that we were trying to get out of the system more than existed and, as long as we all kept doing what was right for how we were going to be measured, we were going to fail.

And yet, no one wanted to be the one that would be criticized by his management for doing something foolish with his component in order to make somebody else successful. And once they laid it out and it was clear to all of them that this was happening, they said, "Okay, we're not going to resolve this but in the end, the [team] manager could just order it on the basis of the whole. I, as engine engineer can never agree that's what I want to do. But, I can accept that's what I have to do to make the APDP work if the [team] manager tells me that's what I have to do.

Otherwise, I going to say, "Why don't you do it" to somebody else. "Why don't you fail. I don't want to fail." And, so you keep going around in circles, saying, "You do it, I'm not going to do it." And, nobody agrees. And so, they took it to me. We made the decisions and it works.

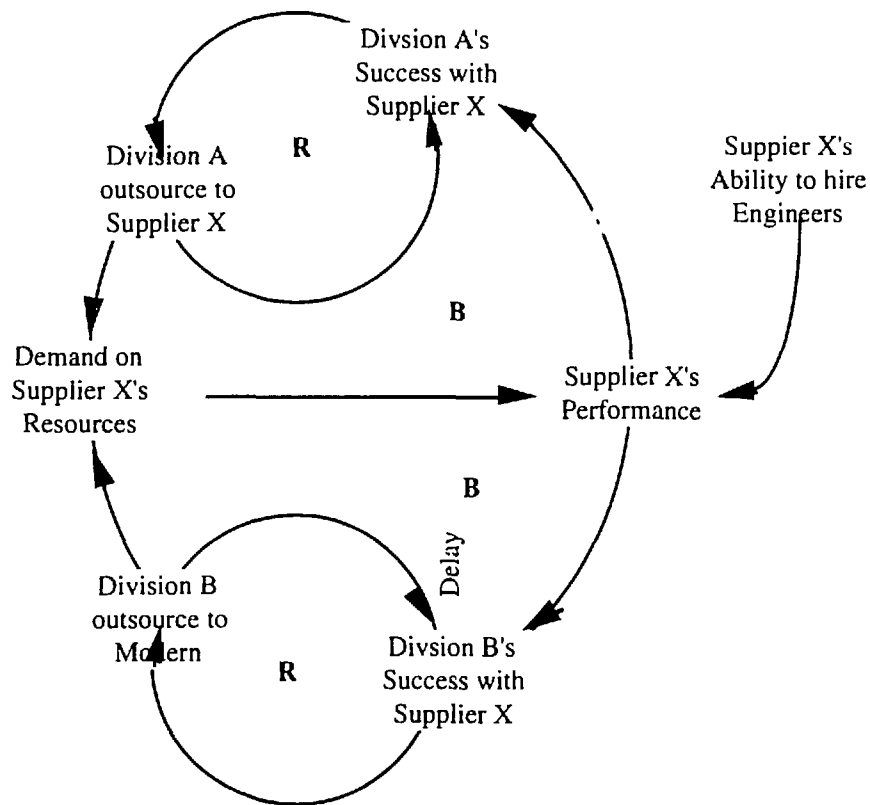
The decision was seen to work from the power supply team's perspective as well:

[SSM A] The team left, not feeling [like] they had failed. They just felt they had done as much as they could within their role, and they needed more leverage. The leverage came from management.

Subsequent teams looked at other common resource issues and realized that they, too, fell into the Tragedy-of-the-Commons structure. For example, mapping their relationships with component suppliers also revealed a Tragedy-of-the-Commons situation (see Figure 6.14). The "commons" in this case was the performance of a particular supplier. As a division builds up a history of a good working relationship with Supplier X, that division is more likely to contract with Supplier X in the future. If another division decides on the same supplier, Supplier X's ability to satisfy both divisions' requirements can erode. Successive demands by more and more divisions can over-burden the supplier until no division is being served well.

Supplier X, given its dependence on and historical relationship with Cars, feels that they are not in a position to say "no" to any of the requests in fear of losing future work. The unfortunate outcome is that Supplier X becomes over-burdened and fails to deliver on some of its commitments. Cars' purchasing department concludes that Supplier X is not reliable and drops them from their list of suppliers.

What is emerging from the realization of the ubiquitousness of the Tragedy-of-the-Commons archetype in the product development setting is a systemic understanding of why a heavyweight program manager with wide authority over the entire product program makes sense. The ToC archetype provides a theoretical explanation for the successes of companies using heavyweight project managers (Clark & Fujimoto, 1991) the implications of which we will explore further in section 6.6.1.



Tragedy of the Suppliers

Figure 6.14

6.6 RESEARCH FINDINGS

In the preceding sections, we presented the process of applying the methodology developed in Chapter 4 to the product development setting. Two insights were gained from the work at Cars, Inc., which contain broad implications for managing the product development process: (1) Tragedy-of-the-Commons as a significant dynamic that must be explicitly managed in the product development setting; and the communication bottlenecks which have a compounding effect on delays and future levels of communication. In the following sections, we will explore these research findings and their implications in more detail.

6.6.1 A "Commons Management" View of Product Development

Clark and Fujimoto (1991) list several characteristics that a heavyweight product manager must have to be effective. Among them include the following:

- Coordination responsibility in wide areas, including production and sales as well as engineering,
- Coordination responsibility for the entire project period from concept to market,
- Responsibility for concept creation and championing as well as cross-functional coordination,
- Frequent and direct communication with designers and engineers at the working level as well as through liaisons,
- Possess multi-lingual and multi-disciplined abilities in order to communicate effectively with marketers, designers, engineers, testers, plant managers, controllers, and so forth,
- Role and talents in managing conflict surpass those of neutral referees or passive conflict managers; they may initiate conflicts to prevent product designs or plans from deviating from the original product concept,
- Circulate among project people and strongly advocate the product concept rather than do paperwork and conduct formal meetings. (Clark & Fujimoto, pp. 256-257)

The above list identifies *what* characteristics are required to be a successful heavyweight product manager; the experience with the APDP program is beginning to show the *why* and the *how* of being successful.

Each of the characteristics mentioned above sounds attractive. Who wouldn't argue for having a product manager possess multi-lingual and multi-disciplined abilities or conflict management skills? Recognizing a desirable set of characteristics is relatively easy; obtaining the skills and managing by them is much more difficult. Part of the difficulty lies in the fact that learning new tools and skills alone is inadequate by themselves.

As the Cars case showed, there are organizational norms and beliefs that influence the way people behave. When you put the organization under stress, either by trying to change it or when it encounters a crisis, you are likely to run into the organization's culture (Schein, 1987). Such occasions

can also be opportunities for the organization to learn and affect its culture, as Schein (1987) observed:

When an organization faces a crisis, the manner in which the leaders and others deal with it creates new norms, values, and working procedures and reveals important underlying assumptions. Crises also are significant in culture creation and transmission partly because the heightened emotional involvement during such periods increases the intensity of learning. If people share intense emotional experiences and collectively learn how to deal with very emotionally involving situations, they are more likely to remember what they have learned. (Schein, p. 230)

The APDP core team, through the changed behaviors of that program's leaders and the changes in others who have been involved in the pilot project, have begun to challenge and change some basic operating norms. As the team manager of APDP reflected:

...[E]very time you reach a stalemate in your working level group and you have to elevate to the next level of management to get a resolution, that stuff takes time, wastes a lot of people's energy. You have to prepare papers for it, and everyone has to come in and posture, and you bring all your management together. Every time you have to do that, you waste time. Sometimes it takes weeks to get resolutions.

It also causes hard feelings. Somebody wins, and somebody loses in those battles.

So that's what you don't want to do. You want to get the team to consense. And for me, one of the **paradigm shifts I had to make** [emphasis added] was for a long time, when you look at a situation, you see things going down the downhill side of the problem resolution, you say,

'Somebody's screwing up here. There's got to be somebody I can go out and find and crucify for this.'

That's just typical of how we think about situations, and when you take a step back and say, 'If I change, if I start to think, I've learned from experience, when you really get into these issues, you'll find that it's not any one individual that isn't trying as hard as he absolutely could, and didn't think he wasn't doing the right thing.' I haven't met one person at [Cars] who isn't trying to do a good job, isn't trying to do the best he can from his perspective.

The secret is, when you go in with that mindset, instead of looking for someone to hang, you might learn something [emphasis added]. You might find out that this person has another issue that he's been struggling with, and maybe can't talk about because of a management issue, or in the case of a supplier who's terrified about saying anything negative about the company for fear that maybe he won't get the next program. You got to get past that stuff or you'll never learn anything from each other. You still walk around trying to find somebody to point the finger at. 'Oh, your part's late, so it must be your fault.' It just isn't productive. And it doesn't help.

Accomplishing such a shift in paradigm requires learning at the operational and conceptual levels. In terms of our framework for organizational learning developed in Chapter 2, we need to provide new frameworks and develop a new organizational *Weltanschauung* to complement the tools in order to produce new routines and procedures to advance organizational learning.

6.6.1.1 *Tragedy-of-the-Commons Revisited*

A shift in thinking that may help product development organizations in general is to view their whole process as a Tragedy-of-the-Commons issue. This is an alternative to the “zero-sum” game, “survival of the fittest,” and the “us vs. them” assumptions on which most are based.

The dynamic script for the Tragedy-of-the-Commons archetype was laid out by Garrett Hardin (1977). Hardin’s work was a reaction to Adam Smith’s popular idea of the “invisible hand,” which states that the individual who intends only his own gain is believed to be led by an invisible hand to promote the public interest. Although Smith never asserted this to be true, enough of his ideas support this thinking that many still assume that decisions based on individual self-interest will be in the best interest of society as a whole. An implicit assumption that is rarely made explicit, however, is that the invisible theory is contingent upon the existence of unlimited resources.

Hardin’s description of managing the commons is based on a little-known pamphlet written by mathematical amateur William Forster Lloyd in 1883.⁹ Hardin describes the Tragedy-of-the-Commons using the analogy of a

⁹Cited in Hardin (1977) as W.F. Lloyd. *Two Lectures on the Checks to Population* (Oxford University Press, Oxford, England, 1833).

common pasture that is used by herdsman to graze their cattle. In order to maximize the growth of their herd, it is expected that each herdsman will try to graze as many cattle as possible. As long as nature, wars and disease keep numbers of humans and their animals below the carrying capacity of the land, this system works well. However, each additional animal increases the load upon the land, until each added cattle will decrease the benefit to all herdsman. It is still in the best interest of the individual herdsman to keep adding animals to his herd, but each time he does, the commons suffers, the effects of which are shared by all herdsman:

Each is locked into a system that compels him to increase his herd without limit in a world that is limited. Ruin is the destination toward which all men rush, each pursuing his own best interest in a society that believes in the freedom of the commons. Freedom in a commons brings ruin to all...(Hardin, 1977, p.20.)

Where does the leverage lie in this situation? Although we can legislate prohibition, as Hardin says, we cannot legislate temperance. It is difficult to persuade the individual herdsman to forego the incremental gain each additional cattle adds to his herd, especially when the collective negative result of that action appears slowly over time.

6.6.1.2 Economic Externalities

The situation is the same when a firm has limited resources and each department vies for more use of a common resource in order to produce the best product. As the product development team members acknowledged, appealing to the “good of the whole car program” backfires, because no component group leader is willing to sub-optimize his component (and face the backlash from his functional area or team) in order to optimize the whole. One possible solution is what Hardin calls “mutually agreed upon

coercion," that is, mutual coercion, mutually agreed upon by the majority of people affected.

Economists refer to this whole dynamic as the theory of externalities. Meade (1973, p. 15) defines externalities in the following way:

An external economy (diseconomy) is an event which confers an appreciable benefit (inflicts an appreciable damage) on some person or persons who were not fully consenting parties in reaching the decision or decisions which led directly or indirectly to the event in question.

Meade refers to externality in those matters involving "ill-defined ownership," where there is some scarce resource such that the more 'you' use it, the less there is for 'me,' a clear linkage to the notion of the commons. In these instances, Meade asserts, we cannot rely upon voluntary groups to internalize all important externalities, but governmental compulsory intervention will often be necessary (Meade, 1973, p. 57). This is precisely the role of the heavyweight product manager.

Similarly, Baden (1977) asserts the need for governmental intervention in cases of common pool resources, where one or a set of users can have adverse effects upon the interests of other users. In the situation where there is no agency with the power to coordinate or to ration use, *action which is individually rational can be disastrous*. Baden also mentions the "free-rider problem," which states that in cases where investment by each member of a program would benefit all, the incentive for an individual, is not to contribute and still reap the benefits of his fellows who seemingly do invest. If every individual thought this way, however, there would be no contribution by any member.¹⁰

¹⁰Baden suggests that the world, as a whole, is taken on "an ever-greater resemblance to a common pool" (Baden, 1977, p. 141).

6.6.1.3 *Strategic vs. Operational Autonomy*

In organizational settings, managing the commons requires suspending willing consent, relinquishing autonomy in certain areas in order that the collective good is served. Bailyn (1984) explores the meaning of autonomy among workers in an R&D lab, making an important distinction between strategic and operational autonomy. She defines the former as “the freedom to set one’s own agenda,” vs. the latter, “the freedom, once a problem has been set, to attack it by means determined by oneself, within given organizational resource constraints” (Bailyn, p. 7). In her research she found that most scientists valued operational over strategic autonomy; unfortunately, so did their managers.

In matters affecting the commons, it may be more effective for managers to take charge of strategic decisions regarding scarce resources. Engineers, once they become aware that the Tragedy-of-the-Commons structure is operating may recognize the necessity for management to take charge of the strategic decisions, such as allocating power requirements or weight limits. At Cars, Inc., the team members voluntarily relinquished strategic autonomy, while maintaining operationally autonomy to engineer the best parts and use-systems possible within the pre-set constraints and objectives.

6.6.1.4 *Managing the Product Development “Commons”*

The Tragedy-of-the-Commons findings of this study and the above discussion provide theoretical support for Clark & Fujimoto’s (1991) findings on the importance of having a heavyweight project manager. Their studies showed that a product managers who have broad authority in setting agenda, and who clearly convey that information so that the engineers know what is expected of them have significantly greater success than those who have less

authority and influence. How does viewing the product development process as a commons management issue change the way it is managed? The following is a tentative list of implications:

At the sub-system or component team level

- At the start of each product program, all the potential Tragedy-of-the-Commons (ToC) issues should be identified,
- All the individual players who are part of a ToC should know what commons they are a part of and with whom they are sharing it with,
- Each commons member should know who the governing authority is for his or her commons,
- Those who share a commons should have systems in place to maximize coordination and sharing of information,

At a program manager level

- Identify all the common “pools” of company resources that the program depends on for success,
- Identify all the critical suppliers who are susceptible to being overburdened by demands from all programs,
- Meet regularly with other program managers to update and communicate for the purposes of managing the commons,

At the product development organization level

- Design an information system that provides real-time feedback to all sub-systems and component teams how their commons is being affected by everyone’s collective actions,
- Design a dual recognition system that gives teams credit for producing the best individual part and also credits them for “give-ups” which sacrifice part functionality for product integrity.

A lot of items listed above may sound like basic resource allocation issues, and at a basic level, they are. The difference lies in the underlying assumptions which governs all the prescribed actions. As noted above, a “zero-sum game” mentality or “individuals are at fault for anything that goes wrong” belief will produce a very different system and outcome when operationalizing the above points. Viewing it as “managing the Tragedy-of-the-Commons,” leads one to focus on how the system as a whole is interacting to produce undesirable results and direct efforts to work on the *system* rather than on the individuals.

6.6.2 Structural Impediments to Intensive Communication

Another major finding in the research involved the leverage point discovered from examining the mapping of parts behind schedule. Team members on the ADPD program knowingly delayed providing information about late parts to management because of the measurement system which punished them for being the bearers of bad news. The system encouraged each team to delay reporting its own problems in the hopes that some other team’s problem would be announced first. Once somebody’s part is late, it often meant that everyone else’s schedule gets revised because of the impact of one part is usually felt by multiple parts.

The case of the reporting of late parts (described in section 6.4.1) illustrates the importance of *timely* communication in the product development process. As Clark and Fujimoto’s diagram showed (Figure 6.3), lack of early communication in a product development process creates confusion and delays in the downstream process. Two reasons for lack of communication that were uncovered during the work with Cars, Inc. were the incentive system and the functional gridlock that can occur when individual team

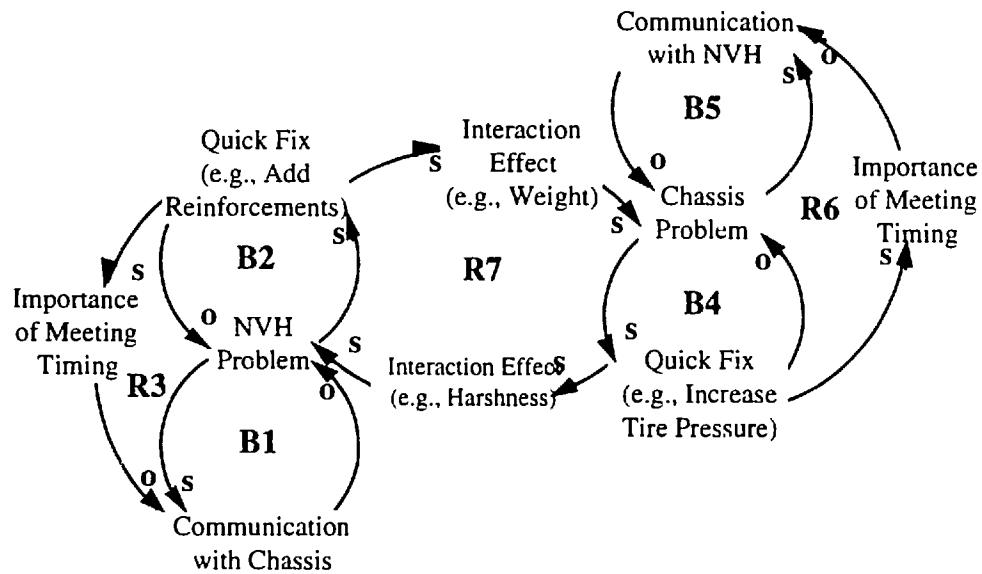
actions inadvertently create problems for other teams. This phenomenon was mapped into a Shifting-the-Burden archetype by the core team (see Figure 6.16).

6.6.2.1 *Shifting-the-Burden and Functional Gridlock*

As explained in Chapter 3, in a Shifting-the-Burden structure, a problem is “solved” by applying a symptomatic solution that diverts attention away from more fundamental solutions. When the symptomatic solution creates another problem, prompting *further* symptomatic solutions, the double Shifting-the-Burden pattern that results can spawn a whole maze of interlocking problems. In the process, the organization’s ability to fundamentally resolve the problem atrophies.

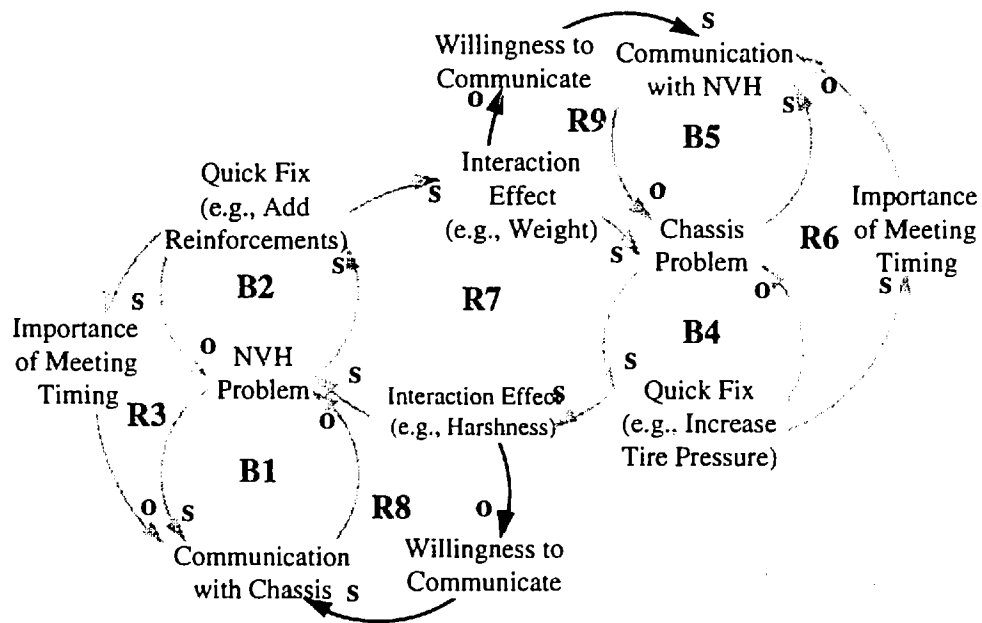
In the APDP program, a Noise, Vibration, and Harshness (NVH) team encounters a noise problem (see Figure 6.15). Ideally, they should coordinate with all potentially affected parties (B1) but, given the time pressures they are under, they devise a fix independently by adding reinforcements to the car (B2). This solves their problem which improves their ability to meet timing (R3), and they move on to their next problem.

NVH’s fix for the noise problem, however, increases the car’s weight and presents a problem for the chassis team. Chassis, facing the same timing pressures in turn, “fix” their problem by increasing the tire pressure (B4), instead of first communicating with the affected parties (B5). This action helps the chassis team meet their timing objectives (R6), but now has created a harshness problem that NVH will later discover and have to fix. Another round of NVH quick fixes lead to another round of chassis quick fixes in a vicious reinforcing spiral (R7).



NVH-Chassis Reinforcing Cycle of Interlocking Problems

Figure 6.15



NVH-Chassis Interlocked in an "Us vs. Them" Structure

Figure 6.16

As the fixes may lead each team to focus more and more on meeting their own timing targets, they invest even less in cross-team communication (R3

and R6). The interaction effects (e.g., reinforcements leading to an added weight problem for chassis) lead to an increasing unwillingness to communicate with the other team, and further reinforces an "us versus them" mentality (see Figure 6.16). Over time, that mentality becomes more entrenched through more cycles of the reinforcing loops R8 and R9.

6.6.2.2 *Teams: Definition vs. incentive structure.*

Given the impediments to communication described above and in section 6.4.1, what are the leverage points for fostering improved team communication? Senge (1990b) asserts that an essential element of learning within teams is alignment around a common goal. Yet the recent experience with team management programs has illustrated that grouping people in a company together to produce a certain product or goal is not enough. Many traditional organizational norms and routines, such as incentive systems, run counter to the creation of an optimal team.

In many companies, performance reviews are conducted on an individual basis, as are the rewards and punishments of individual players. In many product development organizations, like the one at Cars, Inc., team members are evaluated by the functional "home" organization who have little stake in the specific car program. In addition, Ostrom (1977) states, "Consequences which cannot be measured directly in dollar terms have often been ignored. While we may not be able to attach money value directly to some consequences, all events can be evaluated in terms of the alternatives foregone."

The Tragedy-of-the-Commons and the Shifting-the-Burden archetype offers some general insights into managing teams for optimal performance. The basic insight is that as long as there is a difference between what is good

for the team and what is good for individual team members, attempts to create team unity are at risk of being undermined by the fundamental dynamics at work.

6.6.2.3. *Implications*

A system that discourages timely reporting of problems not only hampers an organization's ability to respond effectively to the problem by ruling out certain options, it also shortens the window of opportunity for responding in a timely and effective manner. The leverage in breaking through the communication gridlock is to change the system from one that tracks problems for "controlling" (which in reality gets translated into "punishing") purposes to one that is used for planning and coordinating. As the APDP program management discovered, encouraging the early reporting of problems and working extra hard not to be punitive (both intentionally and unintentionally) can produce dramatic results (recall that the APDP team achieved a part availability of 87% vs, a company average of 50% for their EP prototype build). This requires a shift in program management's thinking to become less fixated in today's problems and to focus on the long-term systemic result that is desired.

Making such a shift requires a relationship of trust between program management and the teams, as well as between each of the teams. Building that trust is not easy, especially when different people from different parts of the organizations have very different mental models about the "right way" of doing things or "what's good for the program." The framework and methodology for building shared mental models presented in this dissertation is proposed as an important first step that a product development team can take to build the common ground from which everyone involved

in a program can work together towards achieving their shared vision of the final product.

6.7 BUILDING SHARED MEANING

Archetypes such as Shifting-the-Burden and Tragedy-of-the-Commons can help to elicit and capture the intuitive understanding of experienced managers about complex dynamic issues. They are particularly powerful for advancing conceptual learning because they help explicate a better understanding of know-why and offer guidelines for operationalizing those insights into know-how.

The above Tragedy-of-the-Commons example shows how mapping the dynamic structure of the teams' inter-relationships allowed them to make explicit a common view of their issue in a framework that improved their conceptual understanding of the situation. The systems archetype also helped in prescribing a course of action to resolve the issue once there was a *shared* understanding of that structure. The archetype provides a way of mapping the individual instance to a general framework to guide future learning.

6.7.1 The Role of Computer Simulators and Learning Labs

As pointed out in Chapter 4, however, there are limitations to what can be accomplished with archetypes and causal loop diagrams alone. A computer model can capture all the causal relationships from the pen-and-paper diagrams and simulate the consequences of intended actions over a long period of time. A person does not give up mental models easily unless they are proven to be no longer useful. Even then, it is difficult to let go of what is familiar. One of the benefits of a simulation model is in helping an individual become more open to other possibilities and to be exposed to others' assumptions and mental models by 'running' multiple scenarios and

testing one's assumptions repeatedly. But, translating an individual's (or a small group's) mental model representations into a good computer model can take an enormous amount of time and effort. It is not a process that is economically feasible to replicate with everyone in the organization, and yet having the insights generated by such a process may be very important to have widely shared. How can one address this dilemma?

One way is through the design and implementation of *microworlds* or *learning laboratories* (Kim, 1989; 1990; Senge, 1990b). Senge & Lannon (1990) describe them as the equivalent of managerial practice fields where teams of managers can practice and learn together. Learning laboratories are designed, in part, around the learnings captured through the systems archetypes. The spirit of the learning lab is one of active experimentation and inquiry where everyone participates in surfacing and testing each other's mental model. Through this process, a shared understanding of the key assumptions and inter-relationships of the organization emerges.

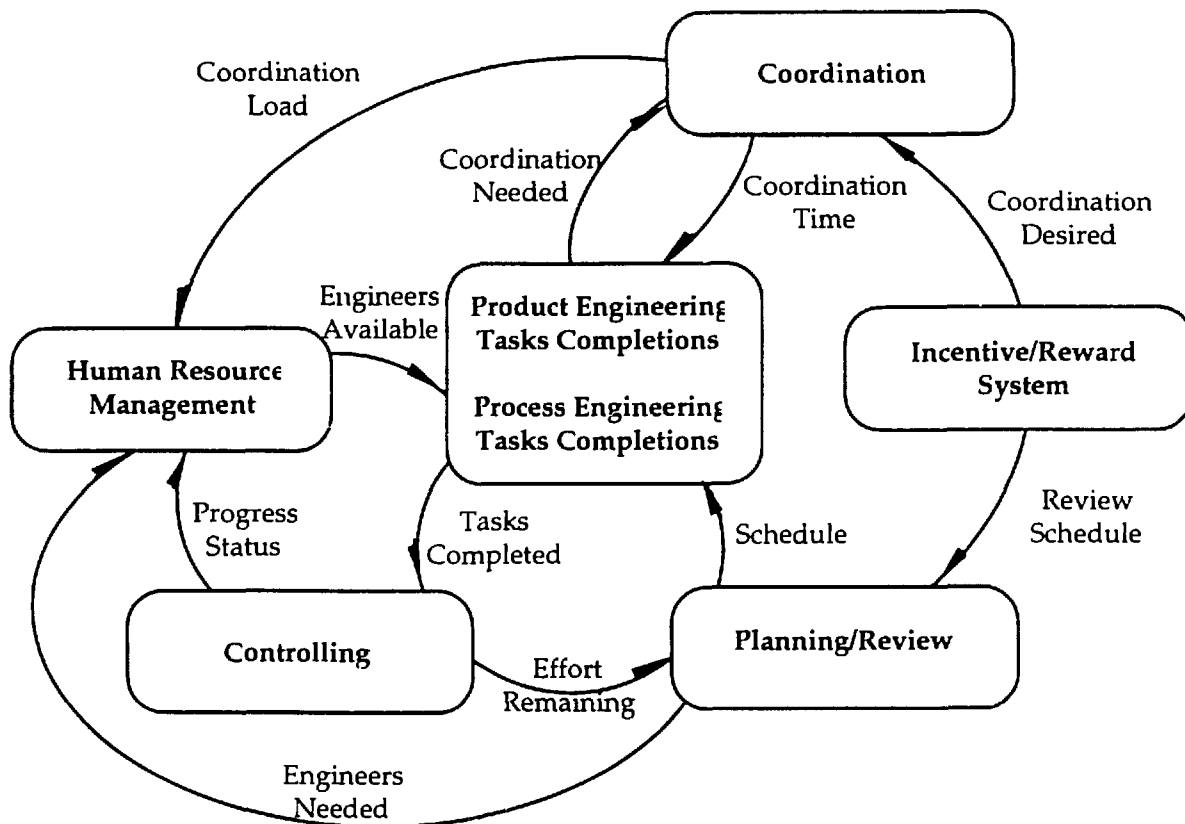
6.7.1.1 *Overview of Computer Model*

The use of an interactive computer management "flight" simulator (based on a system dynamics computer model) offers participants an opportunity to test their assumptions and to viscerally experience the consequences of their actions. Management flight simulators represent mental models that have been translated into a more formalized and explicit computer model.

The product development management flight simulator (PD-MFS) focuses on managing the up-stream (product engineering) and down-stream (process engineering) activities in a typical product development program. The computer model and simulator that has been developed thus far addresses the following issues:

- parallel/serial development,
- degree of coordination between product engineering and process engineering,
- the manner in which one staffs up the effort from beginning to end,
- the influence of budget pressure, schedule pressure, or both on decisionmaking,
- the extent to which one allows the target scheduled date to slip,
- ensuring quality throughout the process

The major issues addressed by the PD-MFS used in the learning lab are shown in Figure 6.18 (for a more complete description of the PD-MFS computer model, flight simulator, and user's guide, see Appendix D and E)



New Product Development Model Overview

Figure 6.17

6.7.1.2 *Learning Labs*

At Cars, Inc., we conducted several two-day learning labs which provided an opportunity for APDP team members to learn new conceptual tools and apply them to their work situation. The learning lab design combined the research process and mapping methodology described in Chapter 4 with a computer simulator on product development management. The learning lab consisted of an introduction that discussed the relevance of Senge's (1990b) five disciplines of the learning organization, moving from problem-solving paradigm to problem articulation, and the opportunity inherent in a learning lab to practice learning together.

The participants worked in pairs on the computer simulator and tried to manage a product development project to meet cost, timing, and quality objectives. By working in pairs and encouraging them to make explicit the reasoning behind their decisions, the mental models that drove their decision-making were surfaced. They discovered, for example, how their assumptions about the right pace of staffing and coordination between product and process engineering led to missing all three targets. The learning lab is currently being used as a mechanism for sharing the Shifting-the-Burden and Tragedy-of-the-Commons stories by allowing others in the car program to work through similar issues and discover their own insights.

Through wide use and successive iterations of the learning lab, the practice sessions are expected to impact the organization's shared mental models via its *Weltanschauung* and organizational routines. People leave the learning lab with tools that they can use in their work settings which advances operational learning. Tools like systems archetypes also embody a set of principles which helps advance conceptual learning as individuals use them.

6.7.2 My *Weltanschauung*

I believe that the process of surfacing individual mental models and making them explicit can help accelerate *individual learning*. As mental models are made explicit and actively shared, the base of shared meaning in an organization expands as does its capacity for effective coordinated action which advances *organizational learning*. The way in which Johnson & Johnson handled the Tylenol poisoning incident is a vivid example of what a powerful force a deeply shared value (or *Weltanschauung*) on the value of a human life can have on an organization's ability to mobilize a coordinated response in a short space of time.

On a less grand scale, systems archetypes provide a glimpse of the possibility of developing a similar capability for enhancing coordinated action. As more people understand the meaning behind the Tragedy-of-the-Commons archetype, the use of the term itself will conjure up the whole storyline behind it as well as its implications for action. Vygotsky (1989), offers an analogy that highlights the potential.

Very good examples of the condensation of external speech and its reduction to predicates are found in the novels of Tolstoy, who quite often dealt with the psychology of understanding: "No one heard clearly what he said, but Kitty understood him. She understood because her mind incessantly watched for his needs" (*Anna Karenina*, part V, Chapter 18). We might say that her thoughts, following the thoughts of the dying man, contained the subject to which his word, understood by no one else, referred. But perhaps the most striking example is the declaration of love between Kitty and Levin by means of initial letters (*AK*, part IV, chapter 13):

"I have long wished to ask you something."

"Please do."

"This," he said, and wrote the initial letters: *W y a: i c n b, d y m t o n*.

These letters meant: "When you answered: it can not be, did you mean then or never?" It seemed impossible that she would be able to understand the complicated sentence.

"I understand," she said, blushing.

"What word is that?" he asked, pointing to the *n* which stood for "never."

"That word is 'never,'" she said, "but that is not true." He quickly erased what he had written, handed her the chalk, and rose. She wrote: *I c n a o t*.

His face brightened suddenly: he had understood. It meant: "I could not answer otherwise then."

She wrote the initial letters: *s t y m f a f w h*. This meant: "So that you might forget and forgive what happened."

He seized the chalk with tense, trembling fingers, broke it, and wrote the initial letters of the following: "I have nothing to forget and forgive. I never ceased loving you."

"I understand," she whispered. He sat down and wrote a long sentence. She understood it all, and without asking him whether she was right, took the chalk and answered at once. For a long time he could not make out what she had written, and he kept looking up into her eyes. His mind was dazed with happiness. He was quite unable to fill in the words she had meant; but in her lovely, radiantly happy eyes he read all that he needed to know. And he wrote down three letters. Before he had finished writing, she was already reading under his hand, and she finished the sentence herself and wrote the answer, "Yes." Everything had been said in their conversation: that she loved him, and would tell her father and mother that he would call in the morning.

These examples show clearly that when the thoughts of the speakers are the same, the role of speech is reduced to a minimum.

I believe that group efforts to clarify and share our individual mental models can produce positive results in terms of effecting coordinated responses in less time than it currently takes organizations to mobilize in the face of today's challenges.

6.7.3 A Final Note

In this Chapter, we loosely followed the 10-step process in an interactive, real-time working sessions with line managers. In the early stages of the project, more emphasis was placed on the first six steps as the team focused on getting themselves grounded in their own data as well as those whom they interviewed (6.4). The team looked at the data through multiple lenses, including action maps and KJ diagrams (described briefly in Chapter 3) as well as causal loop diagrams and systems archetypes.

As the team became more certain of the data and their understanding of them, we shifted our emphasis to the latter step where the focus is on making the explicit mental model representation shared (section 6.5). At this stage, the use of the systems archetypes and a learning lab became more significant for sharing purposes. In terms of the OADI-SMM model, managing the transfer mechanism of mental models made explicit through systems archetypes and computer simulators can reduce situational and fragmented

learning and enhance organizational learning. Preliminary assessments indicate that the learning labs are succeeding to a degree in transferring and spreading the knowledge and skills to a broader group. Specific new actions and benefits gained have been recorded.¹¹

The research process and mapping methodology described in this dissertation shows a lot of promise in helping product development teams to speak a common language—the language of “systems thinking”—to articulate and work through their issues in a productive and effective manner. Further field work with additional teams can provide a richer, more diverse experience base with which to refine the methodology and research approach.

Chapter 6 References

- Adler, Paul S., Riggs, Henry E., & Wheelwright, Steven C. (1989). Product Development Know-How; Trading Tactics for Strategy. *Sloan Management Review*(Fall), 7-17.
- Allen, Thomas J. (1977). *Managing the Flow of Technology*. Cambridge, MA: MIT Press.
- Allen, Thomas J. (1980). R&D Performance as a Function of Internal Communication, Project Management, and the Nature of Work. *IEEE Transactions on Engineering Management*(Feb), 2-12.
- Ancona, Deborah G., & Caldwell, David E. (1992). Cross-Functional Teams: Blessing or Curse for New Product Development. In T. Kochan & M. Useen (Eds.), *Transforming Organizations* Oxford: Oxford University Press.
- Argyris, Chris, & Schön, Donald (1978). *Organizational Learning: A Theory of Action Perspective*. Reading, MA: Addison-Wesley.
- Baden, John (1977). A Primer for the Management of Common Pool Resources. In G. Hardin & J. Baden (Eds.), *Managing the Commons* (pp. 137-146). New York: W. H. Freeman & Company.
- Bailyn, Lotte (1984). Autonomy in the Industrial R&D Lab. *MIT Sloan School of Management, Working Paper 1592-84*.
- Beer, Michael (1980). *Organization Change and Development: A Systems View*. Santa Monica, CA: Goodyear Publishing.
- Clark, Kim B., & Fujimoto, Takahiro (1990). The Power of Product Integrity. *Harvard Business Review*(Nov.-Dec.), 107-118.

¹¹OLC follow-up interview notes and direct conversations with participants of learning labs.

- Clark, Kim B., & Fujimoto, Takahiro (1991). *Product Development Performance: Strategy, Organization and Change in the World Auto Industry*. Boston: Harvard Business School Press.
- Crawford, C. Merle (1983). *New Products Management*. Homewood, IL: Richard D. Irwin.
- Dertouzos, Michael L., Lester, Richard K., & Solow, Robert M. (1989). *Made in America: Regaining the Productive Edge*. Cambridge, MA: MIT Press.
- Dougherty, Deborah J. (1987) *New Products in Old Organizations: The Myth of the Better Mousetrap in Search of the Beaten Path*. Unpublished Ph.D. Dissertation, MIT Sloan School of Management.
- Dumaine, Brian (1989). How Managers Can Succeed through Speed. *Fortune*, February 13, 54-59.
- Eppinger, Steven D., Whitney, Daniel E., Smith, Robert P., & Gebala, David A. (1990). Organizing the Tasks in Complex Design Projects. In *ASME 2nd International Conference on Design Theory and Methodology*. Chicago, IL.
- Feigenbaum, Armand V. (1991). A Global Vision of Quality. *Journal for Quality and Participation*(July-Aug.), 6-9.
- Fine, Charles H., Hauser, John, Clausing, Don P., & Sachs, Emanuel M. (1989). *Developing a Manufacturing Science for Designing and Managing Systems for Total Quality Commitment*. Cambridge, MA: MIT Sloan School of Management.
- Forrester, Jay W. (1971). Counterintuitive Behavior of Social Systems. *Technology Review*, 73(3), 52-68.
- Gebala, David A., & Eppinger, Steven D. (1991). Methods for Analyzing Design Procedures. In *Third International ASME Conference on Design Theory and Methodology*. Miami, FL.
- Giancola, Paula (1992). *A Systems Thinking Perspective of the Product Development Process: A Study of Business and Academic Collaboration*. Cambridge, MA: MIT Sloan School of Management.
- Gomory, Ralph E. (1989). From the 'Ladder of Science' to the Product Development Cycle. *Harvard Business Review*(Nov.-Dec.), 99-105.
- Griffin, Abbie, & Hauser, John R. (1990). Patterns of Communication Among Marketing, Engineering and Manufacturing--A Comparison Between Two New Product Teams. *MIT Sloan School of Management Working Paper*(3209-90).
- Gupta, Ashok K., & Wilemon, David L. (1990). Accelerating the Development of Technology-Based New Products. *California Management Review*(Winter), 24-44.
- Hardin, Garrett (1977). The Tragedy of the Commons. In G. Hardin & J. Baden (Eds.), *Managing the Commons* New York: W. H. Freeman & Co. First appeared in *Science* 162:1243-1248, 1968.
- Hauser, John R., & Clausing, Don (1988). The House of Quality. *Harvard Business Review*(May-June).
- Hayes, Robert H., Wheelwright, Steven C., & Clark, Kim B. (1988). *Dynamic Manufacturing: Creating the Learning Organization*. New York: Free Press.
- Henderson, Bruce D. (1972). *Perspectives on Experience*. Boston, MA: The Boston Consulting Group.
- House, Charles H., & Price, Raymond L. (1991). The Return Map: Tracking Product Teams. *Harvard Business Review*(Jan-Feb), 92-100.
- Imai, Kenichi, Nonaka, Ikujiro, & Takeuchi, Hirotaka (1985). Managing the New Product Development Process: How Japanese Companies Learn and Unlearn. In K. Clark, R. Hayes, & C. Lorenz (Eds.), *The Uneasy Alliance: Managing the Productivity-Technology Dilemma*. Boston: Harvard Business School Press.
- Kim, Daniel H. (1989). Learning Laboratories: Designing Reflective Learning Environments. In P. Milling & E. Zahn (Eds.), *Computer-Based Management of Complex Systems*:

- Proceedings of the 1989 International Conference of the System Dynamics Society*. Berlin: Springer-Verlag.
- Kim, Daniel H. (1990). *Toward Learning Organizations: Integrating Total Quality Control and Systems Thinking*. Cambridge, MA: Pegasus Communications.
- Krubasik, Edward G. (1988). Customize Your Product Development. *Harvard Business Review*(Nov-Dec), 46-52.
- Larson, Erik W., & Gobeli, David H. (1988). Organizing for Product Development Projects. *Journal of Product Innovation Management*, 5, 180-190.
- Lorsch, Jay W., & Lawrence, Paul R. (1965). Organizing for Product Innovation. *Harvard Business Review*(Jan-Feb), 109-122.
- Maidique, Modesto Al, & Zierger, Billie Jo (1985). The New Product Learning Cycle. *Research Policy*, 299-313.
- McDonough, Edward F., & Leifer, Richard P. (1986). Effective Control of New Product Projects: The Interaction of Organizational Culture and Project Leadership. *Journal of Product Innovation Management*, 3, 149-157.
- Meade, James E. (1973). *The Theory of Economic Externalities*. Geneva: A.W. Sijthoff-Leiden.
- Meyers, Patricia W., & Wilemon, David (1989). Learning in New Technology Development Teams. *Journal of Product Innovation Management*, 6, 79-88.
- Ohmae, Kenichi (1990). *The Borderless World: Power and Strategy in the Interlinked Economy*. New York: HarperCollins.
- Ostrom, Elinor (1977). Collective Action and the Tragedy of the Commons. In G. Hardin & J. Baden (Eds.), *Managing the Commons* (pp. 179-181). New York: W. H. Freeman & Company.
- Pessemier, Edgar A. (1986). *Product Management: Strategy and Organization* (2nd ed.). Malabar, FL: Robert E. Krieger Publishing.
- Peters, Thomas J. (1991). Beyond Speed: We're All in the Fad & Fashion Business. *Industry Week*(June 3), 20-25.
- Roberts, Roger (1992). *Implementing the Learning Organization: The Application of Systems Thinking Tools to Product Development in a Manufacturing Firm*. Cambridge, MA: MIT Sloan School of Management.
- Roseneau, Milton D. (1990). *Faster New Product Development: Getting the Right Product to Market Quickly*. New York: AMACOM.
- Rosenthal, Stephen R. (1992). *Effective Product Design and Development: How to Cut Lead Time and Increase Customer Satisfaction*. Homewood, IL: Irwin.
- Roussel, Philip A., Saad, Kamal N., & Erickson, Tamara J. (1991). *Third Generation Research & Development*. Boston: Harvard Business School Press.
- Rubenstein, A.H., Chakrabarti, A.K., O'Keefe, R.D., Souder, W.E., & Young, H.C. (1976). Factors Influencing Innovation Success at the Project Level. *Research Management*(May).
- Schein, Edgar H. (1987). *Organizational Culture and Leadership*. San Francisco: Jossey-Bass.
- Senge, Peter M. (1990a). The Leader's New Work. *Sloan Management Review*, 12(1), 7-23.
- Senge, Peter M. (1990b). *The Fifth Discipline*. New York: Doubleday.
- Senge, Peter M., & Lannon, Colleen (1990). Managerial Microworlds. *Technology Review*(July).
- Smith, Preston G., & Reinertsen, Donald G. (1991). *Developing Products in Half the Time*. New York: Van Nostrand Reinhold.
- Stalk, George, Evans, Philip, & Shulman, Lawrence E. (1992). Competing of Capabilities: The New Rules of Corporate Strategy. *Harvard Business Review*(March-April), 57-69.
- Stalk, George, & Hout, Thomas M. (1990). *Competing Against Time: How Time-Based Competition is Reshaping Global Markets*. New York: The Free Press.

- Sterman, John, & Jenge, Peter M. (1991). Systems Thinking and Organizational Learning: Acting Locally and Thinking Globally in the Organization of the Future. In K. Thomas & M. Useem (Eds.), *Transforming Organizations* Oxford, England: Oxford University Press.
- Takeuchi, Hirotaka, & Nonaka, Ikujiro (1986). The New New Product Development Game. *Harvard Business Review*(Jan-Feb), 137-146.
- Ulrich, Karl , Sartorius, David, Pearson, Scott, & Jadiela, Mark (1993). Including Value of Time in Design for Manufacturing Decision Making. *Management Science*, 39(4), April.
- Urban, Glen L., & Hauser, John R. (1980). *Design and Marketing of New Products*. Englewood Cliffs, NJ: Prentice-Hall.
- Vygotsky, Lev (1989). *Thought and Language*. Cambridge, MA: MIT Press.

Chapter 7

Discussion and Summary

"[L]earning organizations may be a tool not just for the evolution of organizations, but for the evolution of intelligence."

—Peter M. Senge

7.0 CONTRIBUTIONS

The contributions of this dissertation are three-fold:

1. It advances the theory of organizational learning by developing a framework that makes an explicit link with individual learning and by specifying the mechanism—mental models—through which the organization can absorb individual learning.
2. It refines the development of tools for advancing organizational learning by providing a methodology for making implicit individual models explicit and mapping them into a representation that is accessible to others.
3. It applies the framework and methodology to two substantive areas—TQM and product development management—providing interesting insights in both areas.

These contributions are of particular relevance to the fields of operations management, system dynamics, and organizational studies.

7.1 CONTRIBUTIONS TO OPERATIONS MANAGEMENT

The contributions of this work that may be of interest to the Operations Management community are the insights gained in applying the framework and methodology of organizational learning to two substantive areas—TQM and product development management. In the TQM setting, the methodology was used to map the learnings of individuals about their company's TQM false start experience which led to the emergence of potential common themes that may be applicable to other organizations.

7.1.1 Total Quality Implementation Setting

In the Total Quality Implementation setting, the methodology was used to analyze several companies' experiences in trying to implement TQM (although the learnings could probably apply to any change effort). Out of that process, they obtained a rich set of systemic insights about what happened.

The systems archetypes provided a common structure for analyzing the stories. In particular, we discovered that one archetype, Limits-to-Success, was prevalent, playing itself out in various ways involving issues of training capacity, management commitment and attention, and others. This suggests that a useful step to take at the beginning of any implementation effort is to identify *in advance* all possible sources of organizational limits to success. Plans can then be put in place to anticipate or eliminate the barriers one is likely to encounter as the implementation rolls out through the organization.

As the research study is expanded to include sites at different companies and those who are at different phases of implementation, a pattern of dominant archetypal dynamics may emerge that could be useful for managing later stages of an implementation. For example, Limits-to-Success

may be of predominant importance in the initial stages, while Shifting-the-Burden may become more prevalent in the middle of the process. The initial results of this study provide preliminary evidence for the usefulness of the methodology outlined in Chapter 4.

7.1.2 Product Development Management Setting

In the product development management setting, the research took an approach different from that used in the TQM setting. The researcher did not go in and interview people after the fact, but was actually part of the team that worked together to look at the process. Several valuable lessons emerged from this effort.

One of the key issues in product development is how to compress the cycle time. There are many techniques, but one fundamental concern is managerial: In a complex development effort, how can the organization best manage the interconnections of all the sub-teams, to compress the cycle time and help the different teams work smoothly together. In this setting, the bottleneck Clark and Fujimoto describe is the quality and frequency of communication in the overlapping activities, e.g., product engineering and process engineering.

The use of action inquiry exercises and the approach of mapping mental models with systems archetypes helped product development managers at Cars, Inc. to begin identifying systemic causes of why they couldn't get important information on a timely basis. Through the process change that occurred over time, they were able to get information about problems earlier. The use of the archetypes also provided the means to gain consensus around an important issue with which the team is struggling.

A substantive outcome of this research was the pervasiveness of the Tragedy-of-the-Commons archetype in product development at Cars. The principle of the Tragedy-of-the-Commons in the product development process is that the leverage usually lies at a higher level than the one at which the problem occurs. Therefore, each person needs to understand at what level he or she is operating in a particular ToC, and at what level does the situation need to be resolved.

In a Tragedy-of-the-Commons archetype, there are basically two options: (1) find a collective solution at the individual level, or (2) give power to a higher level to allocate or otherwise control the consumption of the shared resource. In general, by the very nature of the system of rewards and incentives that are in place in a ToC structure, finding a solution at the individual level is usually extremely difficult. Therefore, the second lever is to defer the managing of the commons to a higher level of management that can weigh the global implications of all the individual local actions. The striking outcome of team training in systems thinking was the collective identification and agreement on when autonomous decision makers should voluntarily relinquish negotiating control over decisions critical to their sub-organization's interests and to the performance of the sub-systems for which they were responsible.

The study result suggests that product development efforts, in general, could benefit by identifying all potential Tragedy-of-the-Commons issues that the team may encounter at the beginning of a project. This can help the program managers flag potential problems early on in the process and design management systems to address them.

7.2 CONTRIBUTION TO SYSTEM DYNAMICS:

The field of system dynamics, its methodology, principles, and practices, can make a significant contribution to organizational learning, especially in helping to understand dynamic complexity. Intuitively, managers find value in the approach because of its ability to capture complex organizational dynamics, but the methodology can be very difficult to put into practice. We gained some important insights in our attempt to translate the methodology into more digestible pieces for mass consumption.

One lesson is that the basic pen-and-paper tools like causal loop diagrams can be made easier to use by delineating a step-by-step process for constructing them. Computer modeling expertise, in and of itself, is not the ultimate goal. I liken the development of system dynamics to that of TQM. The goal of TQM efforts was not to have people attain PhD's in statistics, or to even become practicing statisticians. Simple tools which were based on statistical theory and principles were developed and then used by managers to look at real world problems, and see where the principles might be relevant. Over time, a methodology that people could easily learn and use was developed.

Although TQM and system dynamics originated at about the same time (circa 1950's), today TQM is practiced worldwide, compared to the still relatively unfamiliar field of system dynamics. That continues to be true even though system dynamics arguably has greater potential for increasing insight into managing complex organizational dynamics. This work takes a beginning step toward realizing that potential.

7.2.1 Rigor, Relevance, and Usability

In any academic field, there is always an ongoing tension between rigor vs. relevance. While considering relevance, one must also underscore usability.

A theory can be relevant but unusable. Through the process detailed particularly in Chapter 5, relatively untrained graduate students were able to use the mapping tools to analyze TQM implementations and develop relatively sophisticated causal loop diagrams. This is a step forward for the potential to broaden the audience of system dynamics.

Clearly, there is still much to be done. We should continue to develop a methodology that non practitioners can use to look at and diagnose organizational problems from a systemic perspective. People may then be able to transfer their structural understanding into another setting and identify, not only isolated individual events that happened in a particular TQM implementation, but also the systemic reasons for the events.

7.2.2 Levels of Use of Systems Archetypes

The product development setting work brought out three levels of use for systems archetypes, distinctions that had never explicitly been made before. We discovered at least three distinct levels of use, as well as the need for managers to feel comfortable about their ability to use the archetypes at each level, rather than become frustrated because they can't master them or use them the 'right' way. The three levels are:

- 1. Systems archetypes as metaphorical stories**
- 2. Systems archetypes as structural pattern templates**
- 3. Systems archetypes as dynamic scripts (or theories)**

First, people acquire the ability to use the archetypes as a simple story *metaphor*. They understand the basic lessons, key elements, and the outcomes or the high leverage actions that are embodied in each one. This allows people to go into a situation, identify the storyline that is at work,

explore the implications of that storyline, and gain some initial understanding of the problem under study.

Level two use occurs as people start to map a particular problem to the relevant archetype *template*, by taking a story and pattern-matching it to fit an appropriate archetype—identifying each of the core categories, e.g., problem symptom, symptomatic fix, etc. in a template. At this point, people are becoming explicit with the naming of variables, linking them with directed arcs, distinguishing between balancing and reinforcing loops, and understanding conceptually why one is a balancing loop, and another is a reinforcing loop.

At the third level, people are quite familiar with not only the loops, but also the signing of the links in s's or o's, and they are able to coherently construct each of the loops. The archetypes serve as *dynamic scripts* (or theories) which guide their inquiry in different settings, with the ability to transfer insight from one setting to the next. People can go beyond identifying the core categories in the archetype's template and begin building on to it with additional loops. The archetype serves as a core script or theory that guides their inquiry to better understand their system.

These three levels of distinction in the use of archetypes are important, because once they have been mastered, its a matter of practice to become more comfortable with multiple layers of loops. After those skills are acquired, working with the archetypes involves more gradations of differences and complexity.

This discovery of the three levels of use came from the work with the learning team at Cars, Inc.:

IC: What you want to be careful of, is that the archetypes --you can spend days, hours, trying to figure out what the right one is, and that's a good thing, getting everybody to talk, but it's sometimes difficult to do, difficult to conceptualize and

what we found is that we spent time--we all knew how to articulate the problem in words, but couldn't figure out how to draw the picture. And you spend a lot of time trying to get the arrows going in the right direction, that can be not productive. Especially if everybody can just talk a problem through, at some common level of understanding.

SSM D: We try to understand the basic archetypes, the examples from the book and the stuff we've done at core competency, so we have an understanding of what the basic problem statements are, and we use those to articulate the problems vs. trying to draw our own charts in a lot of cases. So, yes, we do not draw a lot of archetypes, but do we use the archetype methodology? I think we do.

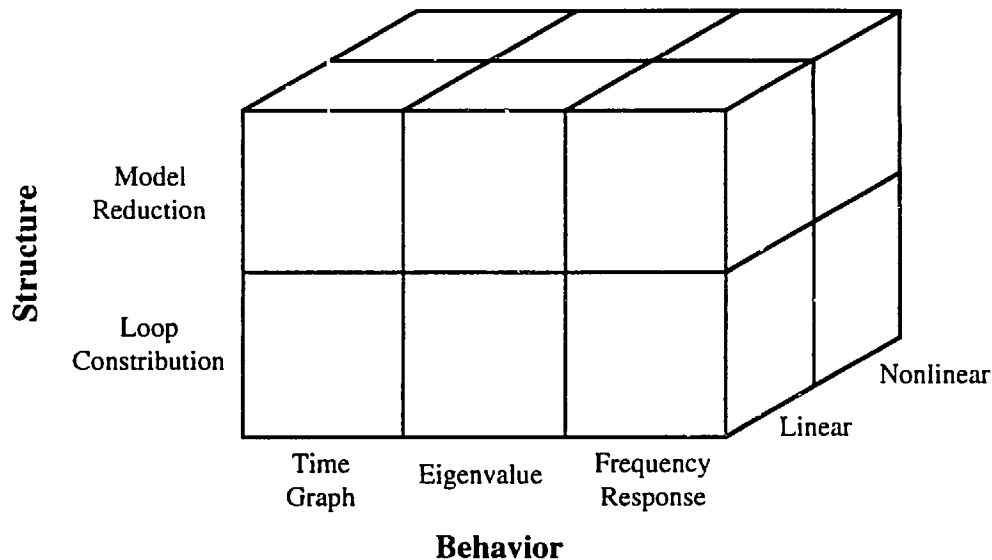
Our intention with this work and further study is to improve the causal looping methodology continually for mass consumption. The diagnostic approach demonstrated in Chapter 5 provides a systematic way of interviewing and mapping one's experiences, while the product development setting described in Chapter 6 is oriented toward real time problem articulation. Both are attempts at making the methodology more usable for mass consumption.

7.2.3 Model Decomposition and Identification of New Archetypes

The area of model decomposition offers some exciting possibilities in its application to causal loop diagrams. Another way to look at the issue of model decomposition is to see it as a search for dominant structure. Richardson (1986) presents a classification scheme that shows how the different approaches map into three different types of behavior analysis: time graph, eigenvalue, and frequency response (see Figure 7.1). The diagram decomposition guidelines used in Chapter 4 were based on model reduction work that used linear eigenvalue analysis (Eberlein, 1986; 1989). Although the methods involving eigenvalues and frequency response require mathematical representations in order to apply them directly, the time graph behavior analysis can be applied to causal loop structural analysis.

Efforts to explore and develop better and more efficient ways of "decomposing" large diagrams into smaller ones, while retaining the

“dominant structures” in the original diagram, may be very fruitful. A systematic process of mapping and decomposing may lead to the discovery of new archetypes as many diagrams from many studies are decomposed into their minimal structural form. These new archetypes could then be translated into computer models for further testing and validation. Through this process, system dynamicists may then be able to systematically build a library of these *generic structures* which can provide a systemic structural theory for understanding a significant majority of organizational dynamics.



Classification Scheme for Determining Dominant Model Structure
(source: Richardson, 1986)

Figure 7.1

7.3 CONTRIBUTIONS TO ORGANIZATIONAL LEARNING

In this dissertation, we developed a conceptual framework and a theory for linking individual learning to organizational learning. The mechanism identified for achieving that transfer was through making implicit mental models through the use of representational systems. Several criteria were identified as important requirements for a representational system to capture

dynamic complexity: that it be abstract, generic, runnable, theory-driven, and testable. A methodology and research process was proposed for making implicit mental models explicit using grounded theory-building and system dynamics causal loop diagrams.

The two case settings, TQM and product development, illustrated the use of the methodology and produced several insights as described in Chapters 5 and 6. The proposed framework and methodology, in addition to providing methods and tools for mapping, can also help shift people's perspective by virtue of using the methodology itself because it encourages people to look at their experience more systemically. Organizationally, if this perspective is shared, it has the potential of changing the organization's *Weltanschauung* from an event-causes-event orientation to a systemic understanding of how structure influences behavior.

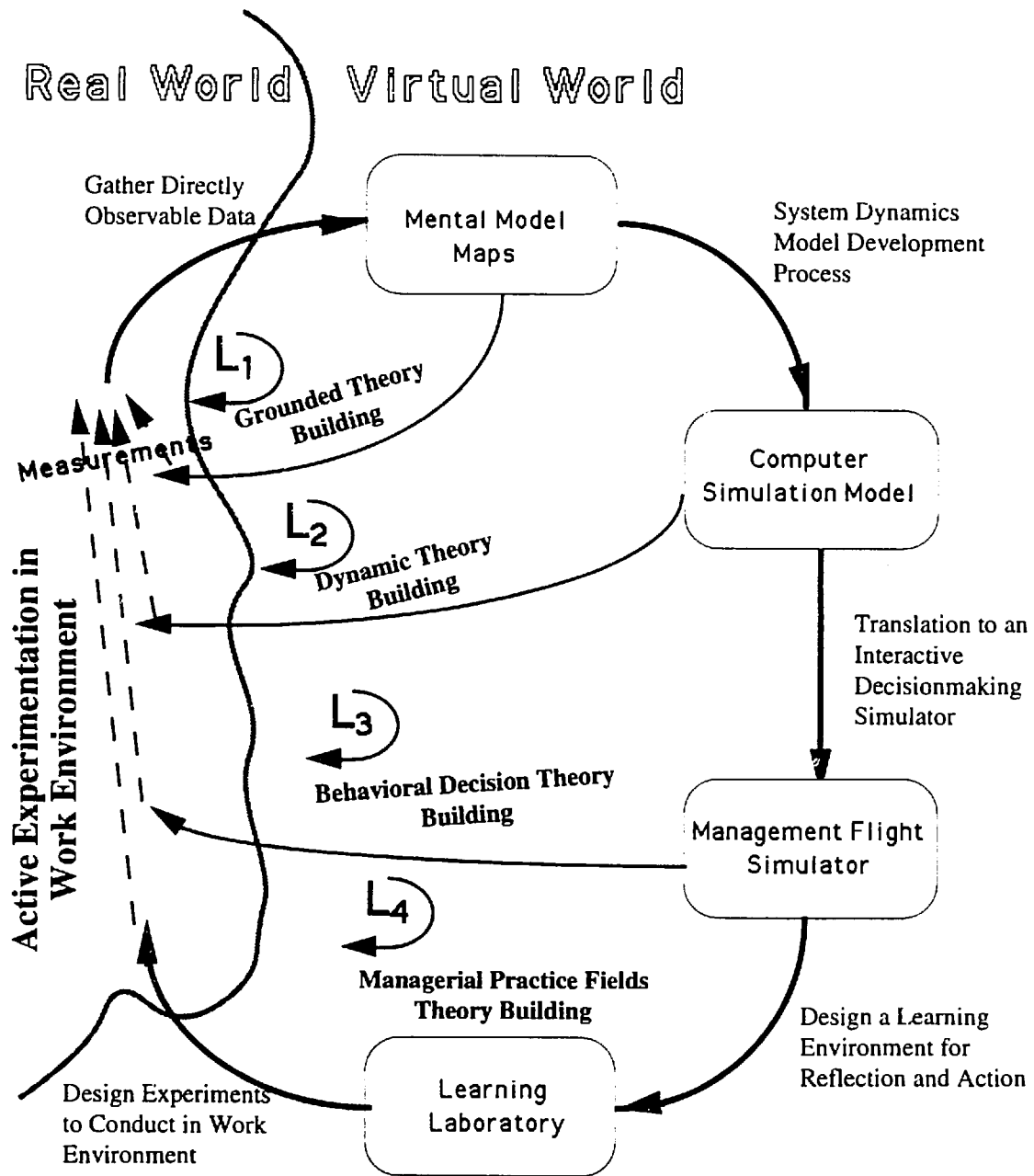
Although "the basic processes that contribute to the occurrence, breadth, and depth of organizational learning depends on organizational memory," relatively little empirical work has been done on the construct of organizational memory and shared mental models, (Huber, 1991, p.107). Further work is needed for a better understanding of the role of mental models in individual and organizational learning, especially the types of mental models that are appropriate for representing dynamic complexity, the methods with which to capture the understanding of such complexity, and the means through which new learnings can be transferred to the whole organization. The framework of organizational learning and the methodology developed in this dissertation is proposed to serve as a guide in pursuing these goals.

7.4 FUTURE DIRECTIONS

Until recently, on-the-job learning appeared to provide adequate preparation for managing in a world where change, though accelerating, still occurred across generations rather than within generations. Schön (1971) points out that a major disruption occurred when the pace of change crossed into the intra-generational state—lessons learned became obsolete within the same generation. As the world grows increasingly more complex, problems take longer to solve and the proposed solutions have shorter lives. In fact, Ackoff (1981) posits that solutions are often stillborn because problems change so rapidly that solutions, when found, are often no longer relevant.

One consequence of such rapid change is that managers are forced to make decisions with equal rapidity. However, the complexity of the problems makes it imperative that managers take more time to reflect on their decisions. How, then, can a manager speed up and slow down at the same time? How can one manage in a world where experience is no longer the best—or even adequate—teacher? Where change makes yesterdays' lessons obsolete? How can organizations remain viable given this paradoxical dilemma? At a more macro level, how can we learn about the tremendous messes we face today and implement solutions in time? These are the kinds of questions which we are trying to address at the MIT Organizational Learning Center (OLC).

The project at Cars, Inc. is one of several projects being conducted at the OLC which attempt to design and embed a learning process within an organization. We are trying to integrate a variety of research methods and the building of theory at various levels (See Figure 7.2). Although a lot of the OLC work lies in the (virtual) world of ideas, model formulation, design of flight simulators and learning labs, an important aspect of the OLC's theory



Organization Theory Building Cycle

Figure 7.2

building process is its close integration with active experimentation in the field.

The OLC research efforts can be viewed in terms of four different, but inter-related, learning loops which guide all projects—grounded theory

building, dynamic theory building, behavioral decision theory building, and managerial practice field theory building. A brief description of each follows.

Grounded Theory Building Loop L₁: This loop represents the field research tradition of building theory based on a firm grounding on observable data. Action science, clinical research, and ethnography are probably the most relevant approaches. It is an inductive process of building a general theory based on intensive field study. Mapping tools such as systems archetypes and causal loop diagram can play an important role in building grounded theory as demonstrated in this dissertation.

Dynamic Theory Building Loop L₂: This loop includes the traditional system dynamics model building process of data collection, model formulation, testing, revising, and validation. It includes some of the work represented by L₁, and builds on it in a more rigorous fashion. Operations management and system dynamics computer models are included in this loop, although they use different methods.

Behavioral Decision Theory Building Loop L₃: In terms of work being done at the OLC, Sterman's (1989) work on dynamic decisionmaking is a good representation of this process. Using interactive computer simulators to study how managers make decisions in a laboratory setting is one part of the work; linking the impact of those studies to actions in the work place is another (Bakken, 1993; Diehl, 1992; Kampmann, 1992).

Managerial Practice Field Theory Building Loop L₄: This loop describes the process of helping an organization engage in its own process of developing a theory about itself, starting with a particular focus area, like product development. The OLC pilot projects can be viewed as a vehicle for helping managers and researchers work collaboratively on this process.

All four learning loops are important to organizational theory building. In the short term, no single project is likely to adequately address all four loops at once, but each project should be focusing on at least one of these loops. The two case settings presented in this dissertation were most firmly rooted in the grounded theory building loop, although the product development case also utilized some of the work that involved the other loops.

7.5 REFLECTIONS

Given today's pace of change and organizational complexity, managers need to be competent in applying the research skills of a scientist to better understand their organizations. The old paradigm of experiments in organizations being fed into research institutions that receiving the output and feed the results back into the organizations is no longer adequate. Intra-generational change means that the research cycle must be done within a much shorter time frame, otherwise solutions (in the form of research results) will be stillborn—the problems which they were addressing will no longer be relevant.

7.5.1 Managers' New Roles: Researcher and Theory-Builder

The dichotomy between manager and researcher must end because the pace of change is such that one can no longer separate the two functions—managers must wear both hats simultaneously. Ed Baker's (1989)¹ proposal that the CEO's new role should be that of Head of Research and Development

¹E.M. Baker. "The Chief Executive Officer's Role in Total Quality: Preparing the Enterprise for Leadership in the New Economic Age," Conference on Quality, Madison, Wisconsin, April 19-21, 1989.

for the Enterprise is exactly the type of leadership needed for supporting such a shift.

Managers also need to become *theory-builders* within their own organizations. They must create new frameworks within which they continually test their strategies, policies, and decisions to inform them of improvements on the organization's design. It is no longer sufficient to apply generic theories and frameworks like band-aids to one's own specific issues. Managers must take the best of the new ideas around and then build a workable theory for their own organization. As theory-builders, managers must have an intimate knowledge of how their organization works together as a whole. But they also require some guiding theory and methodology to make sense of their experience and learning.

There is no "golden formula" that will hold for all time, or even for one's tenure in a present position. Companies who lived by the learning curve theory almost died by the learning curve theory (as in the case of Texas Instruments and the personal computer debacle). Others who followed the BCG business portfolio theory also had their share of problems by either giving up entire markets or not taking full advantage of synergies among their different businesses. Theory building should not be done as an academic exercise but as a process grounded in reality which continually helps provide a framework for interpreting one's competitive environment.

7.5.2 Collaborative Research/Practitioner Model

The research partnership at the OLC is, in part, a mutual mentoring process. It would be helpful for researchers to move away from the detached, purely rationalistic perspective and recognize the inherently chaotic nature of organizations. In this spirit, the collaboration engendered at Cars, Inc. is one

of mutual mentorship, one of understanding each role better so that each can grow to become a better manager-researcher or researcher-manager.

This type of partnering and collaboration requires significant immersion into both corporate and academic worlds, producing results that can be interpreted in each other's culture and language rather than each being sufficient only to its own separate community. In this kind of research, the messenger has to be consistent with the message. The work would not be as effective if the researchers were not always, in their best efforts, trying to model the principles and practices that are part of the process. Strauss has his own reflections on the relationship between the researcher and the research which resonates with my own:

We should add that while much research involves routine operations and can at times be boring, assuredly also at its most creative it is exciting, fun, challenging, although sometimes extremely disturbing and painful. This means that researchers, as workers, **can and should care very deeply about their work** [emphasis added]—not being simply possessive about its products or jealous of their research reputations, but find deep and satisfying meaning in their work. (Strauss, 1987, p. 9)

It is to that spirit of deep caring that this dissertation is dedicated.

Chapter 7 References

- Ackoff, Russell L. (1981). *Creating the Corporate Future: Plan or Be Planned For*. New York: Wiley & Sons.
- Baker, Edward M. (1989). The Chief Executive Officer's Role in Total Quality: Preparing the Enterprise for Leadership in the New Economic Age. In *Conference on Quality*. Madison, WI.
- Bakken, Bent (1993). *Learning and Transfer of Understanding in Dynamic Decision Environments*. Unpublished dissertation, MIT Sloan School of Management.
- Diehl, Ernst (1992). *Effects of Feedback Structure on Dynamic Decision Making*. Unpublished dissertation, MIT Sloan School of Management.
- Eberlein, Robert L. (1986). Identifying and Displaying Important Feedback Paths. In *International Conference of the System Dynamics Society*, (pp. 159-177). Sevilla: System Dynamics Society.
- Eberlein, Robert L. (1989). Simplification and Understanding of Models. *System Dynamics Review*, 5(1, Winter), 51-68.
- Huber, George, P. (1991). Organizational Learning: The Contributing Processes and the Literature. *Organization Science*, 2(1), 88-115.

- Kampmann, Christian E. (1992). *Feedback Complexity and Market Adjustment: An Experimental Approach*. Unpublished Ph.D. thesis, MIT Sloan School of Management.
- Richardson, George P. (1986). Dominant Structure. *System Dynamics Review*, 2(1, Winter), 68-75.
- Schön, Donald (1971). *Beyond the Stable State*. London: Temple Smith.
- Sterman, John D. (1989). Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment. *Management Science*, 35(3), 321-339.
- Strauss, Anselm L. (1987). *Qualitative Analysis for Social Scientist*. Cambridge: Cambridge University Press.

FULL REFERENCE LIST

- Ackoff, Russell L. (1981). *Creating the Corporate Future: Plan or Be Planned For*. New York: Wiley & Sons.
- Ackoff, Russell. *The Second Industrial Revolution*. Unpublished manuscript.
- Adler, Paul S., Riggs, Henry E., & Wheelwright, Steven C. (1989). Product Development Know-How; Trading Tactics for Strategy. *Sloan Management Review*(Fall), 7-17.
- Ackoff, Russell L. (1981). *Creating the Corporate Future: Plan or Be Planned For*. New York: Wiley & Sons.
- Allen, Thomas J. (1977). *Managing the Flow of Technology*. Cambridge, MA: MIT Press.
- Allen, Thomas J. (1980). R&D Performance as a Function of Internal Communication, Project Management, and the Nature of Work. *IEEE Transactions on Engineering Management*(Feb), 2-12.
- Ancona, Deborah G., & Caldwell, David E. (1992). Cross-Functional Teams: Blessing or Curse for New Product Development. In T. Kochan & M. Useen (Eds.), *Transforming Organizations* Oxford: Oxford University Press.
- Anderson, John R. (1983). *The Architecture of Cognition*. Cambridge, MA: Harvard University Press.
- Argyris, Chris (1990). *Overcoming Organizational Defenses*. Wellesley, MA: Allyn-Bacon.
- Argyris, Chris, & Schön, Donald (1978). *Organizational Learning: A Theory of Action Perspective*. Reading, MA: Addison-Wesley.
- Argyris, Chris, Putnam, R., & Smith, D.M. (1985). *Action Science*. San Francisco: Jossey-Bass.
- Baden, John (1977). A Primer for the Management of Common Pool Resources. In G. Hardin & J. Baden (Eds.), *Managing the Commons* (pp. 137-146). New York: W. H. Freeman & Company.
- Bailyn, Lotte (1984). Autonomy in the Industrial R&D Lab. *MIT Sloan School of Management, Working Paper 1592-84*.
- Baker, Edward M. (1989). The Chief Executive Officer's Role in Total Quality: Preparing the Enterprise for Leadership in the New Economic Age. In *Conference on Quality*. Madison, WI.
- Baker, Edward M. (1989). The Evolution to Total Quality Excellence. *Dialogue, Winter*, 7-13.
- Bakken, Bent (1989). Learning in Dynamic Simulation Games: Using Performance as a Measure. In P. Milling & E. Zahn (Eds.), *Computer -Based Management of Complex Systems: Proceedings of the 1989 International Conference of the System Dynamics Society*. Berlin: Springer-Verlag.
- Bakken, Bent (1993). *Learning and Transfer of Understanding in Dynamic Decision Environments*. Unpublished Dissertation, MIT Sloan School of Management.
- Balz, William P., & Garberding, Scott R. (1993) *A System Dynamics Analysis of Total Quality Management Implementation Issues*. Unpublished Master's thesis, MIT Sloan School of Management.
- Beer, Stafford (1972). *Brain of the Firm*. New York: John Wiley and Sons.
- Beer, Michael (1980). *Organization Change and Development: A Systems View*. Santa Monica, CA: Goodyear Publishing.
- Bostrom, A., Fischhoff, B., & and Morgan, G. M. (1992). Characterizing Mental Models of Hazardous Processes: A Methodology and an Application to Radon. *Journal of Social Issues*, 48(4), 85-100.
- Boulding, Kenneth E. (1956). General systems theory—The skeleton of science. *Management Science*, 2(3), 197-208.
- Bower, G. H., Black, J. B., & Turner, T. J. (1979). Scripts in Memory for Text. *Cognitive Psychology*, 11, 177-220.
- Brassard, Michael (1989). *Memory Jogger Plus+*. Methuen, MA: Goal/QPC.
- Brown, Jim , & Tse, Scott S. F. (1992). *A System Dynamics Analysis of Total Quality Management Implementation False Starts*. Cambridge, MA: MIT Sloan School of Management.
- Bruner, Jerome (1966). *Toward a Theory of Instruction*. New York: W.W. Norton.
- Burchill, Gary W. (1993). *Concept Engineering: An investigation of TIME vs. MARKET orientation in product concept development*. Unpublished dissertation, MIT Sloan School of Management.

- Bushe, Gervase R., & Shani, A.B.(Rami) (1991). *Parallel Learning Structures*. Reading, MA: Addison-Wesley Publishing Co.
- Business Week* (October 25, 1991). Where Did They Go Wrong?
- Clark, Kim B., & Fujimoto, Takahiro (1990). The Power of Product Integrity. *Harvard Business Review*(Nov.-Dec.), 107-118.
- Clark, Kim B., & Fujimoto, Takahiro (1991). *Product Development Performance: Strategy, Organization and Change in the World Auto Industry*. Boston: Harvard Business School Press.
- Cohen, Michael D. (1991). Individual Learning and Organizational Routine: Emerging Connections. *Organization Science*, 2(1), 135-139.
- Cohen, Morris R., & Nagel, Ernest (1936). *An Introduction to Logic and the Scientific Method*. New York: Harcourt, Brace & Co.
- Cowan, D.A. (1991). Effect of Decision Making Styles and Contextual Experience on Executive Descriptions of Organizational Problem Formulation. *Journal of Management Studies*, 28(September), 463-83.
- Crowford, C. Merle (1983). *New Products Management*. Homewood, IL: Richard D. Irwin.
- Crosby, Philip B. (1979). *Quality is Free: The Art of Making Quality Certain*. New York: McGraw-Hill.
- Cyert, Richard M., & March, James G. (1963). *A Behavioral Theory of the Firm*. Englewood Cliffs, NJ: Prentice-Hall.
- Daft, R.L., & Weick, K.E. (1984). Toward a model of organizations as interpretation systems. *Academy of Management Review*, 9(2), 284-295.
- de Geus, Arie (1988). Planning as Learning. *Harvard Business Review*, March-April, 70-74.
- deKleer, Johan, & Brown, John Seely (1983). Assumptions and Ambiguities in Mechanistic Mental Models. In G. D. & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Erlbaum Associates.
- Deming, Edwards (1992). *Quality, Productivity and Competitive Position*. Ford Quality Education and Training Center.
- Dertouzos, Michael L., Lester, Richard K., & Solow, Robert M. (1989). *Made in America: Regaining the Productive Edge*. Cambridge, MA: MIT Press.
- Dewey, John (1926). *Experience and Nature*. Chicago: Open Court Publishing.
- Diehl, Ernst (1989). *A Study of Human Control in Stock-Adjustment Tasks*. Unpublished manuscript, MIT Sloan School of Management.
- Diehl, Ernst (1992). *Effects of Feedback Structure on Dynamic Decision Making*. Unpublished dissertation, MIT Sloan School of Management.
- Dougherty, Deborah J. (1987) *New Products in Old Organizations: The Myth of the Better Mousetrap in Search of the Beaten Path*. Unpublished Ph.D. Dissertation, MIT Sloan School of Management.
- Drucker, Peter F. (1990). Emerging Theory of Manufacturing. *Harvard Business Review*(May-June), 94-102.
- Dumaine, Brian (1989). How Managers Can Succeed through Speed. *Fortune*, February 13, 54-59.
- Dumaine, Brian (1989). How Managers Can Succeed through Speed. *Fortune*, February 13, 54-59.
- Duncan, Robert B. (1974). Modifications in decision structure in adapting to the environment: Some implications for organizational learning. *Decision Sciences*, 5, 705-725.
- Dutton, Jane E., & Jackson, Susan E. (1987). Categorizing Strategic Issues: Links to Organizational Action. *Academy of Management Review*, 12(1, January), 76-90.
- Eberlein, Robert L. (1986). Identifying and Displaying Important Feedback Paths. In *International Conference of the System Dynamics Society*, (pp. 159-177). Sevilla: System Dynamics Society.
- Eberlein, Robert L. (1989). Simplification and Understanding of Models. *System Dynamics Review*, 5(1, Winter), 51-68.
- Ehrler, Steven M., & Jansen, Eckart W. (1993) *System Dynamics as an Organizational Diagnostic for Total Quality Management Implementation*. Unpublished Master's thesis, MIT Sloan School of Management.
- Eppinger, Steven D., Whitney, Daniel E., Smith, Robert P., & Gebala, David A. (1990). Organizing the Tasks in Complex Design Projects. In *ASME 2nd International Conference on Design Theory and Methodology*. Chicago, IL:

- Ernst & Young (1991). *International Quality Study*.
- Feigenbaum, Armand V. (1991). A Global Vision of Quality. *Journal for Quality and Participation*(July-Aug.), 6-9.
- Feldman, Daniel C. (1984). The Development and enforcement of group norms. *Academy of Management Review*, 9(1), 47-53.
- Fine, Charles H. (1986). Quality Improvement and Learning in Productive Systems. *Management Science*, 32 (10), 1301-1315.
- Fine, Charles H. (1988). A Quality Control Model with Learning Effects. *Operations Research*, 36(3, May-June), 437-444.
- Fine, Charles H., Hauser, John, Clausing, Don P., & Sachs, Emanuel M. (1989). *Developing a Manufacturing Science for Designing and Managing Systems for Total Quality Commitment*. Cambridge, MA: MIT Sloan School of Management.
- Fiol, C.M., & Lyles, M.A. (1985). Organizational learning. *Academy of Management Review*, 10(4), 803-813.
- Fiske, Susan T., & Taylor, Shelley E. (1984). *Social Cognition*. New York: Random House.
- Forrester, Jay W. (1961). *Industrial Dynamics*. Cambridge, MA: Productivity Press.
- Forrester, Jay W. (1971). Counterintuitive Behavior of Social Systems. *Technology Review*, 73(3), 52-68.
- Forrester, Jay W. (1971). *World Dynamics*. Cambridge, MA: Productivity Press.
- Forrester, Jay W. (1973). Confidence in Models of Social Behavior--with Emphasis on System Dynamics Models. *MIT Sloan School of Management unpublished working paper*.
- Forrester, Jay W. (1979). *Information Sources for Modeling the National Economy*. Cambridge, MA: System Dynamics Group, MIT Sloan School of Management.
- Forrester, Jay W. (1993). Systems Dynamics and the Lessons of 35 Years. In K. B. DeGreene (Eds.), *Systems-Based Approach to Policymaking*. Norwell, MA: Kluwer Academic Publishers.
- Freire, Paulo (1974). *Pedagogy of the Oppressed*. New York: Continuum.
- Garnham, Alan (1987). *Mental Models: A Representation of Discourse and Text*. New York: Halsted Press.
- Gebala, David A., & Eppinger, Steven D. (1991). Methods for Analyzing Design Procedures. In *Third International ASME Conference on Design Theory and Methodology*. Miami, FL.
- Gentner, D., & Stevens, A.L. (1983). *Mental Models*. Hillsdale, NJ: Erlbaum.
- Gentner, Dedre, & Gentner, Donald (1983). Flowing Waters or Teeming Crowds: Mental Models of Electricity. In D. Gentner & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Erlbaum Associates.
- Giancola, Paula (1992). *A Systems Thinking Perspective of the Product Development Process: A Study of Business and Academic Collaboration*. Cambridge, MA: MIT Sloan School of Management.
- Gioia, Dennis A., & Poole, Peter A. (1984). Scripts in Organizational Behavior. *Academy of Management Review*, 9(3), 449-459.
- Glaser, B.G. (1978). *Theoretical Sensitivity: Advances in the Methodology of Grounded Theory*. Mill Valley, CA: Sociology Press.
- Glaser, B.G., & Strauss, A.L. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Chicago: Aldine.
- Goal/QPC (1985). *Memory Jogger*. Methuen, MA: Goal/QPC.
- Gomory, Ralph E. (1989). From the 'Ladder of Science' to the Product Development Cycle. *Harvard Business Review*(Nov.-Dec.), 99-105.
- Griffin, Abbie, & Hauser, John R. (1990). Patterns of Communication Among Marketing, Engineering and Manufacturing--A Comparison Between Two New Product Teams. *MIT Sloan School of Management Working Paper*(3209-90).
- Gupta, Ashok K., & Wilemon, David L. (1990). Accelerating the Development of Technology-Based New Products. *California Management Review*(Winter), 24-44.
- Hackman, J. R. , & Oldham, Greg R. (1976). Motivation through the Design of Work: Test of a Theory. *Organizational Behavior and Human Performance*, 16, 250-279.
- Hardin, Garrett (1977). The Tragedy of the Commons. In G. Hardin & J. Baden (Eds.), *Managing the Commons* New York: W. H. Freeman & Co. First appeared in *Science* 162:1243-1248, 1968.

- Hauser, John R., & Clausing, Don (1988). The House of Quality. *Harvard Business Review*(May-June).
- Hayes, Robert H., Wheelwright, Steven C., & Clark, Kim B. (1988). *Dynamic Manufacturing: Creating the Learning Organization*. New York: Free Press.
- Hedberg, Bo L.T., Nystrom, Paul C. , & Starbuck, William H. (1976). Camping on seesaws: Prescriptions for a self-designing organization. *Administrative Science Quarterly*, 21, 41-65.
- Hedberg, Bo. (1981). How organizations learn and unlearn. In P. C. Nystrom & W. H. Starbuck (Eds.), *Handbook of Organizational Design*. London: Oxford University Press.
- Hempel, Carl Gustav (1965). *Aspects of Scientific Explanation, and other essays in the philosophy of science*. New York: The Free Press.
- Henderson, Bruce D. (1972). *Perspectives on Experience*. Boston, MA: The Boston Consulting Group.
- Hilgard, Ernest R. , & Bower, Gordon H. (1966). *Theories of Learning*. New York: Appleton-Century-Crofts.
- House, Charles H., & Price, Raymond L. (1991). The Return Map: Tracking Product Teams. *Harvard Business Review*(Jan-Feb), 92-100.
- Hrebiniak, Lawrence G. , & Joyce, William F. (1985). Organizational adaptation: Strategic choice and environmental determinism. *Administrative Science Quarterly*, 20, 226-240.
- Huber, George, P. (1991). Organizational Learning: The Contributing Processes and the Literature. *Organization Science*, 2(1), 88-115.
- Hutchins, D. (1986). Having a Hard Time with Just-in-Time. *Fortune*(June 9), 64-66.
- Imai, Kenichi, Nonaka, Ikujiro, & Takeuchi, Hirotaka (1985). Managing the New Product Development Process: How Japanese Companies Learn and Unlearn. In K. Clark, R. Hayes, & C. Lorenz (Eds.), *The Uneasy Alliance: Managing the Productivity-Technology Dilemma* Boston: Harvard Business School Press.
- Isaacs, William (1992). The Ladder of Inference. In Cambridge, MA: MIT Organizational Learning Center.
- Iseberg, Daniel J. (1986). Group polarization: a critical review and meta-analysis. *Journal of Personality and Social Psychology*, 50(6), 1141-1151.
- Ishikawa, Kaoru. (1982). *Guide to Quality Control*. Ann Arbor, MI: UNIPUB.
- Ishikawa, Kaoru (1985). *What is Total Quality Control?* Englewood Cliffs, NJ: Prentice-Hall.
- Jaques, Elliott. (1989). *Requisite Organization*. Arlington, VA: Cason-Hall Associates.
- Johnson-Laird, P.N. (1983). *Mental Models*. Cambridge, MA: Harvard University Press.
- Juran, Joe M. (1982). *Upper Management and Quality*. Wilton, CT: Juran Institute, Inc.
- Kampmann, Christian E. (1992). *Feedback Complexity and Market Adjustment: An Experimental Approach*. Unpublished Ph.D. thesis, MIT Sloan School of Management.
- Kendrick, John J. (1992). Companies Continue to Embrace Quality Programs, but Has Total Quality Generated More Enthusiasm than Results? *Quality*, 31(5 (May)), 13.
- Kepner, Charles H. , & Tregoe, Benjamin B. (1965). *The Rational Manager*. New York: McGraw-Hill.
- Kidder, Louise H., & Judd, Charles M. (1986). *Research Methods in Social Relations* (Fifth ed.). New York: Holt, Rinehart and Winston.
- Kim, Daniel H. (1986). *Identifying Multi-Loop Feedback Structures Relevant to Delivery Delays and Customer Satisfaction in a Major Semiconductor Company..* Cambridge, MA: MIT Electrical Science and Engineering.
- Kim, Daniel H. (1989). Learning Laboratories: Designing Reflective Learning Environments In P. Milling & E. Zahn (Eds.), *Computer -Based Management of Complex Systems: Proceedings of the 1989 International Conference of the System Dynamics Society*. Berlin: Springer-Verlag.
- Kim, Daniel H. (1990). *Toward Learning Organizations: Integrating Total Quality Control and Systems Thinking*. Cambridge, MA: Pegasus Communications.
- Kim, Daniel H. (1993). Creating Learning Organizations: Understanding the Link between Individual and Organizational Learning. Forthcoming in *Sloan Management Review* (Fall).

- Kim, Daniel H., & Burchill, Gary (1992). Systems Archetypes as a Diagnostic Tool. In *Proceedings of the 1992 International System Dynamics Conference*. Utrecht, Netherlands: System Dynamics Society.
- Kofman, Fred (1992). Lecture Slides. In Cambridge, MA: MIT Sloan School of Management.
- Kolb, David A. (1984). *Experiential Learning: Experience as the Source of Learning and Development*. Englewood Cliffs, NJ: Prentice-Hall.
- Krubasik, Edward G. (1988). Customize Your Product Development. *Harvard Business Review*(Nov-Dec), 46-52.
- Kuhn, Thomas (1962). *The Structure of Scientific Revolutions*. Chicago: University of Chicago Press.
- Larkin, Jill H. (1983). The Role of Problem Representation in Physics. In D. Gentner & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Lawrence Erlbaum Associates.
- Larson, Erik W., & Gobeli, David H. (1988). Organizing for Product Development Projects. *Journal of Product Innovation Management*, 5, 180-190.
- Levitt, Barbara, & March, James G. (1988). Organizational Learning. *Annual Review of Sociology*, 14, 319-340.
- Lewin, Kurt. (1951). *Field Theory in Social Science*. New York: Harpers & Brothers.
- Lorsch, Jay W., & Lawrence, Paul R. (1965). Organizing for Product Innovation. *Harvard Business Review*(Jan-Feb), 109-122.
- Maidique, Modesto Al, & Zierger, Billie Jo (1985). The New Product Learning Cycle. *Research Policy*, 299-313.
- March, James G., & Olsen, Johan P. (1975). The Uncertainty of the Past: Organizational Learning Under Ambiguity. *European Journal of Political Research*, 3, 147-171.
- McDonough, Edward F., & Leifer, Richard P. (1986). Effective Control of New Product Projects: The Interaction of Organizational Culture and Project Leadership. *Journal of Product Innovation Management*, 3, 149-157.
- Meade, James E. (1973). *The Theory of Economic Externalities*. Geneva: A.W. Sijthoff-Leiden.
- Meyers, Patricia W., & Wilemon, David (1989). Learning in New Technology Development Teams. *Journal of Product Innovation Management*, 6, 79-88.
- Miles, Raymond E., & Randolph, W. A. (1980). Influence of Organizational Learning Styles on Early Development. In J. R. Kimberly & R. H. Miles (Eds.), *The Organizational Life Cycle*. San Francisco: Jossey-Bass.
- Miles, Raymond E., Snow, Charles C., Meyer, Alan D., & Coleman, Henry J., Jr. (1978). Organizational Strategy, Structure, and Process. *Academy of Management Review*, 3(3), 546-563.
- Miller, Danny , & Friesen, Peter H. (1980). Momentum and revolution in organizational adaptation. *Academy of Management Review*, 3, 546-562.
- Miller, George A. (1956). The Magic Number Seven, Plus or Minus Two: Some Limits on Our Capacity for Processing Information. *The Psychology Review*, 63(2), 81-97.
- Miller, George A., Galanter, Eugene, & Pribram, Karl H. (1960). *Plans and the Structure of Behavior*. New York: Henry Holt.
- Minsky, Marvin (1985). *The Society of Mind*. New York: Simon and Schuster.
- Mizuno, Shigeru (1988). *Management for Quality Improvement: The 7 New QC Tools*. Cambridge, MA: Productivity Press.
- Nelson, R., & Winter, S. (1982). Organizational Capabilities and Behavior. In R. Nelson & S. Winter (Eds.), *An Evolutionary Theory of Economic Change*. Cambridge, MA: Harvard University Press.
- Norman, Donald A. (1983). Some Observations on Mental Models. In D. Gentner & A. Stevens (Eds.), *Mental Models* Hillsdale, NJ: Lawrence Erlbaum Associates.
- Nuttin, Joseph R. (1976). Motivation and Reward in Human Learning: a Cognitive Approach. In W. K. Estes (Eds.), *Handbook of Learning and Cognitive Processes*. Hillsdale, NJ: Erlbaum.
- Ohmae, Kenichi (1990). *The Borderless World: Power and Strategy in the Interlinked Economy*. New York: HarperCollins.
- Ostrom, Elinor (1977). Collective Action and the Tragedy of the Commons. In G. Hardin & J. Baden (Eds.), *Managing the Commons* (pp. 179-181). New York: W. H. Freeman & Company.

- Ozawa, Masayoshi (1988). *Total Quality Control and Management*. Tokyo: JUSE Press.
- Pennington, N. , & Hastie, R. (1991). A Cognitive Theory of Juror Decision Making: The Story Model. *Cardozo Law Review*, 13, 519-557.
- Perrow, Charles (1986). *Complex Organizations: A Critical Essay*. New York: Random House.
- Pessemier, Edgar A. (1986). *Product Management: Strategy and Organization* (2nd ed.). Malabar, FL: Robert E. Krieger Publishing.
- Peters, Thomas J. (1991). Beyond Speed: We're All in the Fad & Fashion Business. *Industry Week*(June 3), 20-25.
- Peterson, John T., & Tok How, Lim (1993) *A System Dynamics Analysis of Total Quality Management Implementation False Starts*. Unpublished Master's Thesis, MIT Sloan School of Management.
- Piaget, Jean (1970). *Structuralism*. New York: Basic Books.
- Postman, Leo (1976). Methodology of human learning. In W. K. Estes (Eds.), *Handbook of Learning and Cognitive Processes*, 3. New York: John Wiley and Sons.
- Powers, William T. (1973). *Behavior: the Control of Perception*. Chicago: Aldine Publishing.
- Randers, Jorgen (Ed.). (1980). *Elements of the System Dynamics Method*. Cambridge, MA: MIT Press.
- Reinhart, C. (1985). The Competency Trap. *Training*, 22(May), 160.
- Restak, Richard M. (1988). *The Mind*. New York: Bantam Books.
- Richardson, George P. (1981). *Problems with Causal-Loop Diagrams*. Cambridge, MA: System Dynamics Group, MIT Sloan School of Management.
- Richardson, George P. (1986). Dominant Structure. *System Dynamics Review*, 2(1, Winter), 68-75.
- Richardson, George P., & Pugh, Alexander L. (1981). *Introduction to System Dynamics Modeling with DYNAMO*. Cambridge, MA: MIT Press.
- Richardson, George , & Senge, Peter (1989). Corporate and Statewide Perspectives on the Liability Insurance Crisis. In P. Milling & E. Zahn (Eds.), *Computer-Based Management of Complex Systems: Proceedings of the 1989 International Conference of the System Dynamics Society*. Berlin: Springer-Verlag.
- Richardson, George, Andersen, David F., Rohrbaugh, John, & Steinhurst, William (1992). Group Model Building. In *Proceedings of the 1992 International System Dynamics Conference*. Utrecht, The Netherlands: System Dynamics Society.
- Richmond, Barry (1990). Systems Thinking: A critical set of Critical Thinking Skills for the 90's and beyond. In D. F. Andersen, G. P. Richardson, & J. D. Sterman (Ed.), *System Dynamics 1990*, 2 (pp. 934-947). Chestnut Hill, MA: System Dynamics Society.
- Richmond, Barry, Vescuso, Peter, & Peterson, Steve (1987). *An Academic User's Guide to STELLA*. Lyme, NH: High Performance Systems.
- Roberts, Nancy (1983). Testing the World with Simulations. *Classroom Computer News*, 3(3).
- Roberts, Roger (1992). *Implementing the Learning Organization: The Application of Systems Thinking Tools to Product Development in a Manufacturing Firm*. Cambridge, MA: MIT Sloan School of Management.
- Roseneau, Milton D. (1990). *Faster New Product Development: Getting the Right Product to Market Quickly*. New York: AMACOM.
- Rosenthal, Stephen R. (1992). *Effective Product Design and Development: How to Cut Lead Time and Increase Customer Satisfaction*. Homewood, IL: Irwin.
- Roth, George (1992). *Organizational-Level Learning as Culture and Routines: A Model for Empirical and Theoretical Development*. Unpublished manuscript, MIT Sloan School of Management.
- Roussel, Philip A., Saad, Kamal N., & Erickson, Tamara J. (1991). *Third Generation Research & Development*. Boston: Harvard Business School Press.
- Rubenstein, A.H., Chakrabarti, A.K., O'Keefe, R.D., Souder, W.E., & Young, H.C. (1976). Factors Influencing Innovation Success at the Project Level. *Research Management*(May).
- Schank, R.D. & Abelson, R. (1977). *Scripts, Plans, Goals, and Understanding*. Hillsdale, NJ: Erlbaum.
- Scheibla, S.H. (1989). So Why Ain't It Dead? *Gramm, Rudman & Hollings on the deficit dragon*. *Barrons*, 69(July 10), 6-7.
- Schein, Edgar (1987a). *The Clinical Perspective in Fieldwork*. Newbury Park, CA: Sage Publications.
- Schein, Edgar H. (1987b). *Organizational Culture and Leadership*. San Francisco: Jossey-Bass.

- Schein, Edgar H. (1987c). *Process Consultation Volume II: Lessons for Managers and Consultants*. Reading, MA: Addison-Wesley.
- Schein, Edgar H. (1988). *Process Consultation Volume I: Its Role in Organization Development*. Reading, MA: Addison-Wesley.
- Schein, Edgar (1993). How Can Organizations Learn Faster? The Challenge of Entering the Green Room. *Sloan Management Review*, Winter, 85-92.
- Schneiderman, Arthur M. (1988). Setting Quality Goals. *Quality Progress*, April, 51-57.
- Schön, Donald (1971). *Beyond the Stable State*. London: Temple Smith.
- Senge, Peter M. (1989). *Organizational Learning: New Challenges for System Dynamics*. Cambridge, MA: MIT Sloan School of Management.
- Senge, Peter M. (1990a). The Leader's New Work. *Sloan Management Review*, 12(1), 7-23.
- Senge, Peter M. (1990b). *The Fifth Discipline*. New York: Doubleday.
- Senge, Peter M., & Lannon, Colleen (1990). Managerial Microworlds. *Technology Review*(July).
- Shiba, Shoji (1992). *Four Years of Management Thinking: Planning & Implementation*. Cambridge, MA: Productivity Press.
- Shiba, Shoji, & Graham, Alan (1993). Total Quality Management as a Learning System. *Center for Quality Management Journal*, 2(1), 48-56.
- Shiba, Shoji, Graham, Alan, and Walden, David (1993). *A New American TQM: Four Practical Revolutions in Management*. Cambridge, MA: Productivity Press.
- Simon, Herbert A. (1957). *Models of Man*. New York: Wiley.
- Simon, Herbert A. (1981). *Sciences of the Artificial*. Cambridge, MA: MIT Press.
- Simon, Herbert A. (1991). Bounded Rationality and Organizational Learning. *Organization Science*, 2(1), 125-134.
- Simon, Herbert A., & Ando, Albert (1961). Aggregation of Variables in Dynamic Systems. *Econometrica*, 29(2), 111-138.
- Sims, Henry P., Gioia, Dennis A., & Associates, & (1986). *The Thinking Organization*. San Francisco: Jossey-Bass.
- Smith, Preston G., & Reinertsen, Donald G. (1991). *Developing Products in Half the Time*. New York: Van Nostrand Reinhold.
- Stalk, George, & Hout, Thomas M. (1990). *Competing Against Time: How Time-Based Competition is Reshaping Global Markets*. New York: The Free Press.
- Stalk, George, Evans, Philip, & Shulman, Lawrence E. (1992). Competing of Capabilities: The New Rules of Corporate Strategy. *Harvard Business Review*(March-April), 57-69.
- Starbuck, William H. (1983). Organizations as Action Generators. *American Sociological Review*, 48, 91-102.
- Stata, Ray (1989). Organizational Learning—The Key to Management Innovation. *Sloan Management Review*, 12(1), 63-74.
- Sterman, John D. (1988). Strategy Dynamics: The Rise and Fall of People Express. In Cambridge, MA: MIT Sloan School of Management.
- Sterman, John D. (1989). Modeling Managerial Behavior: Misperceptions of Feedback in a Dynamic Decision Making Experiment. *Management Science*, 35(3), 321-339.
- Sterman, John, & Senge, Peter M. (1991). Systems Thinking and Organizational Learning: Acting Locally and Thinking Globally in the Organization of the Future. In K. Thomas & M. Useem (Eds.), *Transforming Organizations* Oxford, England: Oxford University Press.
- Stone, Eugene F. (1978). *Research Methods in Organizational Behavior*. Glenview, IL: Scott, Foresman and Company.
- Strauss, Anselm L. (1987). *Qualitative Analysis for Social Scientist*. Cambridge: Cambridge University Press.
- Takeuchi, Hirotaka, & Nonaka, Ikujiro (1986). The New New Product Development Game. *Harvard Business Review*(Jan-Feb), 137-146.
- The Economist* (April 18, 1992). The Cracks in Quality.
- Thornton, P.B. (1992). Teamwork: Focus, frame, facilitate. *Managerial Review*, 81(November), 46-7.
- Tushman, Michael L. , & Anderson, Philip (1986). Technological discontinuities and organizational environments. *Administrative Science Quarterly*, 31, 439-465.

- Tversky, A. , & Kahneman, D. (1974). Judgment under Uncertainty: Heuristics and biases. *Science*, 1985, 1124-1131.
- Tversky, A. , & Kahneman, D. (1987). Rational choice and the framing of decisions. In R. M. Hogarth & M. W. Reder (Eds.), *Rational Choice: The contrast between economics and psychology*. Chicago: University of Chicago Press.
- Ulrich, Karl , Sartorius, David, Pearson, Scott, & Jadiela, Mark (1993). Including Value of Time in Design for Manufacturing Decision Making. *Management Science*, 39(4), April.
- Urban, Glen L., & Hauser, John R. (1980). *Design and Marketing of New Products*. Englewood Cliffs, NJ: Prentice-Hall.
- Vennix, Jac A. M. (1990). *Mental Models and Computer Models: Design and Evaluation of a Computer-based Learning Environment for Policy-making*. Beuningen, Netherlands: University of Nijmegen.
- Vennix, Jac A.M., Gubbels, Jan W., Post, Doeke, & Poppen, Henk J. (1990). A Structured Approach to Knowledge Elicitation in Conceptual Model Building. *System Dynamics Review*, 6(2, Summer), 194-208.
- Vygotsky, Lev (1989). *Thought and Language*. Cambridge, MA: MIT Press.
- Walton, R.E. , & Hackman, J.R. (1986). Groups under contrasting management strategies. In P. S. Goodman (Eds.), *Designing Effective Work Groups*. San Francisco: Jossey-Bass.
- Watson, James (1981). *The Double Helix: A Personal Account of the Discovery of the structure of DNA*. London: Weidenfeld & Nicolson.
- Weick, Karl E. (1991). The Nontraditional Quality of Organizational Learning. *Organization Science*, 2 (1), 116-124.
- Williams, Michael D., Hollan, James D., & Stevens, Albert L. (1983). Human Reasoning About a Simple Physical System. In D. Gentner & A. Stevens (Eds.), *Mental Models*. Hillsdale, NJ: Erlbaum Associates.
- Winston, Patrick (1984). *Artificial Intelligence*. Reading, MA: Addison-Wesley.
- Winter, S.G. (1985). The case for 'mechanistic' decision making. In H. Pennings (Eds.), *Organizational Strategy and Change*. (pp. 99-113). San Francisco: Jossey-Bass.
- Womack, James P., Jones, Daniel T., & Roos, Daniel (1990). *The Machine that Changed the World*. New York: Rawson Associates.

Appendix A
Ten Systems Thinking Tools

Tool		Description
I	<p>"hard" variables</p> <p>1. Double-Q Diagram</p> <p>"soft" variables</p>	A brainstorming tool for capturing free-flowing thoughts in a structured manner and distinguishing between hard and soft variables that affect the issue of interest.
II	<p>2. Behavior Over Time Diagram</p>	Using some of the main branch variables from the fishbone diagram, the behavior of each one can be graphed over time, taking into account any inter-relatedness in their behavior. (Also called reference modes).
	<p>3. Causal Loop Diagram</p>	Drawing out causal relationships using the fishbone and behavior over time diagrams helps identify reinforcing and balancing processes.
	<p>4. System Archetypes</p>	Helps in recognizing common system structures that fit one of the recurring system archetypes such as eroding goals, shifting the burden, limits to growth (compensating feedback), fixes that fail (policy resistance), etc.
III	<p>5. Graphical Function Diagram</p>	Represents the effect of one variable on another graphically by plotting the relationship over the entire range of values that the X variable may theoretically operate.
	<p>6. Structure-Behavior Pairs</p>	A library of simple structure-behavior pairs consist of the basic dynamic structures that can serve as building blocks for developing computer models, e.g. exponential growth, delays, smooths, S-shaped growth, oscillations, etc.
	<p>7. Policy Structure Diagram</p>	A conceptual map of the decision making process that is embedded in the organization. Focuses on the factors which are weighed for each decision point. Build library of generic structures.
IV	<p>8. Computer Model</p>	Allows you to map all the relationships that have been identified as relevant and important to an issue in terms of mathematical equations and run policy analyses through multiple simulations.
	<p>9. Management Flight Simulator</p>	Provides "flight" training for managers through the use of interactive computer games based on a computer model. Through formulating strategies and making decisions to achieve them, help connect consequences to decisions made
	<p>10. Learning Laboratory</p>	A managers' practice-field. It is equivalent to a sports team's experience, where active experimentation is blended with reflection and discussion. Uses all the systems thinking tools, from fishbone diagrams to MFS's.

Ten Systems Thinking Tools
 (source: *The Systems Thinker*TM, Vol. 1, No. 3)

Figure 1

Appendix B
Guidelines for Causal Loop Diagrams

	Guideline	Example
Selecting Variable Names	1 When choosing a variable name, use nouns. Avoid verbs and action phrases since the action is conveyed in the arrows. For example, "Costs" is better than "Increasing Costs," since a decrease in Increasing Costs is confusing. The sign of the arrow ("s" for same or "o" for opposite) indicates whether Costs increase or decrease relative to the other variable.	
	2 Variables should be something that can be measured—quantities that can vary over time. It does not make sense to say that "State of Mind" increases or decreases. A term like "Happiness," on the other hand, can vary.	
	3 Choosing the "positive" sense of a variable name is preferable. An increase or decrease in "Growth" is clearer than an increase or decrease in "Contraction."	
Loop Construction	4 For every course of action included in the diagram, think of the possible unintended consequences as well as the expected outcomes. An increase in "Production Pressure" may increase "Production Output," for example, but it may also increase "Stress" and decrease "Quality."	
	5 All balancing loops are goal-seeking processes. Try to make the goals driving the loop explicit. For example, Loop B1 may raise questions as to why increasing "Quality" would lead to a decrease in "Actions to Improve Quality." By explicitly identifying "Desired Quality" as the goal in Loop B2, we see that the "Gap in Quality" is really driving improvement actions.	
	6 Distinguishing between perceived and actual states "Perceived Quality" vs. "Actual Quality" is important. Perceptions often lag reality, and mistaking the perceived status for current reality can be misleading and create undesirable results.	
	7 If there are multiple consequences of a variable, start by lumping them into one term while finishing the rest of the loop. For example, "Coping Strategies" can represent many different ways we respond to stress (exercise, meditation, alcohol use, etc.).	
	8 There are almost always differing long-term and short-term consequences of actions. Draw loops with increasing radius as they progress from short- to long-term processes. Loop B1 shows the short-term behavior of using alcohol to combat stress. Loop R1, however, draws out the long-term consequences which will actually increase stress.	
General Tips	9 If a link between two terms is not clear to others and requires a lot of explaining, the variables probably need to be redefined or an intermediate term needs to be inserted. "Higher Demand" leading to lower "Quality" may be less obvious than when "Production Pressure" is inserted in between.	
	10 A short-cut to determining whether a loop is balancing (B) or reinforcing (R) is to count the number of "o's" in the loop. An odd number of "o's" indicates a balancing loop, an even number (or none) means it is a reinforcing loop. CAUTION: After labeling the loop, you should always talk yourself around the loop and make sure the story agrees with your R or B label.	

Guidelines for Drawing Causal Loop Diagrams

(source: *The Systems Thinker*™, Vol. 1, No. 3)

Figure 1

Appendix C
MIT Organizational Learning Center Pilot Projects

MIT ORGANIZATIONAL LEARNING CENTER PILOT PROJECTS

Approximately fifteen sponsor companies work collaboratively with MIT to develop a research agenda at the Organizational Learning Center (OLC). One of the ways in which the companies collaborate is through pilot projects. The pilot projects have two primary challenges: (1) to contribute significantly to participating corporations and (2) to generate research results in the form of new tools, methods, and understandings of the ways that learning is promoted or thwarted. These challenges are aligned with the "basic charter" of the Center, restated here:

"to draw closer together the communities of management research and management practice so as to make learning about complex, dynamic managerial issues a way of life in organizations."¹

The project definition process received a lot of consideration at the OLC, and the notes from a meeting on the topic emphasizes its importance:

"By a 'good project,' we agreed that we meant, 'projects that meet the standards of both a CEO and a rigorous researcher.' In other words, the project as conceived and articulated stands to help managers deal with crucial issues and produce usable knowledge that can advance management practice and theory more broadly. This is a demanding standard. Very few academic research projects would meet it, as would few organization change initiatives. But we feel it is realizable... because management's and academia's goals are fundamentally aligned."²

To facilitate a "good project," the OLC established an outline for project definition process, which will be explored below. The OLC also strongly recommended that the first-generation projects concern an area where a strong foundation of system dynamics and a management flight simulator (computer simulation game) exist. This project, as it was defined with Cars, Inc., fit the desired criteria.

¹Information from a Center memo from Peter Senge and Bill Isaacs to Corporate Affiliates about the project definition process, August 6, 1991, p. 1.

²Information from Center memo from Peter Senge, Dan Kim, David Kreutzer, Fred Kofman, Janet Gould-Kreutzer and Bill Isaacs to Corporate Affiliates about defining projects, July 4, 1991, pp. 3-4.

**Appendix D:
Product Development Management Flight Simulator
Facilitator's Guide**

Prepared by
Donald A. Seville & Daniel H. Kim
Organizational Learning Center
MIT Sloan School of Management
1 Amherst Street, E40-294
Cambridge, MA 02139
(617) 253-1587

Table of Contents

1. INTRODUCTION	364
1.1 The Product Development Management Flight Simulator	364
1.2 Major Components of Flight Simulators	365
2. The Microworld	366
2.1 Product Engineering	366
2.1.1 Task Sector	367
2.1.2 Staffing Sector	368
2.1.3 Schedule Sector	370
2.1.4 Quality, Work Week, and Rework	375
2.1.5 Coordination	380
2.2 Process Engineers	385
3. Simulator Controls	388
3.1 Simulator Cockpit	388
3.2 Management Decisions	389
3.3 Initial Decisions	390
4. Facilitating Simulator Sessions	392
4.1 Suggestions On Running A Session	392
4.1.1 Introduction To Management Flight Simulators	393
4.1.2 The Product Development MFS	395
4.1.3 Beginning The Simulation	398
4.1.4 The Simulation	398
4.1.5 Outcome & Debrief	399

1. INTRODUCTION

The Facilitator's Guide was prepared for the training of future facilitators of Product Development Management Flight Simulator Sessions. The purpose of this guide is to help the facilitators develop an understanding of the interconnections and assumptions that form simulator's underlying model. This

will prepare the facilitator to describe the underlying model to users during a session and show how the management decisions influence the product development process in the flight simulator. The second purpose of this guide is to show how a facilitator might run a session, including the session introduction, various simulation scenario possibilities, session debrief, and an example of a successful product development strategy. The guide will discuss: (1) the flight simulator in general, (2) the underlying model called the *microworld*, (3) the management decisions, and finally (4) an overview of how to facilitate a flight simulator session.

This guide assumes that all facilitators have previously run the simulator in a Learning Lab environment and are familiar with causal loop diagrams and stock and flow structures. This Guide is meant to be used together with the User's Guide, which contains more detailed information on the flight simulator interface.

1.1 The Product Development Management Flight Simulator

The Product Development Management Flight Simulator (MFS) gives you the opportunity to "pilot" a product development process. A typical challenge in managing a product development program is to meet three objectives: finishing in a specified **time** limit, under an allocated **budget**, and with a competitive **quality**. In the simulator, the project is separated into two categories of tasks that must be completed. There are 100 generic product engineers "tasks" and there are 100 generic process engineering "tasks". The product engineers (PE) do the product design and testing, while the process engineers (PcE) design and build the manufacturing process to turn the design into a product. When both engineering groups finish all their tasks, the product is released into the market and will be sold. The resulting sales depend heavily on whether the product was released on time with a competitive quality.

As the "pilot" of this simulator, you are expected to make decisions each week about:

- adding or removing engineers,
- the desired number of work hours in a week,
- the scheduled completion date for each team,
- the amount of time spent coordinating between the two teams,
- and the quality "goal" for the product.

Your decisions will influence the rate at which the engineers can accomplish tasks, the quality they will build into the project, and even the rate they will leave the project because of stress.

There is no winning or losing. The purpose of the flight simulator is to provide the opportunity to experiment with management decisions that are too costly and lengthy to experiment with in real time. The default target completion date is 40 weeks. Although no decisions need be made after the project is completed, the simulation will continue to run for 80 weeks to show the sales from releasing the product into the market.

1.2 Major Components of Flight Simulators

There are three major components of the management flight simulator, the *microworld*, the *information system*, and the *simulator controls*.

Microworld: The microworld is the system dynamics model that underlies the flight simulator. Here the interactions between the different variables are explicitly described. It has been tested and calibrated; but like any model, it is a simplification of reality and is based on a set of assumptions about the product development process. The Facilitators Guide is designed to familiarize the you with this model.

Information System: The product development progress is monitored through a complex information system. You have access to multiple reports that detail the current status of different facets of the project. Historical information can be obtained as either graphs or tables and show the behavior of the variables over time. The information system is described in depth in the User's Guide to the MFS.

Simulator Controls: The simulators controls are in the "cockpit" of the flight simulator. Here you can make decisions and control the information system by selecting the report or graph you would like to view.

The facilitator's guide is intended to describe the microworld and the simulator controls. The information system is described in the User's Guide, including a description of each report and graph available.

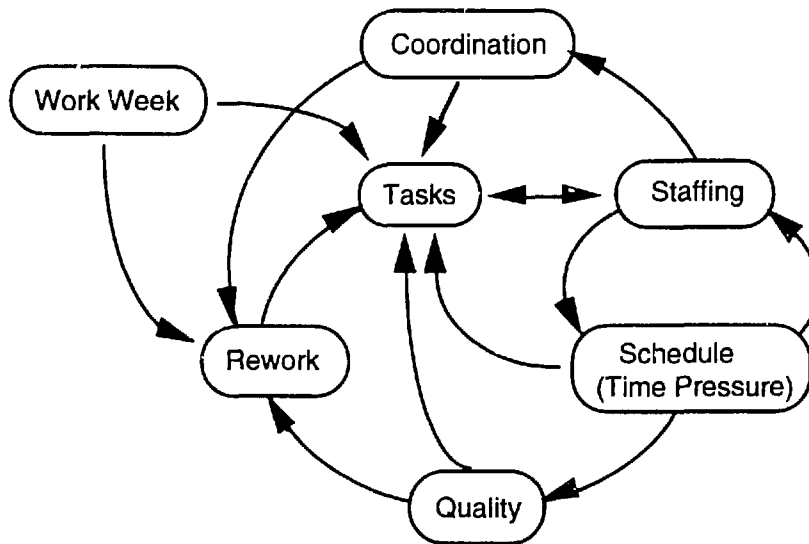
2. THE MICROWORLD

To explain the behavior of the product development management flight simulator during a session, it is necessary to understand the microworld behind the model. We begin our exploration of the microworld by examining the major stock and flow structures and critical feedback loops that govern the behavior of the system. There are two major parallel structures underlying the microworld; the product engineering (PE) structure and the process engineering (PcE) structure. These two structures are almost identical, so we will describe only one of them in depth. First the product development structure will be presented, and then the differences in the process engineering structure will be discussed. Using the causal loop diagrams developed, it should be possible to examine how management decisions can have both obvious and unexpected impacts on the product development "system" and understand how they influence the progress towards the project goals.

2.1 Product Engineering

The product engineering system has been separated into 7 sectors for discussion purposes. These sectors are Tasks, Staffing, Schedule, Quality, Rework, and Coordination. The general interactions of the seven sectors are shown in Fig. 1, the Microworld Overview Diagram. The objective of the project is to manage the timing, cost, and quality of completing all of the product and process engineering tasks. Thus, the heart of the product development microworld in the Tasks structure that governs the process of task completion. The available staff, the time left in the schedule, the length of the work week, the time spend coordinating, the quality of practice, and the amount of rework generated all influence the rate of which tasks are going to be completed each week. This overview diagram will be expanded into a much more detailed causal loop diagram for the product engineering "system" by discussing each sector and then adding it to the diagram. First, the basic structure that governs the product development tasks will be presented. To the Task Structure, the Staffing,

Schedule, Quality, Rework, Work Week, and Coordination Structures will be added one at a time until the full causal loop diagram is developed.

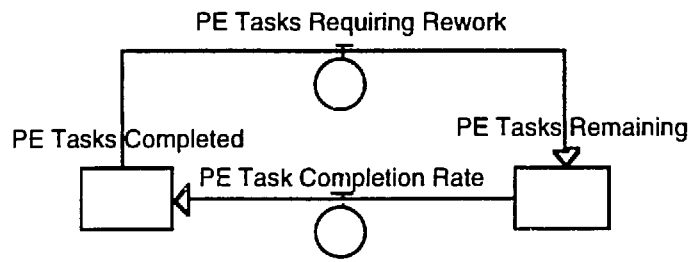


Microworld Overview Diagram

Figure 1

2.1.1 Task Sector

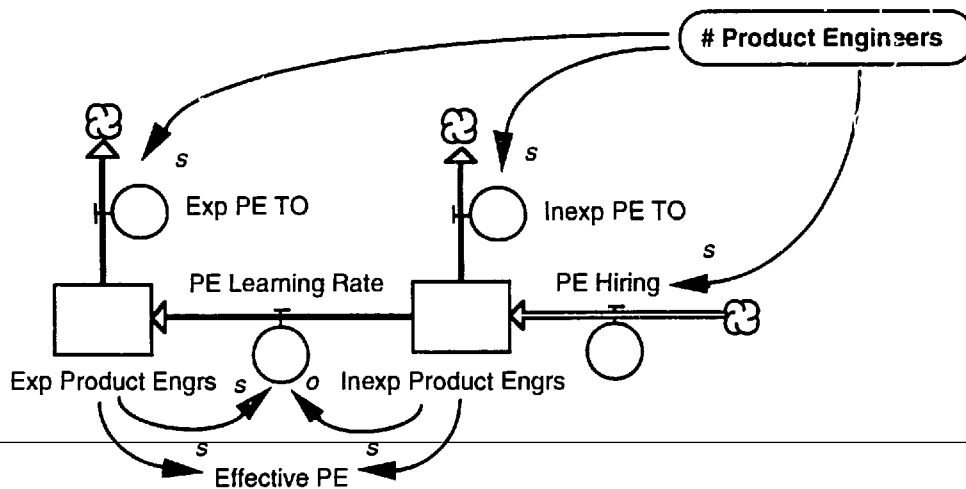
The model assumes that there are 100 generic “tasks” that must be completed by both the process and product engineers. We begin by examining the product engineering (PE) sector. The stock and flow diagram showing the heart of the task structure is shown in Fig. 2. The product engineers begin the project with 100 tasks remaining in the project. These are in the PE Tasks Remaining Stock. They are completed at a certain PE Task Completion Rate. The number of tasks completed flow into the PE Tasks Completed Stock. However, all tasks are not done correctly, nor are the mistakes found immediately. As the project progresses, some tasks are discovered to require rework, so are removed from the stock of PE Tasks Completed and are returned to the pool of PE Tasks Remaining. The rate at which tasks are completed, and the rate at which they are discovered to require rework is a function of many other variable in the product development process. This same stock and flow layout is used to show the current status of the product and process engineers in the Overview Report of the Management Flight Simulator.



PE Causal Looping Diagram, Task Structure
Figure 2

2.1.2 Staffing Sector

For tasks to be completed, there obviously must be engineers available to work on them. One of the major management decisions made in this flight simulator is to add or remove engineers. This project is assumed to be a one project of many in a large company so you are not hiring or firing engineers, but rather moving them internally from one project to another so there is no hiring delay. The stock and flow structure that captures the staffing dynamics for the product engineers is shown in Fig 3. This structure assumes that there are two different kinds of engineers that work at different levels of productivity. Experienced Product Engineers are assumed to work at a certain level of efficiency, measured by productivity (tasks per person per week). The project begins with several



Product Engineering Staffing Stock and Flow Structure
Figure 3

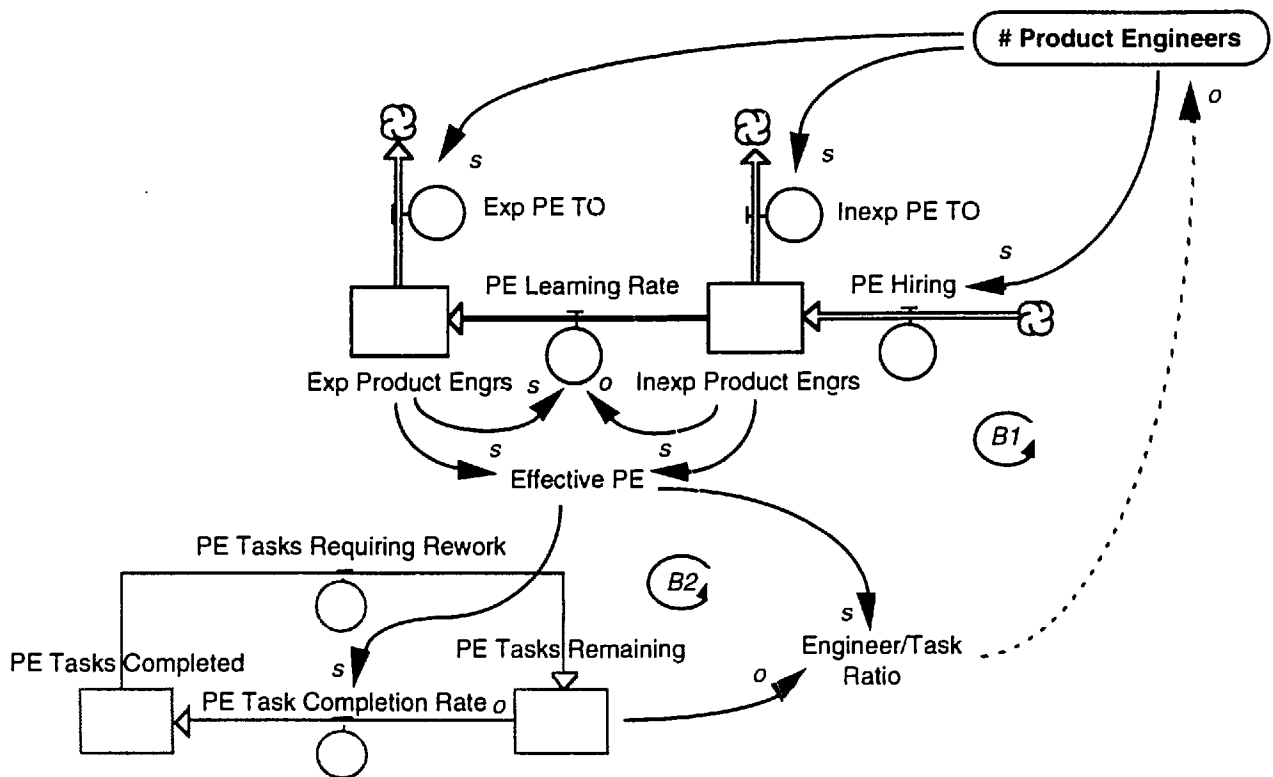
experienced product development engineers who are assumed to be the core team that worked on the earlier phases of this project. All other engineers added to the project through the **# Product Engineers** decision are assumed to be inexperienced to the project and must be trained before they are able to operate

at the full productivity of an experienced engineer. Inexperienced is defined as a fully trained engineer, but one unfamiliar to the project. Note that decision variables in the management flight simulator are displayed in bold and are contained by a rounded rectangle.

The rate at which inexperienced engineers become experienced is the PE Learning Rate. It is a function of the ratio between the experienced and inexperienced engineers. As shown by the causal links, the more experienced product engineers available to train the engineers relative to the number of inexperienced engineers, the greater the learning rate. Both inexperienced and experienced engineers leave the project through Project Engineering Turnover, which is modeled as flows out of the engineering stocks. There are both voluntary and involuntary turnovers. The involuntary turnovers are caused by a decision to remove engineers. A decision to remove engineers is made by entering a negative number in # **Product Engineers**. Inexperienced engineers will be removed first by an decision to remove engineers because they are less productive. Only after there are no more inexperienced engineers will experienced engineers be removed. The voluntary turnovers are a function of pressures on the engineers and will be discussed when other variables are added to the system. Because the experienced and inexperienced engineers have different productivity's, they are added together with a separate weighting to calculate the effective number of product development engineers. Although it is not shown in the Staffing diagram, maintaining an engineering team costs money. The total number of product engineers accumulate costs each week based on the Expense Per Engineer (an initial decision). The cumulative PE costs are called the PE Team Costs and are displayed in the Forecast Report of the MFS. This limits the number of engineers that can be hired while trying to finish the project under budget.

Figure 4 shows how the Staffing Sector interacts with the Tasks Sector. The previously discussed sectors and relations are shown in gray while the new variables and causal links are shown in black. When the effective number of engineers is increased by adding engineers, the PE Task Completion Rate will increase, decreasing the Tasks Remaining and increasing the Tasks Completed relative to what they would have been had the engineers remained constant. There is assumed to be an ideal Engineer/Task ratio that will allow the project to be finished in time and within budget. As more engineers are added, the ratio will approach its ideal value, which means less engineers need to be added. This

is a balancing loop and denoted *B1* in the diagram. Similarly, more effective engineers means a higher task completion rate, less tasks remaining, and so less engineers than would have been needed. This second balancing loop is denoted *B2*. The link between the Engineer/Task Ratio and the # **Product Engineers** is special because it is *not* a link in the model, but rather an implicit link that causes the “manager” to decide whether or not to add more engineers based on feedback (information) from the model. If there are too many engineers relative to the number of tasks remaining, then engineers can also be removed.



PE Causal Looping Diagram, adding Staffing Sector

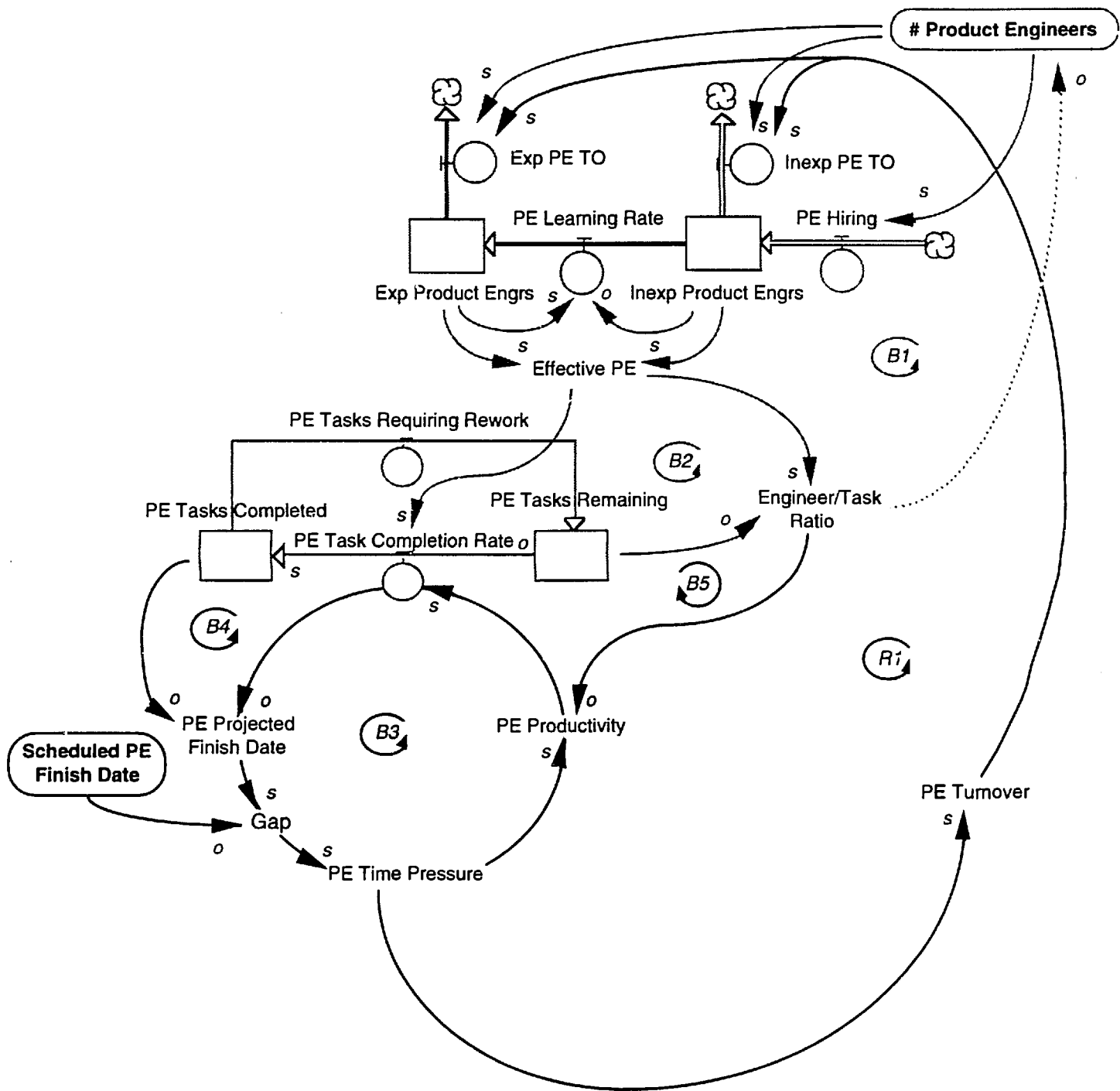
Figure 4

2.1.3 Schedule Sector

An important element in managing a product development project is to establish reasonable scheduled completion dates. There are two schedule dates that must be set in this project. A Project Release Date must be chosen for the project to be released into the market. This is also the deadline for the process engineers to finish all of their tasks since the process engineering is the last phase of the product development process. The Product Engineers have a separate deadline that is the Scheduled PE Finish Date. This enables the managers to set different

time pressures on the two groups on engineers. The key to managing the scheduled completion dates is to be ready to flexible enough to change the release dates to maintain the desired time pressure and thus quality of practice. The Schedule Sector for the product engineers in shown added to the causal loop diagram in Figure 5.

From the current # of tasks completed and the current task completion rate, a projected finish date is estimated. This is the week # that all the PE tasks will be completed by if the task completion rate remained constant. If there is a *gap* between the **Scheduled Finish Date** and the projected finish date, there is either excess time or too little time. Suppose the projected finish date was greater than the **Scheduled Finish Date**. Time pressure on the product engineers would increase since the engineers feel there is too little time in the schedule. High time pressure leads to a greater productivity, which increases the task completion rate, narrowing the gap between the projected and **Scheduled Finish Date**. This is a balancing feedback loop and is denoted *B3*. Similarly, if the task completion rate increases, the tasks completed would be greater than they otherwise would be, decreasing the projected finish date. This is also a balancing loop, *B4*. The exact way in which time pressure influences productivity is clearly an assumption. To show the assumption, the graphical function of the effect of time pressure on productivity used in the *ithink* model is shown in Fig. 6. It is the graphical pattern that demonstrates the assumed relationship rather than the exact numbers.



PE Causal Looping Diagram, adding Schedule Sector

Figure 5

In Fig. 6, the x-axis is the time pressure while y-axis is the effect of time pressure on productivity (called PE Eff SchedPress). When the time pressure is one, indicating that there is exactly enough time in the schedule to finish

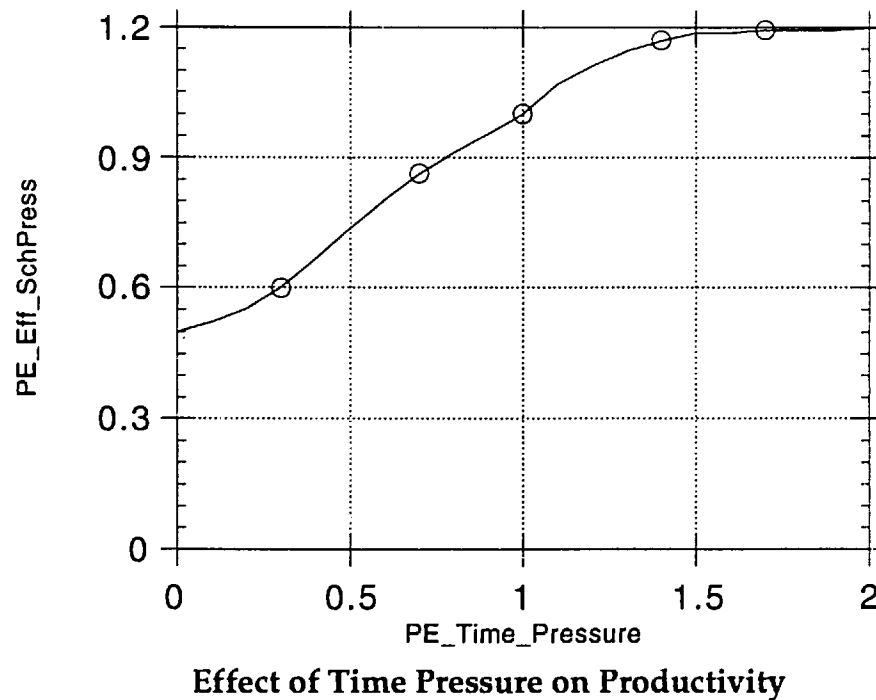
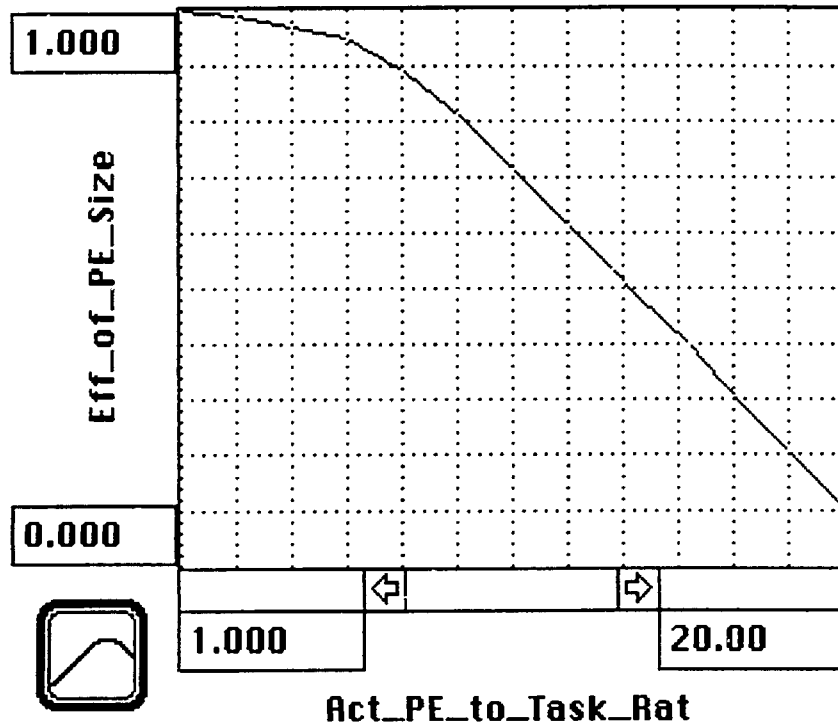


Figure 6

the project, the effect is one, which is a neutral effect and does not alter the productivity. When time pressure exceeds one, the effect can increase to 1.2, equivalent to a 20% increase in productivity. The most productivity can increase due to just time pressure is 20% because the quality of the work is assumed to stay constant. On the other hand, if the time pressure goes below 1, the productivity can drop by as much as 50%. This is because the engineers have the time in the schedule to keep engineering (trying to increase product quality), and are assumed to do so. Any time pressure beyond the limits in the graphical function will simply be the effect of the time pressure at the limit.

Returning to the causal loop diagram of Fig. 5, an increase in time pressure will also increase the turnover rate for engineers (meaning they leave for a less stressful work environment). They will turnover from both the inexperienced and experienced engineers. This is a reinforcing loop (*R1*): an increase in time pressure will increase turnover, which decreases the effective number of engineers, decreasing the task completion rate, increasing the gap between the

projected and **Scheduled Finish Date**, increasing time pressure. Another assumption is that the ratio of engineers/tasks remaining will influence productivity: too many engineers on too few tasks will become increasingly unproductive. This relationship is also a graphical function (see Fig. 7).



Effect of Engineer/Task Ratio on Productivity

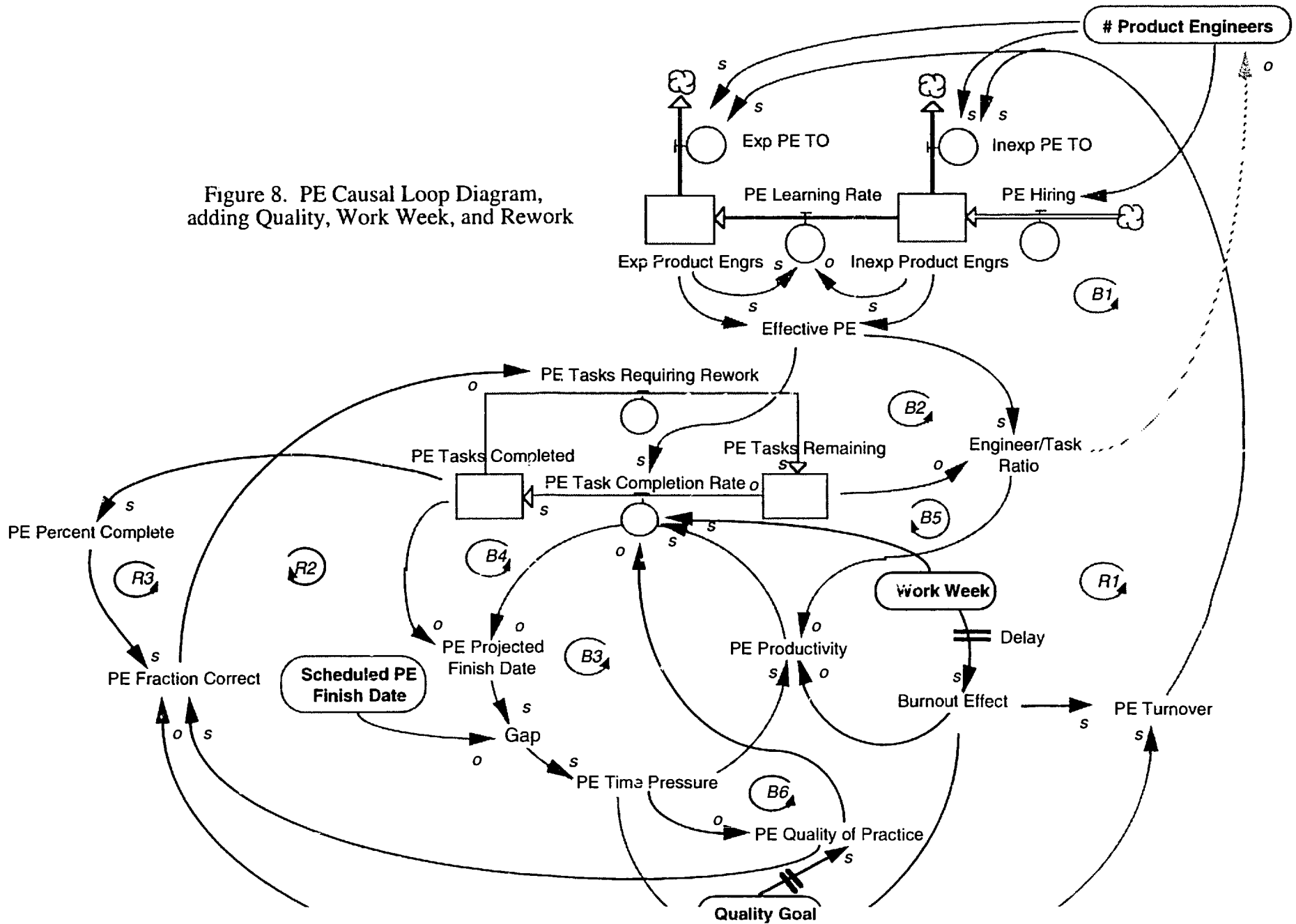
Figure 7

The limits on the axis of engineer to task ratio are complicated because they are a function of the Scenario Scale. Scenario Scale is an initial decision (described in the Simulator Controls Section) that sets the magnitude of the number of engineers required for this project. The graphical function in Fig. 7 is shown with a Scenario Scale of 1. For any scenario scale of less than one, multiply the upper limit of the PE to Task Ratio by the scenario scale. This assumption indicates that as the engineers/task ratio goes above one, they work with decreasing productivity (tasks per engineer per week). This creates a balancing loop (B5). An increase in the engineer/task ratio above 1 will decrease productivity, decreasing task completion rate, increasing the number of tasks remaining *relative to the number that would have remained if there were no change in productivity*. This leads to a decrease in the engineer/task ratio relative to what it would have been.

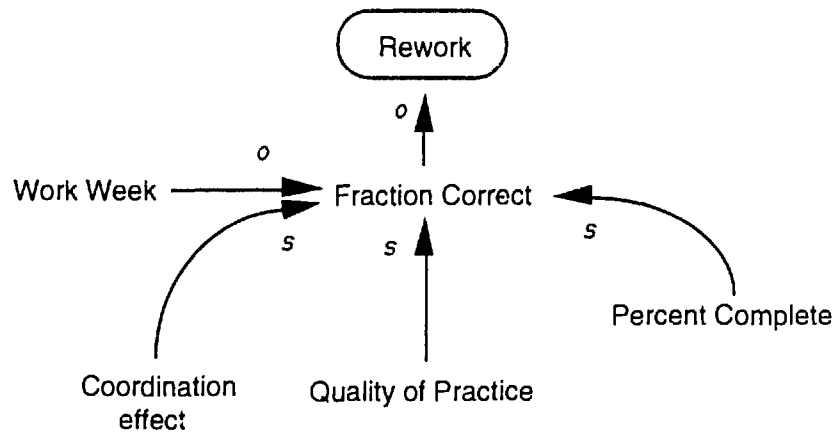
2.1.4 *Quality, Work Week, and Rework*

Quality. One of the three objectives in the flight simulator is to complete the project with a competitive quality. This quality is the quality of the product and is the final accumulation of all the tasks weighted by the quality of practice at the time each task was completed. The quality of practice is a measure of the amount of time and care put into completing each task. The PE quality of practice incorporated into the PE causal loop diagram is shown in Fig. 8. Note that this figure shows the additions of Quality, Work Week, and Rework. The quality of practice is a function of time pressure and the **Quality Goal**. An increase in time pressure is assumed to decrease the quality of practice as engineers are rushing to complete each task, which increases the task completion rate since less time is being spent on each task. As the task completion rate increases, time pressure will decrease, which will allow the quality of practice to increase. This is a balancing loop (*B6*). The **Quality Goal** will influence the quality of practice in the direction of the goal because this is the quality level the engineers are striving to work at. However, there is a time delay which represents the time delay between announcing a new goal at management level and actually seeing the goal implemented in engineering practice. Although there is only one **Quality Goal**, the product and process engineers have separate quality of practice because they have different time pressures influencing their portions of the project.

Figure 8. PE Causal Loop Diagram, adding Quality, Work Week, and Rework



Rework.. Rework is a measure of tasks that are completed incorrectly for a variety of reasons and must be reworked. As discussed in the Task Sector, when a task is discovered to require rework, it is removed from the tasks completed stock and returned to the tasks remaining stock. In this sector, we will examine what variables in the product development system influence the fraction of the tasks done correctly. Figure 9 shows an outline of the causes of rework. Each week, a certain percentage of the tasks completed are



Causes of Rework

Figure 9

completed correctly. Those tasks not completed correctly require rework. Obviously, the more tasks correct, the less rework, so there is a causal connection in the opposite direction. The fraction of tasks completed each week is influenced by the work week, the coordination effect, the current quality of practice, and the percent of the total product engineering tasks complete.

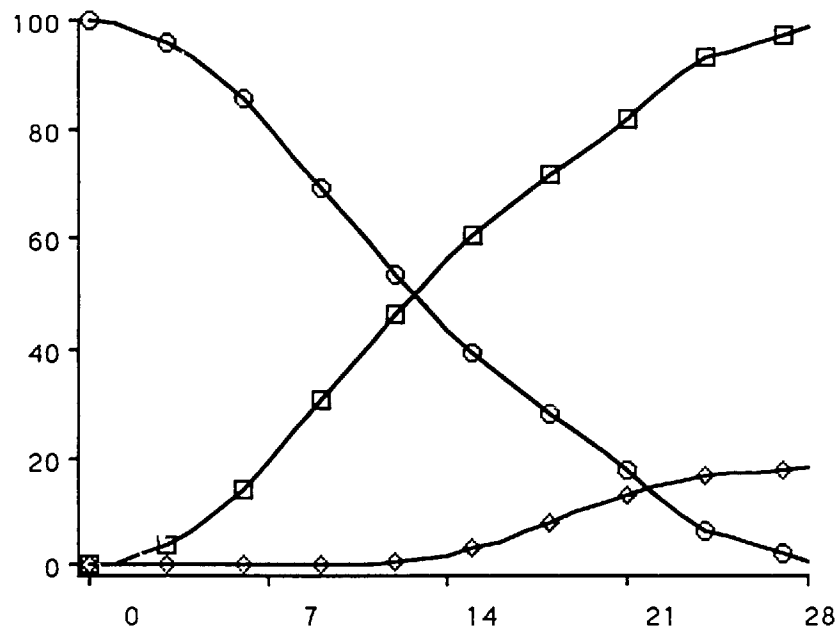
If we return to the causal loop diagram shown in Figure 8, we can tie each of the causes of rework into the system. First, there is assumed to be some inherent error associated with the project that is a function of the percent of the total PE tasks complete. In the beginning of the project, the most unknowns exists and there is the highest probability of error. As the project progresses, the fraction of tasks done correctly will increase, approaching 1 when all the tasks are complete. This feedback loop is a reinforcing loop (R3), the more tasks complete, the greater the fraction of the total PE tasks complete, the greater fraction correct, leading to less tasks requiring rework, and a higher number of tasks complete than there would otherwise have been. The fraction of the tasks completed correctly is also influenced by several characteristics of the working environment. Clearly, the quality of practice influences the fraction of tasks completed correctly. This

creates a reinforcing loop (*R2*), as an increase in quality of practice will lead to an increase in the fraction correct, decreasing the rework required, increasing the tasks completed, decreasing the gap between projected and scheduled finish date, decreasing time pressure, increasing the quality of practice. Work week influences the rework because prolonged period of a long work week can cause burnout, which increases the probability of errors being made. Finally, the coordination effect can increase the fraction correct because increased coordination means more time is being spend making sure the part is appropriate for its intended usage. The effect of coordination is not shown in Figure 8 because coordination has not been introduced into the system yet. Coordination is discussed in the next section.

Another important assumption is that mistakes are not discovered immediately. To simulate this phenomena, the rework will stay in the stock of tasks completed as *undiscovered* rework. Only when the rework is “discovered” does it flow from the stock of tasks completed into the stock of tasks remaining. This means the tasks requiring rework will appear “completed” until the mistakes are discovered leading to over optimistic projections of completion. The rate at which the tasks requiring rework are discovered is a function of the amount of the project completed and the coordination.

As the project progresses, and different designs are integrated into a single product, errors become obvious. This is why the tasks requiring rework seem to all be found as the project nears completion even though rework is generated throughout the project. Increased coordination also increases the rework discovery rate as the product engineers and process engineers can review each other’s work. Figure 10 shows a “base case” of the discovered product engineering rework relative to the total number of tasks completed, taken from a run on the management flight simulator. The line denoted with the diamonds is the total number of tasks discovered to require rework. Although the tasks found to require rework are returned the stock of tasks remaining, they are added to a stock called *total discovered rework* to keep track of the total number of tasks that required rework. As is clearly shown, the tasks are discovered to require rework as the project nears completion, slowing down the project progress substantially. The actual rework discovery rate (tasks found to require rework each week) for the same time period is shown in Figure 11. This clearly shows when the rework was discovered during the project.

- PE_Tasks_Completed
- PE_Tasks_Remaining
- ◇ Total_Disc_PE_Rework



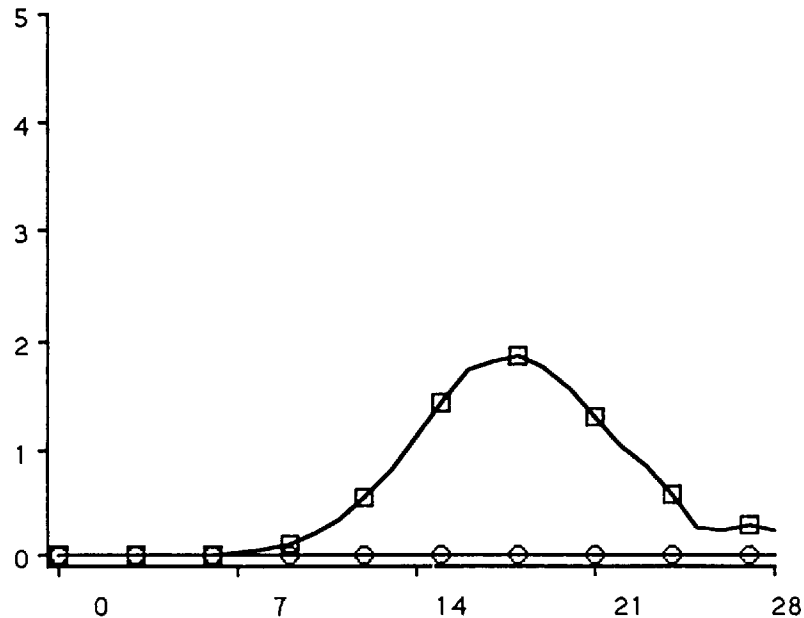
"Base case" product engineering rework

Figure 10

Work Week. Since the task completion rate is a function of the number of engineers, their current productivity, and the amount of time spent working each week, an easy way to increase the task completion rate is to increase the Work Week. There is a separate work week decision for each engineering team, the **PE Work Week** and the **PcE Work Week**. As shown in Fig. 8, the immediate effect of the work week is to increase the task completion rate. If the monthly average work week becomes too long, the engineers will begin to burnout. An increase in the burnout effect will decrease productivity, increase engineering turnover, and decrease the fraction correct of the tasks completed. These causal links are not shown as full feedback loops because **PE Work Week** is a decision in the game rather than being an indigenous relationship in the microworld. The feedback loops are connected outside the microworld when the decision to change work week is made in the game based on information from the microworld. There can be overtime costs associated with the work week. A cost for overtime is calculated based the overtime rate (an initial decision) and the engineering salary (another initial decision) for the number of engineering hours

worked over the base week of 40 hours. If overtime costs are not desired, the overtime rate should be set to zero at the beginning of a simulation session.

□ PE_Rework_Discovery
○ PcE_Rework_Discovery



“Base case” product engineering rework discovery rate

Figure 11

2.1.5 Coordination

The final component to the PE causal loop diagram is coordination. Coordination between product and process engineers will allow them to exchange useful information and increase cooperation such that they can improve their productivity and avoid making errors, hence reducing the rework necessary. Since the process engineers are designing a manufacturing process to build the product that was designed by the development engineers, coordination is clearly a crucial element to an efficient project.

“Coordination” in this model tries to capture some of the differences in product development performance between a “lean production” organization³

³ “Lean Production” concept as defined in The Machine That Changed The World: The Story of Lean Production by James Womack, Daniel Jones, and Daniel Roos. New York: HarperCollins, 1990.

and the traditional “mass production” organization. According to Clark and Fujimoto⁴, the average Japanese manufacturer (traditionally associated with lean production) required about 1.7 million engineering hours for a new automotive product development project and took about 46 months from first design to customer deliveries. In contrast, the average American and European producers (traditionally associated with mass production) required about 3.1 million engineering hours and took 60 months. Obviously, not all of this difference in performance occurs during the process and product engineers phases. The average U.S. product and process engineering phases required some 34 months while the average Japanese producer complete these phases in about 24 months. Clark and Fujimoto found that the average Japanese had higher productivity, more overlap between product and process engineering, and less required rework. In The Machine That Changed The World, the difference in product development performance between lean production and mass production is attributed mostly to four project characteristics; leadership, teamwork, communication, and simultaneous development.⁵ Each of these project characteristics is an important aspect of project management. In the microworld, we try to capture the concepts of leadership, teamwork, and communication under the broad heading of coordination for simplicity. Coordination makes simultaneous development possible because the process engineers can begin designing the manufacturing process based on informal communication instead of having to wait for the finished product designs. Obviously, better communication (more coordination) leads to less mistakes during simultaneous development.

Returning to the microworld we have been discussing, the PE causal loop diagram with the effects of coordination added is shown in Fig. 12. Coordination is based on the **Coordination Fraction** decision. The **Coordination Fraction** is the fraction of the work week to be spent on coordination activities. Because the number of engineers in the product team and the process team can be different at any given time, the coordination is based on the smaller of the two team sizes. For example, if the **Coordination Fraction** was .1 (10%) and there were 10 total product engineers and 20 total process engineers, the coordination effect would

⁴ Kim B. Clark and Takahiro Fujimoto, Product Development Performance: Strategy, Organization, and Management in the World Auto Industry, Boston, MA: Harvard Business School Press, 1991, p. 78.

⁵ The Machine That Changed The World: The Story of Lean Production, p. 117

be based on 10 engineers from each team contributing 10% of their productive time towards coordination. This is equivalent to 1 engineer from each team spending 100% of time coordinating. The equivalent number of engineers spending 100% time coordinating are called the coordinating engineers. The immediate effect of coordination is that the coordinating engineers are coordinating rather than completing tasks, so they are subtracted from the effective engineers, reducing the task completion rate.

Coordination involves changing work habits and learning new communication skills, so there is actually a time delay between the investing time in coordination and feeling the positive effects of the coordination. The delayed coordination is called the coordination effect. The coordination effect is a function of the ratio of coordinating engineers to the total number of engineers in the entire project. The more coordination engineers there are relative to the total number of engineers, the greater the coordination effect will be. The coordination effect will range from zero to 1; 0 indicates there is no coordination effect, .5 indicates that overall, 50% of the engineering time is effectively spent coordinating, and 1 indicates that all engineers in the project spend 100% of their time effectively coordinating.

Based on the research previously discussed, the coordination effect is assumed to influence the fraction of work done correctly, the productivity of the engineers, and although not shown in the causal loop diagram, the rework discovery rate as discussed in the section on Rework. The effect of the coordination effect on the *product engineering* fraction of tasks completed correctly is displayed as a graphical relationship in Fig. 13. When the coordination effect is zero, indicating no effective coordination, the effect on fraction correct is 1. Remember that an output of 1 is neutral, no influence on the fraction correct. The fraction correct increases as the coordination effect increases, but with diminishing returns. The maximum gain in the product development fraction correct from coordination is about 10%. The gain from coordination is relatively low for the product engineers because product engineering is an upstream activity relative to the process engineering. This means that the process engineers base their design on the work of the product engineers. Because of this, coordination has a different influence on the process engineering fraction correct as will be discussed in the section on Process Engineering.

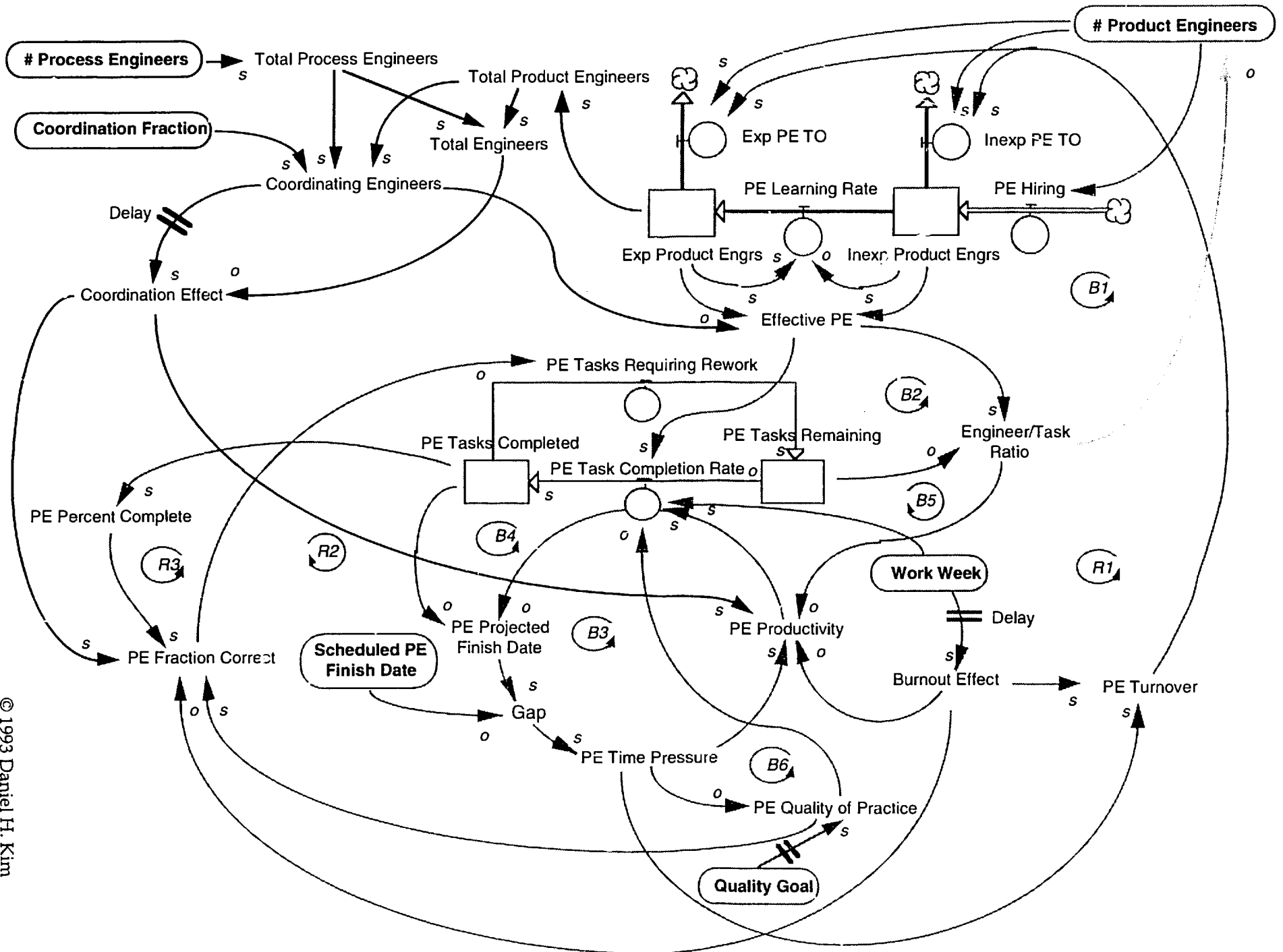
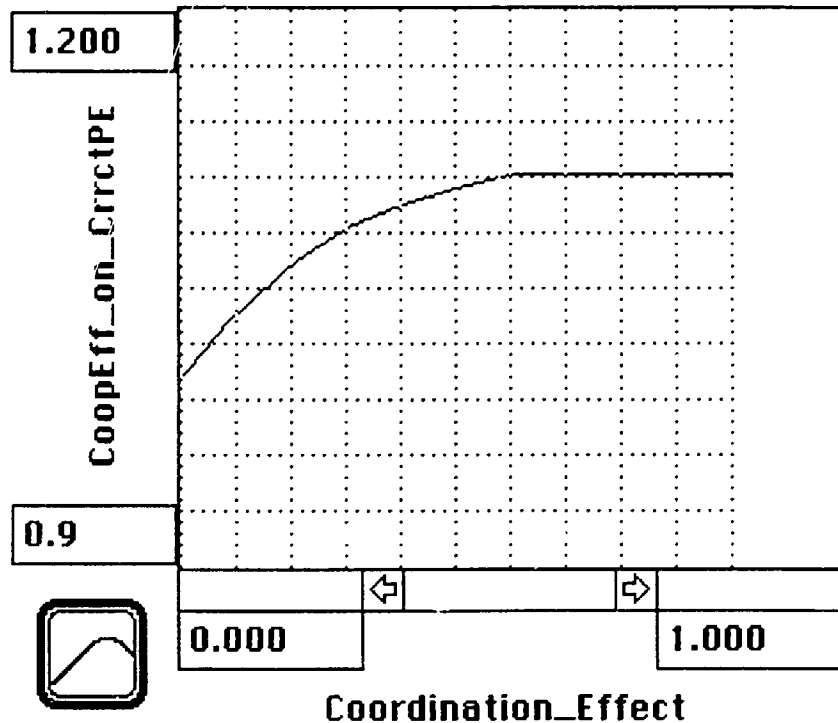


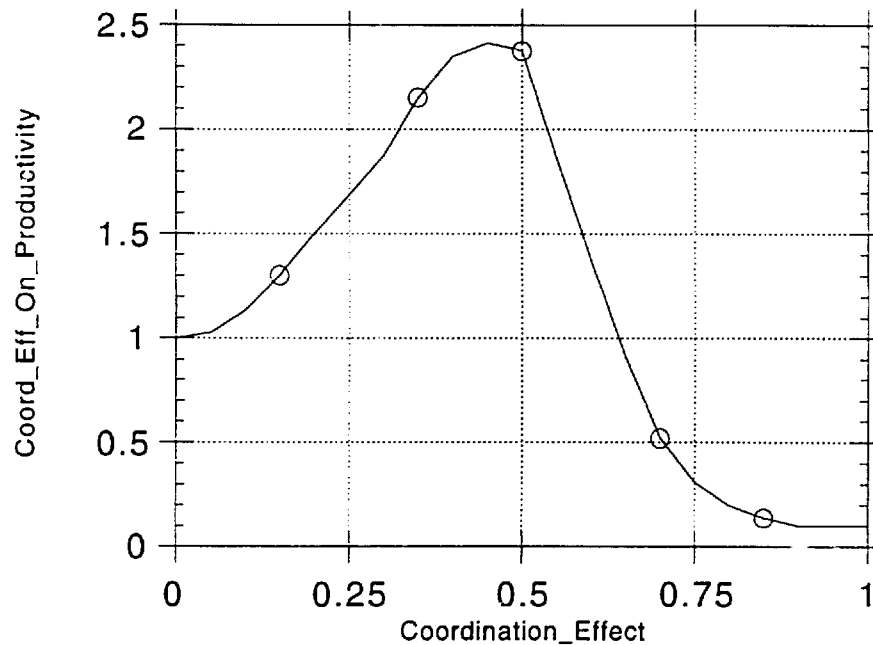
Figure 12. PE Causal Loop Diagram, adding Coordination.



The Coordination Effect on PE Fraction Correct

Figure 13

Another assumed impact of coordination is that it can increase productivity significantly. The research of Clark and Fujimoto has suggested that the lean production projects have achieved productivities of 2 1/2 time greater than the average mass production organization. While there is no data to numerically define the relationship between coordination and productivity, we do know that it requires a substantial investment in coordination to realize the full potential of productivity gain. The graphical function showing our assumed relationship between productivity and the coordination effect is displayed in Fig. 14. Again, no coordination will have no influence on the system so has output of 1. We have assumed that the optimum coordination effect is around .5. Above this, coordination effect has a negative effect on productivity which demonstrates the assumption that too much coordination will result in confusion and uncertainty about individual tasks. The exact relationship is not important. What is important is that it has gives the same challenge to the MFS user that a real manager would have in experimentally trying to discover what the optimum amount of coordination is. The coordination effect on productivity is the same for both the product engineers and the process engineers.



The Coordination Effect on PE Productivity

Figure 14

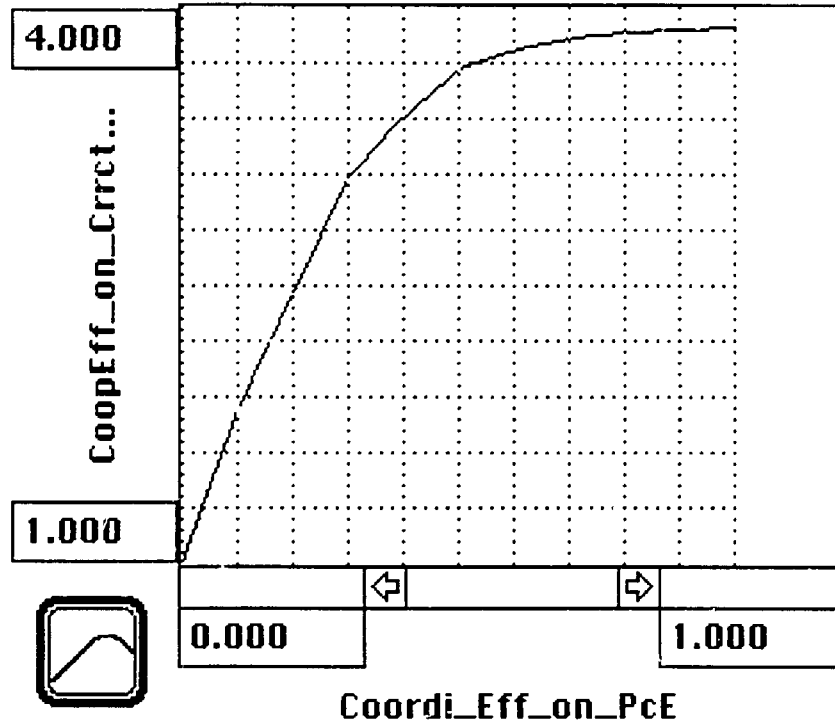
Two important data points were discussed when comparing lean production to mass production. The average productivity of a lean production project was 2 1/2 times greater while the project completion time was only 70% of the average mass production project. While the reason for this in a real project is quite complex, this discrepancy is accounted for in the model by the difference between the task completion rate and productivity. With a coordination effect of .5, the engineers work 2 1/2 time faster, but they also spend 50% of their time on coordination activities rather than completing tasks. The net gain is substantially less than 2 1/2 times. When the work saved from decreased required rework is also accounted for, using significant coordination in the MFS can reduce the overall time required for the project by about 30%. This is comparable with the data presented in Clark and Fujimoto and is accurate enough to convey the importance of coordination.

2.2 Process Engineers

The process engineering is the last phase in the product development project, other than the pilot run. The general causal loop diagram used to describe the process engineering system is almost identical to the product engineering

diagram except there is a separate decision to hire process engineers, an independent work week, and a separate scheduled completion date. The hiring decision is # **Product Engineers** and will add or remove process engineers in the same way the product engineering staffing works. The scheduled completion date for the process engineers is the **Scheduled Release Date** because the process engineers are assumed to be the last phase of the project and should be finished before the product is released into the market. The schedule is different from the product engineer's so that the process engineers can be scheduled to work through a different time phase of the project. Be aware that when the scheduled release date is reached, the product will be released into the market place whether or not the process engineers have finished all their tasks. This means that the customers will receive the product at whatever the quality the product has at the release date. The quality can be improved by finishing all the tasks, but it takes a long time to change the customer's perception of quality.

Because the process engineers are the downstream phase of the project, there are some minor feedback differences. The primary difference between the process and the product engineers is what influences the fraction of the tasks completed correctly. Since process engineering is essentially developing a manufacturing process to manufacture the product designed by the product engineers, the process engineers are dependent on the designs of the product engineers. If the product engineers have not completed their design, the process engineers will have an increased uncertainty about the needed manufacturing process and are less likely to do it correctly. The closer the product engineers are to completing their design (all tasks completed) the more likely the process engineers will be to finish their tasks correctly. In the microworld, the process engineering fraction correct is a function of the percent of the total product engineering tasks completed, the higher the percent completed, the higher the PcE fraction correct.



The Coordination Effect on Process Engineering Fraction Correct

Figure 15

The second difference is the influence of coordination on the fraction correct. Because the process engineering is the downstream activity, coordinating with the process engineers has a much more powerful influence. The graphical function that shows the assumed relationship between the coordination effect and the process engineering fraction correct is shown in Fig. 15. Again, there is no influence from a coordination effect of 0. As the coordination increases, the effect on the PcE fraction correct increases with diminishing returns. The influence on coordination is more complicated than the graphical function would suggest. Remember that the coordination is only one effect. The fraction correct is also influenced by burnout, the PcE quality of practice, the fraction of the total PE tasks complete, the inherent errors associated with any project. Although the graphical function suggests that the coordination effect can increase the fraction correct by a factor of 4, the fraction correct will never exceed the limits of inherent error. Also, coordinating does not prevent errors resulting from a low quality of practice or a high burnout from overwork. What the coordination effect does influence is the effect of the fraction of the PE total project complete. The process engineers can successfully design a manufacturing process even if

the product engineers are not finished if the coordination is high, indicating that the process engineers have a good understanding of the work begin done by the product engineers.

3. SIMULATOR CONTROLS

Part of facilitation a session of the management flight simulator is to explain the simulator interface the users will be interacting with. This section briefly covers the components of the "cockpit", the weekly management decisions, and the initial decisions available to the facilitator for changing the scenario.

3.1 Simulator Cockpit

When the flight simulator is launched, game screen in Fig. 16 will appear:

Menu Bar. Click and hold mouse button on the menu choice and drag the mouse to the desired command. Start a new game by selecting the restart option under the explore menu.

The "Cockpit"

Weekly Decisions

d_Process_Engin	0	↑
e_Coordination_F	.00	↑
f_Schd_PE_Finisl	25	↑
g_SchdReleaseDa	40	↑
h_Quality_Goal	1.00	↓

Step 1 time

Reports

1. Project Overview
2. Product Eng Status
3. Process Eng Status
4. Quaiity Report
5. Time & Cost Forecast
6. Project Summary
7. Sales & Marketing

Graphs/Tables

- A_Decisions2
- All_Staffing
- COP_Team_Costs
- Cost_of_Overtime
- Cost_of_Project
- Engineer_Experience_Mix

Decision Area: the decision you to change and the new value.

Click on the step button to advance simulation one v

Reports show t current project Click on the des report.

Graphs show h of the project. Click on desired grap

Use up and arrows to sc through chc

Click on graph l and drag to des viewing option.

Main Viewing Area. Any graph or report selected in the "cockpit" will appear here. To close a graph or report, click anywhere in the view, then click in the box that appears in the left corner of the gray bar at the top of the viewing area.

Product Development MFS "cockpit"

Figure 16

Decisions are displayed in the upper portion of the “cockpit”. The decisions are described in Management Decisions sections of this Guide and are separated into weekly and initial decisions. The list of reports is displayed in the second section of the cockpit. Each report selected will appear to left of the cockpit. In the figure above, the overview report is currently selected. The last portion of the cockpit displays the possible graphs to view. Toggle to the table view: by selecting the graph bar and choosing tables. For detailed information and variable definitions about reports and graphs, refer to the User’s Guide for this Management Flight Simulator.

3.2 Management Decisions

The management decisions are the possible weekly decisions in the management flight simulator. If the weekly decisions are not currently displayed in the cockpit, they can be found by toggling the decision bar in the simulator cockpit from initial decisions to weekly decisions. The consequences of each decision should be explored by returning to the causal loop diagrams developed during the Microworld section of this Guide.

- a) *PE_Work_Week*: Number of hours per week the product engineering team will work. A long work week (above 40 hours) increases the task completion rate, but can eventually lead to burnout if the work week is too high for too long. Burnout will lead to lower productivity, more errors, and a higher turnover. Beware that there may be overtime costs for work weeks above 40 hours.
- b) *PcE_Work_Week*: Number of hours per week the process engineering team will work.
- c) *Product_Engineers*: Number of new product engineers to add each week. Enter a negative number to remove product engineers. Additions to the product engineering team are assumed to come from a different project within the company, so are assumed to be competent engineers, but inexperienced because they are unfamiliar with this specific project.
- d) *Process_Engineers*: Similar to the product engineers, process engineers can be added or removed each week.
- e) *Coordination_Frac*: Fraction of total hours worked to be spent coordinating activities between the product engineers and the process engineers. The coordinating effect is based on the amount of time spent by the smaller of

the product and process engineering teams. For example, if you have 20 product engineers and 10 process engineers and the Coordination Frac is .2, the coordination effect will be based on 20% of 10 engineers from each team. Coordination between product and process engineers will allow them to exchange useful information.

- f) *Schd_PE_Finish*: This week is chosen as the completion date for all product engineering tasks. As the scheduled finish approaches, the product engineers will feel a time pressure if the time required to finish the project exceeds the time remaining in the schedule.
- g) *Scheduled Release Date*: The week number chosen for product release. The goal is to release the product on the target release date, an initial decision. When the scheduled release date is reached, even if the project is not complete, it is launched and the marketing force begins selling the product with the current quality. Because process engineering is the last phase of the project, the scheduled release date is the deadline for the process engineering team to complete all their tasks.
- h) *Quality_Goal*: This is the quality the engineers are striving to build into the product. A higher quality goal will pressure both the process and product engineers to have a higher quality of practice leading to a better final quality of product. The quality goal is defined as a quality relative to the competitors in the field. The competitors are assumed to have a quality of 1. For example, a quality goal of 1.2 indicates a desired product quality 20% better than the competitors.

3.3 Initial Decisions

The initial decisions are only changed at the beginning of a simulation in order to try a scenario based on a different set of assumptions. They are currently set to a "default" set of values which establish a standard product development scenario. These decisions are found in the initial decisions list. The values of the initial decisions can be found in the initial decision list and most are also shown in the Introduction Report of the MFS.

- A) *Target_Release_Date*: The original product release date set by marketing. If this date is missed, the product loses competitive advantage.
- B) *Core_PE_Team*: The number of experienced product engineers already committed to the project when it enters the product development phase. The core PE team will be automatically updated if any changes are made in the Scenario Scale.

- C) *Base_Productivity*: Base number of tasks that can be accomplished by an experienced engineer in a week. Both product and process engineers are assumed to have the same base productivity. Changing the Scenario Scale will automatically adjust the Base Productivity so that a different number of engineers will be requiring to finish the task in the same time.
- D) *Project_Budget*: Total budget allocated for this project in thousands of dollars. The budget is calculated by the estimating number of engineer-weeks it would take to finish the project working at the base productivity rate and multiplying by the expense per engineer week. A 10% cushion is added to the estimate. The project budget will automatically be updated with any changes in the Scenario Scale.
- E) *Eng_Salary*: The weekly engineering salary in thousands of dollars. The default is \$800/week, which is shown as \$.8 thousand in the actual initial decision listing.
- F) *Eng_Overhead*: The overhead is meant to include all the costs of supporting an engineering team beyond their salary. This includes benefits, equipment, support services, prototyping, etc. Default is \$1.2 thousand per engineer per week. Together, the engineering salary and overhead make up the total cost per engineer-week shown on the Introduction Report.
- G) *Overtime_Rate*: This is the fraction of the engineering salary to be paid for engineering overtime work. Overtime is considered to be any hours worked beyond the base 40 hour week. The default is 1.5, equivalent to time and a half. If no cost for overtime is desired, the overtime rate should be set to zero. The cumulative overtime costs are shown in the Time & Cost Forecast Report on the MFS.
- H) *Scenario_Scale*: The approximate scale of the project. A scale of 1 assumes that 100 engineers in 40 weeks would be able to finish the project. For a product development team of 20 people, a scenario scale of .2 would be appropriate. Changing the Scenario Scale will automatically change the *Core PE Team*, the *Base Productivity*, and the *Project Budget* to made the project suitable for different approximate team sizes.
- I) *Sched_Sensitivity*: The schedule sensitivity is the number of weeks ahead of the actual time that the engineers will feel time pressure from the schedule date. Changing this variable will change the sensitivity of the engineering teams to the approach of the completion date.

4. FACILITATING SIMULATOR SESSIONS

4.1 Suggestions On Running A Session

In order to run a session of the management flight simulator as an effective learning experience, the MFS must be placed in context. Otherwise, there is a tendency for the MFS to be used as a video game where the participants compete to win (finish with best time, etc.) and miss some of the potential learnings that can come from a more rigorous exploration of the product development system. This section is intended to provide a rough guideline of how to run an effective session of the management flight simulator. Once you have run a session or two, you should feel free to innovate.

Before beginning the session, make sure that the initial decisions are set for the scenario you desire. If you are just planning to use the default setting, check to make sure you know what the Target Date, Budget, and Base Productivity are. These will allow you to estimate the scope (magnitude of engineers required) of the project. Also, the MFS should be set to allow the user to go back and correct a mistake. The "allow go back" option under the Options menu must be checked to allow this feature. We have found the following format to be an effective structure for a MFS session. Each section will later be discussed in detail.

- Introduction:
 - Use of management flight simulators in general
 - Overview of product development microworld
 - How to use computer interface (simulator controls)
- Beginning the simulation:
 - Break the group up into teams of at least two people
 - Each team should have a User's Guide for reference
 - Introduce strategy sheets
 - Allow teams 15 minutes to familiarize themselves with the interface
- Simulation, should take around 45 minutes:
 - Each team should record their objectives & plans on the strategy sheet before starting the simulation
 - Facilitator should float around answering questions. The objective is not to tell the teams the optimum decisions to make, but rather to point out important feedbacks.

- When the teams finish, they should save each exploration under the team name they have used on the strategy sheet.
- Final results & comments should be recorded on the second page of the strategy sheet.
- Debrief:
 - Collect results from strategy sheets
 - Display results to the group (different experiments, not performance comparison)
 - Explore feelings during MFS
 - What appeared to be critical variables?
 - Discuss the role of coordination
 - Try to link discussion back into the microworld underlying the MFS
 - Can expand on the causal loop diagrams discussed during the introduction.
 - If time permits, run through another simulation using different teams.

4.1.1 Introduction To Management Flight Simulators

The first part of the introductory discussion should cover the general usage of Management Flight Simulators as learning tools. In this section, we will trace through the general introduction that has been used in previous learning labs. Again, this material is just intended to create an outline of how to run a session. A good reference source for general information on MFS is the chapter on *Microworlds: The Technology of the Learning Organization* in The Fifth Discipline by Peter Senge.

Management flight simulators are intended to create a managerial practice field. In all other arenas where teams are expected to perform they have the ability to practice. When we think of great teams, we tend to think of football teams, ballets, etc. All of these teams have the ability to work together until they understand the game. It is hard to image one football player saying to another at the end of a game, "So when's the next game? Great, I'll see you then". Yet this is exactly what happens for real product development teams. The only chance to manage a product development project, or to be part of the team, is during a real project when substantial personal and corporate investment is linked to the

performance of your team. With that much riding on a project, learning about managing through rigorous experimentation is impossible.

A practice field has several unique characteristics:

- Control the pace by speeding up or slowing down time. Phenomena that stretch out over many months can be reduced to several minutes to show the long-term consequences of decisions.
- Compressing space. In a practice field, all of your environment is reduced to a manageable size so that the consequences of decisions and actions are visible
- Repeated trials. Sometimes “plays” must be repeated to learn what went wrong, or even to understand what variables helped lead to success.
- Pauses for reflection. To learn from doing, there must be an opportunity to sit down with the team, or with other managers, and try to understand why the system behaved the way it did in response to the strategies used.
- Safe environment to experiment. The idea of a practice field is to create a safe place to experiment with different strategies without worrying about the consequences of failure. Sometimes the greatest learnings come from failing, not succeeding.
- Appropriate tools and equipment. Without the proper environment, it is impossible to practice.

We have called these managerial practice fields ‘management flight simulators’ because we view them as similar in concept to a flight simulator used by pilots rather than a video game. The goal of a video game is to perform and get the best possible score. In contrast, the goal of a flight simulator is to learn how to fly. This involves experimenting with different strategies, varying the conditions, trying maneuvers that will cause failure. But throughout this process, learning about the system and understanding the short term and long term consequences of decisions.

The reason that practice fields are effective is that human beings learn best through doing. We act, observe the consequences of our actions, and adjust our actions to get closer to our goal. But learning only happens when the feedback is fast and local. When cause and effect are separated by space and time, the consequences of our actions become so tangled with other variables and concerns that we no longer recognize them as connected directly to the actions. The computer simulation allows time and space to be compressed. In the microworld we have created, the consequences from actions are fast and local, but still in a complex system. This makes learning possible. The computer also contains the tools and equipment necessary to practice running a project. It simulates the "tasks", the engineers, the time pressure, and other aspects of a real project. One important thing to stress during management flight simulator sessions is that performance is not important unto itself. More learning about a system takes place while trying to battle failure than when coasting through a successful project.

4.1.2 The Product Development MFS

Having placed the management flight simulator in the context of a learning environment, it is time to introduce the details of the product development MFS. A full product development project in the automotive industry is typically separated into 6 phases; concept generation, product planning, advanced engineering, product engineering, process engineering, and pilot run. We have chosen to model the product and process engineering phases because the managerial challenges are more evident than the technical ones in these phases. Additionally, approximately 1/2 of the total disadvantage to the Japanese in the Automotive industry occurs during these phases.

During a session, each of the participants will have the opportunity to "manage" a product development project. To introduce the MFS, you (as facilitator) should run through the introduction that is Section 1.1 of this Guide. Describe the basic project goals and the decisions the participants will be expected to make each week. These decisions should be placed in context by describing (briefly) the microworld that underlies the management flight simulator. Generally, we do *not* develop the full causal loop diagram presented in this Guide before the participants get to use the simulator. Instead, the most important information is presented. Sometimes the full causal loop diagram of the microworld can be discussed during the session debriefing - after the

participants have had exposure to the system. The following points are necessary before starting the simulation:

- **The task structure:** The objective of the MFS is to complete all the product and process engineering tasks within the time, budget, and quality specified. Go over the stock and flow diagram that was presented as Figure 2 to show how rework fits into the model of task completion.
- **Staffing:** First show the staffing stock and flow structure and discuss the difference between experienced and inexperienced engineers and how the hiring works. This was shown in Figure 3 of this Guide.
- **Task & Staffing:** To show how the two stock and flow structures interact, show Fig 4. This also shows the feedback loop between the engineering/tasks ratio and the number of tasks remaining in the project.
- Rather than developing the full causal loop diagram, the rest of the microworld can be generally described using the Microworld Overview Diagram presented in Figure 1. Using this diagram, you can discuss the rest of the available decisions. Here is a quick outline to use when discussing the decisions. Please add more from what you have learned about the microworld.
 - *Staffing* (previously discussed when presented the staffing diagram)
 - *Schedule:* There are dates for both teams, be sure to tell how time pressure will increase the tasks completion rate but will eventually decrease the quality of practice. The product release date pressure the process engineers, and is the date when the product will be released into the marketplace. High time pressure will also cause engineers to turnover.
 - *Quality:* The Quality Goal and time pressure influence the quality of practice. High quality of practice creates less rework, but

increases the time required to complete each task. Tasks finished at the Quality of practice become parts of the quality of product.

- *Workweek*: Different work week for each team. Increases task completion rate, but will eventually cause burnout if too high for too long, leading to increased rework and lower productivity. The engineers might also be required to be paid overtime for work week greater than 40 hours. The exact rate is shown on the Introduction Report of the MFS.
- *Coordination Fraction*: Fraction of each week to be spent on coordination activities. Immediate effect is to reduce the task completion rate. After a time delay, it will improve productivity and reduce rework. Be sure to explain the meaning of coordination as it is used in the MFS. Give an example of the coordination effect if there are two different team sizes.

After the scheduled release date has been reached, the product will be released into the market place. If you are not ready to release the product, be sure to let the product release date slip. When the product is released into the market place, your company will try to sell it. The resulting sales are shown on the Sales & Marketing Report and in the graphs. The sales of your product will depend heavily on whether the product was released by the date set by marketing and on what quality the customer perceive you product at. You can continue to work on the project after it has been released, but it difficult to adjust the customers initial perception of quality. After the project is completed (all 100 PE & PcE tasks) all of your engineers will be automatically removed from the project. You can change your time step to continuously and what the sales and marketing efforts. All initial sales will result in losing money because the model assumes that there is a learning curve to production that will cause the production cost to drop as more units are produced.

Note that the model you are presenting is not reality. It is not meant to be reality. All models should be evaluated on the basis on their usefulness with respect to their intended purpose. With the introduction to the microworld, you will have given the participants a broad, fuzzy picture of the exact computer model. This is quite similar to the understanding a real manager might have

when beginning a new project. Aware of the rules and general expected behavior, but nothing more. Part of the objective of the simulator is for the participants to begin learning about the system by experimenting with decisions and trying to understand the feedback from the model. The art of answering questions during the simulation is to encourage this experimental behavior, giving broad pointers, rather than telling what the optimum decisions are.

4.1.3 Beginning The Simulation

Now we are ready to move to the computers. If you have access, the easiest way to begin the computer usage is to have a computer projecting through an overhead projector. Otherwise, use transparencies of the “cockpit” and each of the reports. Show the general structure of the cockpit, where the decisions, reports, and graphs are and how to enter a decision or select a graph or report. Also demonstrate how you can scroll through the complete list of available decisions and reports. Don’t forget that you can toggle between graphs and tables. Simulate forward a few weeks in time to demonstrate the simulator. Next, we usually explain each of the reports quickly to show where various information can be found. It is important to explain the different computer projections shown in the report and point out that they (the participants) should only believe the projections about as much as they would believe one from their own company.

The participants should be broken into teams of at least two people. We have found that team playing is more effective as a learning experience because the decisions and strategies have to be discussed. We usually allow the teams 15 minutes or so just to play around with entering decisions and getting familiar with the information system. Each team should have at least one copy of the User’s Guide which contains definitions for all the variables in the reports and graphs.

4.1.4 The Simulation

After the participants have had a chance to get a little familiar with the MFS, we introduce a strategy sheet. Each team should use a sheet for each game. This helps them develop a strategy in both words and by sketching their hiring and coordination strategy over time. The first page of the strategy sheet should be filled in before beginning a real simulation. The second page is for final information after the game. One of the common difficulties with forming a

strategy is to decide on the scope of the engineers needed to finish the project in time and under budget. This can be easily done by referring the participants to the Introduction Report of the MFS. This shows what the time and budget constraints are, what the base productivity of the engineers is, and what it costs to employ each engineer for one week. For this, it is quite simple to calculate the number of engineers required to finish the project by the chosen scheduled completion dates, assuming the engineers work at the base productivity through the project.

The first run through the simulation usually takes between 45 min. to 1 hour. When teams finish the simulation, we try to have them save the exploration under their team name and record all of the relevant information on the second page of the strategy sheet. This saves all the decisions and model feedback from the game and may be useful for future research on simulator behavior.

4.1.5 Outcome & Debrief

To begin the simulation debrief, we usually collect all the results from the strategy sheet and make a comparative list of:

- Percent Time Used
- Percent Budget Consumed
- Quality at Release Date
- Final Quality
- Final Profit
- Optional: Total Required Rework

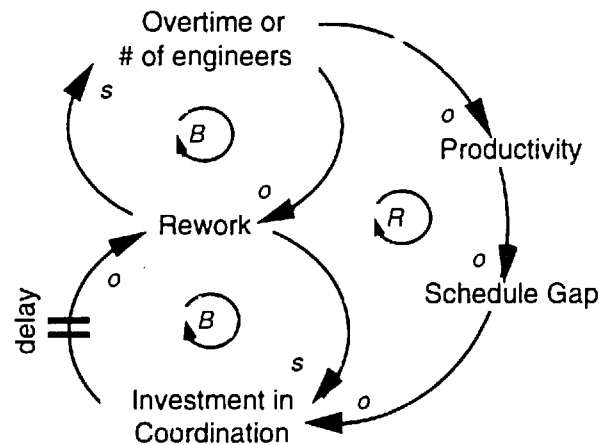
The results are not listed just to compare performances. Rather, each of the teams has just managed an identical project in an identical product development world. Yet the outcomes have probably varied significantly. The only real difference between the projects is in the mental models of the participants that governed their decision making process. Each one of the outcomes is an “experimental” result of how a particular mental model interacted with the product development system.

The second interesting point from examining the results is when it comes to measuring performance. The first question most of the teams ask is how did we do? But that question can only be answered against a bench mark. If everybody misses the original project goals, the bench mark simply becomes the team that did the best. Their results become the measurement for success. Yet it really has nothing to do with the potential “best” that could be achieved in managing the

system. Although we have demonstrated that you can indeed achieve the project objectives, most companies do not have the advantage of knowing what is really achievable in a project and merely set their goals according to the best past performance. This happened in the American automotive industry. They had a vision of what a “good” product development performance was until the Japanese competed with their version of a “good” performance. This forced the American industry to reassess the way it defined a good project.

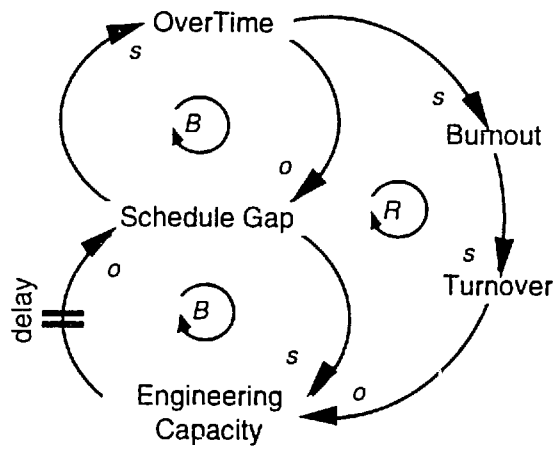
The rest of the debrief should be a discussion of how people felt while they were “managing” the simulator, if they felt in control, what they thought were important variables to what, and so forth. The issue of coordination be explicitly discussed. What are the tradeoff’s between spending time on coordination activities for the long term gain in reduced rework and increased productivity? How does this to relate the participants’ work experience? If it proves useful to the discussion, in the past we have returned to the more complex causal loop diagrams to show the consequences of the various decisions.

One excellent way to help people discuss their experience with the Flight Simulator is to use the archetypes to show some common decision making behaviors. I have included here two Shifting-the-Burden examples that have come up in past learning labs, Figs 17 and 18.



Shifting-the-Burden Response to Increased Rework

Figure 17



Shifting-the-Burden Response to Falling Behind Schedule

Figure 18

Appendix E:
New Product Development Management Flight Simulator
User's Guide

Prepared by
Donald A. Seville & Daniel H. Kim
Organizational Learning Center
MIT Sloan School of Management
1 Amherst Street, E40-294
Cambridge, MA 02139
(617) 253-1587

Table of Contents

1. Introduction.....	404
2. Getting Started.....	405
2.1 Management Decisions	407
2.2 Strategy Suggestions	409
3. Information System Summary	410
3.1 Reports.....	410
3.1.1 Report 0. Introduction.....	412
3.1.2 Report 1. Project overview	413
3.1.3 Report 2. Product Engineering Team Status	415
3.1.4 Report 3. Process Engineering Team Status.	417
3.1.5 Report 4. Quality Report.....	419
3.1.6 Report 5. Time & Cost Forecast	420
3.1.7 Report 6. Project Summary.....	422
3.1.8 Report 7. Sales and Marketing Report.....	423
3.2 Graphs & Tables.....	424

1. INTRODUCTION¹

Our increasingly interconnected and dynamic world challenges managers to find new ways to understand and control change. The accelerating rate of technological, organizational, and social change means managers are faced with situations that are in many ways new, and must increasingly deal with the unexpected. Managers are not alone in facing such daunting tasks. Modern society is built upon systems of enormous complexity, from nuclear power plants to jumbo jets. A pilot, for example, must also control a system of great complexity and be prepared for the unexpected. There is, however, one significant difference between the pilot of a jet and the manager of a project. No one would dream of sending a pilot up in the real thing before they had extensive training in the flight simulator on the ground. The simulator allows the pilot to learn, to make mistakes, to experience the unexpected without risk to passengers or aircraft. Yet managers are expected to fly their projects into unknown skies with their only training being in management “ground school” or experience as junior crew members.

The Product Development Flight Simulator (MFS) gives you the opportunity to “pilot” a product development process. The challenge in managing a product development program is to meet three objectives: finishing in a specified **time** limit, under an allocated **budget**, and with a competitive **quality**. The project is separated into two categories of tasks that must be completed. There are 100 generic product engineers “tasks” and there are 100 generic process engineering “tasks”. The product engineers (PE) do the product design and testing, while the process engineers (PcE) design and build the manufacturing process to turn the design into a product. When time reaches the scheduled release date, the product is released into the market and will be sold, even if all the engineering tasks are not complete. The sales depend heavily on whether the product was released by the target release date with a competitive quality.

As the “pilot” of this simulator, you are expected to make decisions each week about adding or removing engineers, the desired number of work hours in a week, the scheduled completion date for each team, the amount of time spent coordinating between the two teams, and quality “goal” for the product. Your decisions will influence the rate at which the engineers can accomplish tasks, the

¹ Parts of this introduction are adopted from the *People Express Briefing Book*, an unpublished MIT document by Professor John Sterman.

quality they will build into the project, and even the rate they will leave the project because of stress. You may finish far over budget, or over targeted release date, or even with a low product quality. But there is no winning or losing. The purpose of the flight simulator is to give you insight into the product development issues raised and to begin to understand the dynamic interconnections that exists within the system. The flight simulator is a laboratory in which a wide variety of different management strategies can be tested. More importantly, the consequences of various strategies can be explored systematically without risking the futures of a real firm. Or risking a real career.

Most of all, enjoy yourself. This is an experimental laboratory so don't be afraid to try new strategies. You might want manage the first few simulated projects using the strategy you think will most likely to succeed. In later trials, you may wish to systematically vary aspects of your strategy to gain an understanding of how the product development process responds to management decisions. Remember, sometimes you learn more by having to "pilot" through rough weather, poor visibility, and with unexpected mechanical failures. The beauty of a flight simulator is you can walk away from every crash landing.

2. GETTING STARTED

There are three major components of the management flight simulator itself. These are the *microworld*, the *information system*, and the *simulator controls*.

Microworld: The microworld is the systems model that underlies the flight simulator. Here the interactions between the different variables are explicitly described. It has been tested and calibrated; but like any model, it is a simplification of reality, based on a set of assumptions about the product development process. For a detailed description, refer to the Facilitator's Guide.

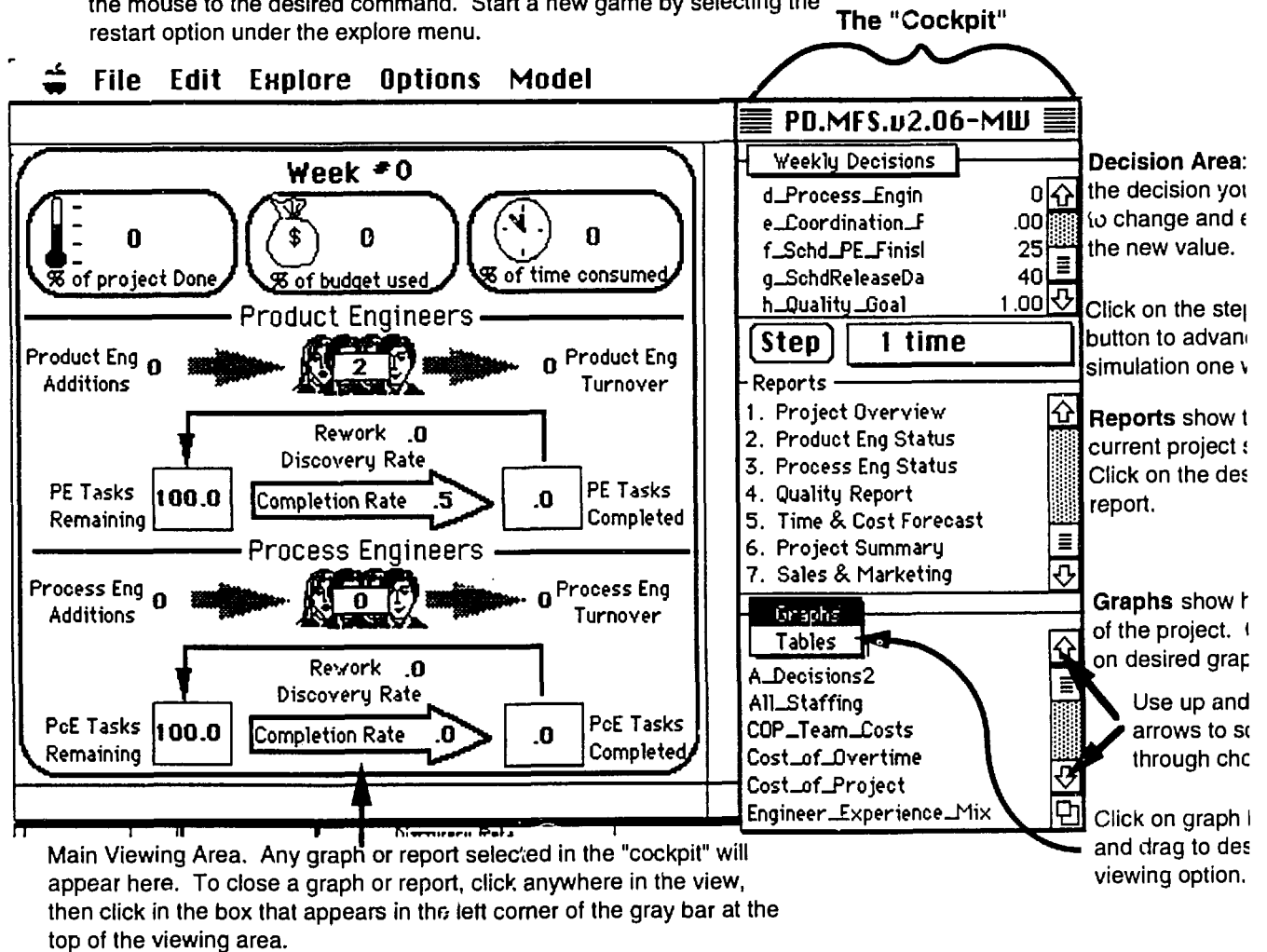
Information System: The product development progress is monitored through a complex information system. You have access to multiple reports that detail the current status of different facets of the project. Historical information can be

obtained as either graphs or tables and show the behavior of the variables over time.

Simulator Controls: The simulators controls are in the "cockpit" of the flight simulator. Here you can make decisions and control the information system by selecting the report or graph you would like to view.

When the flight simulator is launched, the computer screen shown in Fig. 1 will appear. Decisions are displayed in the upper portion of the "cockpit". Selecting and holding the arrows to the right of the decision names will scroll through the list of decisions. Change a decision by selecting (place the mouse pointer over and clicking) and entering the desired value.

Menu Bar. Click and hold mouse button on the menu choice and drag the mouse to the desired command. Start a new game by selecting the restart option under the explore menu.



Product Development MFS "cockpit"

Figure 1

There are two types of decisions in the simulator, initial decisions and weekly decisions. Toggle to the next decision list by clicking the mouse on the decision bar at the top of the decision box in the cockpit. The weekly decisions are described in Management Decisions sections of this Guide. The initial decisions have all been previously set to create the desired scenario, so there is no need to change them.

The list of reports is displayed in the second section of the cockpit. There are 7 reports that give the current information on different aspects of the product development project. Please read the introduction report before beginning the simulation to see what the specific goals and initial parameters are for the project you are going to manage. Select a report simply by "clicking" on the report name with the mouse. The report selected will appear to left of the cockpit. In the figure above, the overview report is currently selected. A printout of each report with the definitions of the relevant variables are included in the Information System Summary of this Guide.

The last portion of the cockpit displays the possible graphs to view. Each graphs contains multiple variables that are listed in the Graphs & Tables section of the Information System Summary. Toggle to the table view by selecting the graph bar and choosing tables.

2.1 Management Decisions

As manager of this simulated project, there are eight decisions you can make each week to guide the project. These are called the weekly decisions and are described in detail below. This is another set of decisions that can be accessed by toggling the decision bar in the cockpit, the initial decisions. These decisions are used only to set up the scenario of the simulation and should **not** be changed during a simulation. If you would like to learn more about the initial decisions, refer to the MFS Facilitator's Guide.

- a) *PE_Work_Week*: Number of hours per week the product engineering team will work. A long work week (above 40 hours) increases the task completion rate, but can eventually lead to burnout if the work week is too high for too long. Burnout will lead to lower productivity, more errors, and a higher turnover.
- b) *PcE_Work_Week*: Number of hours per week the process engineering team will work.

- c) *Product_Engineers*: Number of new product engineers to add each week. Enter a negative number to remove product engineers. Additions to the product engineering team are assumed to come from a different project within the company, so are competent engineers, but inexperienced because they are unfamiliar with this specific project. They become experienced on a learning curve that is based on the availability of experienced engineers to familiarized them with the project. The inexperienced engineers are not as productive as the experienced.
- d) *Process_Engineers*: Similar to the product engineers, process engineers can be added or removed each week. The learning curve for the process engineers is similar except the learning process can be improved by coordinating with the product engineers.
- e) *Coordination_Frac*: Fraction of total hours worked to be spent coordinating activities between the product engineers and the process engineers. The coordinating effect is based on the amount of time spent by the smaller of the product and process engineering teams. For example, if you have 20 product engineers and 10 process engineers and the Coordination Frac is .2, the coordination effect will be based on 20% of 10 engineers from each team. Coordination between product and process engineers will allow them to exchange useful information such that they can improve productivity and avoid making errors, thus reducing the necessary rework. However, the coordinating staff will have to spend some of their productive time on meetings which reduces the number of tasks completed each week.
- f) *Schd_PE_Finish*: This week is chosen as the completion date for all product engineering tasks. As the scheduled finish approaches, the product engineers will feel a time pressure if the time required to finish the project exceeds the time remaining in the schedule. Time pressure will increase productivity, but the quality of practice can decline if the engineers are under too much pressure to work fast.
- g) *Scheduled Release Date*: The week number chosen for product release. The goal is to release the product on the target release date, an initial decision. When the scheduled release date is reached, even if the project is not

complete, it is launched and the marketing force begins selling the product with the current quality. Because process engineering is the last phase of the project, the scheduled release date is the deadline for the process engineering team to complete all their tasks. If the estimated time required for the process engineers exceeds the time left in the schedule near the end of the project, they will feel a time pressure which influences their working ability.

- h) *Quality_Goal* : This is the quality the engineers are striving to build into the product. A higher quality goal will pressure both the process and product engineers to have a higher quality of practice leading to a better final quality of product. The quality goal is defined as a quality relative to the competitors in the field. The competitors are assumed to have a quality of 1. For example, a quality goal of 1.2 indicates a desired product quality 20% better than the competitors.

2.2 Strategy Suggestions

Before starting the management flight simulator, it is quite useful to think up a strategy of what decisions you are going to make over time such as adding engineers, scheduled finish dates, coordination, and quality. To help you formulate a strategy, and to help us gather information about the simulator, you will be requested to maintain a strategy sheet during each session of the product development management flight simulator. This sheet requests that you records the goals of the project, the initial parameters, and the strategy you intend to follow for each simulation. One common difficulty in forming a strategy is to decide the scope of the number of engineers required to finish the project in time and under budget. One way to do this is to estimate the number of engineers that would be requiring to finish the project in the time you would like if they were working at the *base productivity*. The base productivity, target release date, and budget are all displayed on the Introduction Report in the actual management flight simulator. For budget calculations, the cost of hiring each engineer for a week is also included in the Introduction Report.

Another tip to deciding on the scope of the project is to check out the forecasted required number of engineers. These forecasts are done for both the process and the product engineers and are shown in the respective Engineering Team Status Reports. These estimates are the number of engineers that would be required to finish the project by the scheduled finish dates if they worked at the

base productivity throughout the project. However, trust these forecasts as far as you would trust a forecast in your real job. Engineers cannot always be expected to work at the base productivity.

A couple of warnings. Like in real life, some actions have long term consequences. Prolonged high time pressure can cause people to leave the project. Extended long work weeks can decrease the quality of the work being done. Too many engineers with too few tasks can be individually unproductive. If given no time pressure, engineers tend to keep engineering to increase the quality of the part they are working on. These behaviors can be discovered by attending to the information being fed back to you through the simulator reports and graphs.

Good Luck, have fun, and please record your thoughts and actions on the second page of the strategy sheet when the simulation is over. If you have questions about the meanings of variables in the flight simulator, the information system summary of this report describes the variables found in each report, and also the variable found in each graph/table by the order they appear in the MFS.

3. INFORMATION SYSTEM SUMMARY

3.1 Reports

Reports provide weekly feedback about the project. There are eight reports:

Introduction	Displays the project budget , target release date, and other important parameters of the management flight simulator. These parameters are all initial decisions and may vary depending on the scenario chosen for each session. It is important to check this report each simulation to review the goals and parameters.
Project Overview	Overview of project goals and product and process engineering teams.
Product Eng Status	# of Product Engineers, task completion status, and estimated completion date.

Process Eng Status	# of Process Engineers, task completion status, and estimated completion date.
Quality Report	Quality goal, quality of practice, known rework, and quality of product.
Cost & Timing Forecast	Forecasted project completion date and project cost.
Project Summary	Summary of project goals and productivity.
Sales & Marketing	After release information on sales and marketing progress.

There is some duplication of information, so you should feel free to use only the reports that have the information format comfortable to you. To show a report, simply click on the desired report name in the report box of the cockpit. A box containing the report will appear to the left of the cockpit.

3.1.1 Report 0. Introduction

Introduction

New Product Development

Your challenge, as pilot of this management flight simulator, is to complete a development project with the following goals:

Release product by week # **40**
with a budget of **\$1760** thousand dollars
and a competitive final product quality ϕ 1)

The entire project consists of:

100 generic Product Engineering (PE) tasks
100 generic Process Engineering (PcE) tasks

Each engineer has a "base" productivity of **.250** tasks per engineer-week. The cost per engineer is **\$2000/week**, which includes salary (**\$800/week**), benefits, and other overhead costs. The overtime rate is **1.5** times the salary.

There are **2** engineers in the Product Engineering team at the beginning of the project.

Project Goals:

- Target Release Date** Your objective is to complete and release the product by this week. Releasing after this date will result in lost competitive advantage in the marketplace.

- Budget** Allocated money for this project. This restricts the number of engineers you can afford to hire.

- Product Quality** The competitor is assumed to have a product quality of one. Your goal is to achieve a product quality equal to or better than your competitor's.

Initial Parameters:

- Engineering Tasks** Number of tasks for both the product and the process engineers to complete by the target release date.

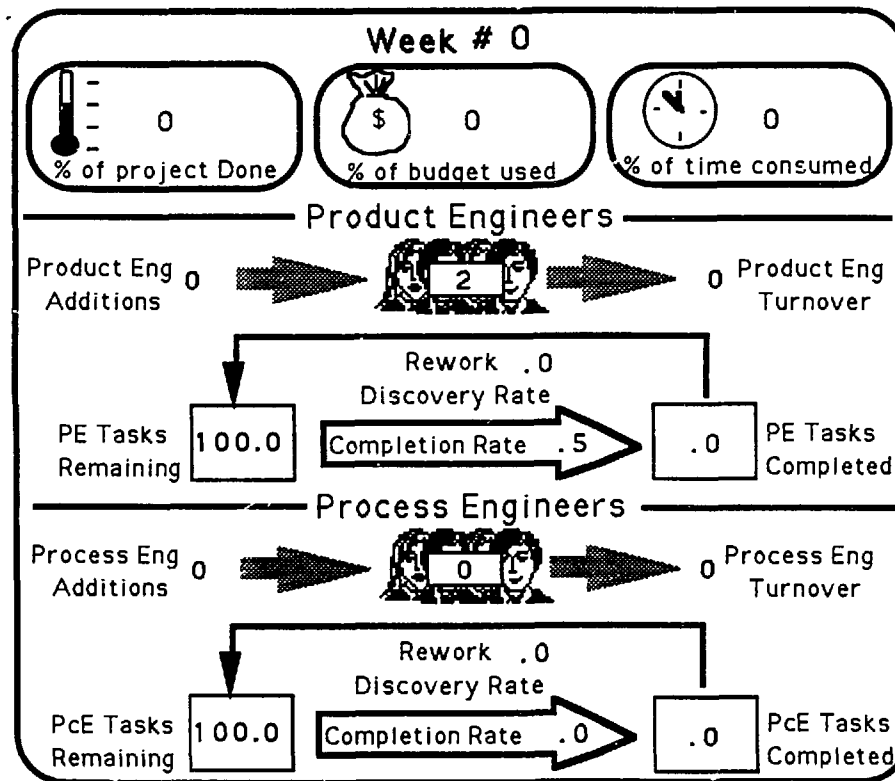
- Base Productivity** Estimated number of tasks one experienced engineer would complete in a week if nothing else were influencing productivity.

Expense per engineer Cost to maintain each engineer for one week, includes both salary and overhead. Salary and overhead are initial decisions

Overtime Costs Based on the overtime rate, which is the fraction of the base salary for overtime. For example, an overtime rate of 1.5 is time and 1/2. One engineer working a 50 hour week with a salary of \$800/week would charge 1/4 week * \$800 /week * 1.5 overtime rate or \$300 overtime.

Initial Product Engineers Core team of experience product engineers starting the project.

3.1.2 Report 1. Project overview



The overview report is divided into three sections (see picture of report).

Project Goals Overview:

Week #

Current week of the simulator. Simulations run for 100 weeks.

% Project Complete

Percent of entire project (total tasks) complete.

% Budget Consumed	Percent of project budget currently consumed.
% Time Consumed	Percent of time in the original target date consumed.
Product Engineering Team PE Additions	(Process Engineering has similar variables) Number of product engineers added or removed each week. PE additions are made by using the c_product_engineers decision. (weekly decision)
Total Product Engineers	Current number of product engineers includes both experienced engineers and inexperienced engineers. (the total number of product engineers is displayed in the box superimposed on the picture of people in the product engineering portion of the report.)
PE Turnover	Product engineers voluntarily leaving the project each week. This is a natural turnover of engineers resulting from burnout or extended high time pressure.
PE Tasks Remaining	Both product and process engineers begin the project with 100 tasks to complete. The product engineering tasks remaining are shown here.
Completion Rate	The product engineers complete tasks as a function of the number of product engineers and their productivity, skill level, work week, and quality of practice.
PE Tasks Completed	Number of product engineering tasks complete.
Rework Discovery Rate	A certain percentage of the tasks completed by the product engineers are incorrect and required rework. Each task discovered to require rework is taken out of the task completed pool and put back into the tasks remaining pool. The fraction of tasks done incorrectly is a function of the product engineering time pressure, their quality of practice, the amount of burnout from overwork, and the level of coordination with process engineering. The tasks requiring rework are not instantly known but rather are discovered as a function of how much of the project is completed and the level of coordination with process engineering.

3.1.3 Report 2. Product Engineering Team Status

Week # 0			
Product Engineering Team Status			
Experienced Engineers (engineers)	2	PE Work Week (hours/week)	40
Inexperienced Engineers (engineers)	0	Sched PE Completion (weeks)	25
Engineer Turnover (engineers/week)	0	Projected Completion Date (weeks)	25
Forecasted Required Engrs	15	PE Completed (0=no, 1=yes)	0

PE Tasks Remaining (tasks)	100.0
PE Task Completion Rate (tasks/week)	.5
PE Tasks Completed (tasks)	.0
PE Cumulative Rework (tasks)	.0

Experienced Engineers The number of experienced product engineers.

Inexperienced Engineers Engineers added through the product engineer decision are assumed to be inexperienced. Inexperienced is defined as a fully trained engineer, but one unfamiliar to the project. After a certain training period that is a function of the availability of experienced engineers for teaching, the inexperienced engineer will become an experienced engineer. Inexperienced engineers are not as productive as experienced engineers. However, a decision to remove engineers will first remove inexperienced engineers because they are less productive. Only after there are no more inexperienced engineers will experienced engineers be removed.

Engineer Turnover Number of product engineers leaving the project. Voluntary leaves resulting from time pressure and burnout will happen for both inexperienced and experienced engineers.

Forecasted Required An estimate of the number of engineers required to

Engineers	complete the project by the scheduled finish date working at the current estimated product engineering productivity.
PE Work Week	Desired number of hours in the product engineering work week. There may be overtime costs for work weeks over 40 hours/week. Check the Introduction report to see if the engineers must be paid for overtime. The length of the work week directly affects the task completion rate. The more hours worked, the more tasks can be completed. However, if the work week is too high for too long, the engineers will begin to burnout. Burnout leads to lower productivity, more errors, and a greater engineering turnover from the stress. (weekly decision)
Sched PE Completion	Desired date for the product engineers to finish. (weekly decision)
Projected Completion	Projected product engineering completion date based on the current number of engineers and the current task completion rate.
PE Completed	A switch to show when the product engineers have finished. 1 indicates all tasks are complete and 0 indicates there are still tasks remaining.
PE Tasks Remaining	Number of product engineering tasks remaining.
PE Task Completion Rate	Current task completion rate, a function of the number of engineers, their productivity, work week, and level of coordination.
PE Tasks Completed	Number of product engineering tasks completed.
PE Cumulative Rework	Product engineering tasks discovered to have been done incorrectly are returned to the stock of tasks remaining. The cumulative rework acts as a final measure of what percent of the tasks had to be reworked. Tasks are done incorrectly as a function of the quality of practice, burnout from overwork, and coordination with process engineers.

3.1.4 Report 3. Process Engineering Team Status.

Week # 0			
Process Engineering Team Status			
Experienced Engineers (engineers)	0	PcE Work Week (hours/week)	40
Inexperienced Engineers (engineers)	0	Sched Release Date (weeks)	40
Engineer Turnover (engineers/week)	0	Projected Completion Date (weeks)	25
Forecasted Required Engrs	10	Project Completed (0=no, 1=yes)	0
PcE Tasks Remaining (tasks)		100.0	
PcE Task Completion Rate (tasks/week)		.0	
PcE Tasks Completed (tasks)		.0	
PcE Cumulative Rework (tasks)		.0	

Experienced Engineers Number of experienced process engineers

Inexperienced Engineers Additions to the process engineering team are also assumed to be inexperienced. The learning curve is similar to that of the product engineers, but coordination can improve the learning time because the product engineers can familiarize the process engineers with the project.

Engineer Turnover Number of process engineers leaving the team each week due to burnout or extended time pressure.

Forecasted Required Engineers An estimate of the number of engineers required to complete the project by the scheduled release date working at the current estimated process engineering productivity.

PcE Work Week Desired number of hours in the process engineering work week. There may be overtime costs for work weeks over 40 hours/week. Check the Introduction report to see what the engineers must be paid for overtime. The length of the work week directly affects

the task completion rate. The more hours worked, the more tasks can be completed. However, if the work week is too high for too long, the engineers will begin to burnout. Burnout leads to lower productivity, more errors, and a greater engineering turnover from the stress. (weekly decision)

Sched Release Date	Scheduled week # for project release and for all process engineering tasks to be complete. (weekly decision)
Projected Completion	Projected product engineering completion date based on the current number of engineers and the current task completion rate.
Project Completed	Displays 1 when all the process engineering tasks are complete.
PcE Tasks Remaining	Current number of process engineers tasks remaining. 100 total tasks to complete over the course of the project.
PcE Task Completion Rate	Number of tasks the process engineering team completes each week. The rate is a function of the number of engineers, their productivity, skill level, work week, and quality of practice.
PcE Tasks Completed	Number of process engineering tasks complete.
PcE Cumulative Rework	A fraction of the process engineering tasks are found to be done incorrectly. Upon discovery, they are removed from the tasks completed stock and returned to the tasks remaining stock. The cumulative rework shows the current total number of tasks that were discovered to require rework. The fraction of the process engineering tasks incorrect is a function of the PcE quality of practice, their burnout, the percent of the product engineering tasks completed incorrectly (faulty designs sent over), the amount of the total product engineering tasks completed, and the coordination with the product engineers.

3.1.5 Report 4. Quality Report

Week # 0	
Quality Report	
Quality Goal (1=competitor)	1.00
Quality of Practice (1=competitor)	
Product Engineering Team	1.00
Process Engineering Team	1.00
	.00
Quality of Product (1=competitor)	.00
Customer Perceived Quality	1.00
<u>Cumulative Known Rework {Tasks}</u>	
Product Engineers	.0
Process Engineers	.0

- Quality Goal** Desired final quality of the product. The competitor is assumed to have a quality of 1. A quality goal of 1.2 would indicate a desired final product quality 20% better than the competitor. (initial decision)
- Quality of Practice** Current quality being built into the product. Both the process and the product engineers strive to work at the quality goal, but because it takes more time to practice high quality, increases in the time pressure will reduce the quality of practice.
- Quality of Product** Current quality of the product. Each week, the tasks completed correctly by the product and process engineers at the current quality of practice contribute to the overall quality of the product. The tasks completed contribute to the quality of the product as a percentage of the overall number of tasks in the project.
- Customer Perceived Quality** As soon as the product is released (time=scheduled release date), the customers obtain an initial perception of the quality. This perception of quality

changes as the quality of the product changes, but more slowly as time passes after the product release date.

Cumulative Known Rework

Total number of tasks requiring rework for both the product and the process engineers. Since there were 100 initial tasks for each group, the cumulative rework is also the percent of the total tasks that required rework. For example, if the product engineers had a cumulative known rework of 11.2 tasks at the completion of the project, it would indicate an average of 11.2% of tasks done incorrectly during the project.

3.1.6 Report 5. Time & Cost Forecast

Week # 0	
Projected Completion Date	
Original Target Date is week #	4 0
Your Scheduled Release Date is week #	4 0
Based on a PE required time estimate of	2 5
and a PcE required time estimate of	2 5
You are projected to finish the project in week #	2 5
Cost Estimate	
Your budget for this project is	\$ 1 7 6 0 . 0 (in thousands)
So far, you have spent:	
Product Engineering Base Costs -	\$. 0
PE Overtime Costs -	\$. 0
Process Engineering Base Costs -	\$. 0
PcE Overtime Costs -	\$. 0
Total:	\$. 0 (in thousands)
Based on your current total project team of 2 engineers, your total project cost are likely to be \$ 1 0 0 (in thousands), which is 6 % of the budget.	

Original Target Date

This is the targeted project completion date established in the beginning of the project. Releasing after the target date makes the product less competitive in the market place. (initial decision)

Scheduled Release Date

Week chosen to release the product. (weekly decision)

PE Required Time

Number of weeks to finish the product engineering

Estimate	tasks based on the current PE tasks completion rate and number of tasks remaining.
PcE Required Time Estimate	Number of weeks to finish the process engineering tasks based on the current PcE tasks completion rate and number of tasks remaining.
Projected Completion	The projected completion is the longer of the two team time requirements plus the current week number.
Project Budget	Total dollar amount allocated to the project. (initial decision)
Costs	The base costs are the cumulative team costs based on the 40 hour work week. Overtime costs are accounted separately. The current total project cost is also shown.
Total Project Team	Total number of engineers working on the project.
Projected Costs	Cost of employing the current process and product teams for the time required to finish the respective remaining tasks.
Projected % of Budget	Projected costs as a percent of the Project Budget.

3.1.7 Report 6. Project Summary

Week # 0		
Project Summary		
	<u>Product Engineering</u>	<u>Process Engineering</u>
Projected Completion Date: {Week #}	25	25
Estimated Productivity: {Tasks/Person-week}	.250	.250
Measured Productivity: {Tasks/Person-week}	.250	.000
Cumulative Known Rework: {Tasks}	.0	.0
Quality of Practice: {Competitor=1}	1.00	1.00
Percent of Project Completed:		0
Percent of Time Used:		0
Percent of Budget Consumed:		0
Product Quality		.00

- Projected Completion Date** Projected week of completion based on task completion rate and number of tasks remaining.

- Estimated Productivity** Estimated number of tasks completed per person per week. The productivity is influenced by the time pressure from the schedules, the engineer to tasks ratio (too many engineers can be ineffective), the amount of coordination, and the fraction of each team that are inexperienced.

- Measured Productivity** Measured by dividing the tasks completion rate by the respective team size. This will often be different from the estimated productivity because the measured productive takes into account losses in productivity from inexperienced engineers and time spend coordinating.

- Cum Known Rework** Total number of tasks requiring rework for each team. This is a significant end of project indicator because it shows the percent of the total tasks that had to be reworked, a sign of inefficiency.

Quality of Practice	Current quality being built into the product by each team.
Project Goals	Current status on the percent project complete, time used, budget consumed, and the quality of product.

3.1.8 Report 7. Sales and Marketing Report

Week # 0	
Sales & Marketing Report	
Unit Sales (units/week)	0
Installed Base (units)	0
Revenues (\$ thous./week)	\$ 0
Cumulative Profits (\$ thousands)	\$ 0
Customer Perceived Quality	1.00
Cumulative Production {units}	0

Unit Sales	Number of units sold each week. Sales begin when the scheduled release date is reached and the product is launched. Sales depend on the customer perceived quality of the product relative to the competitors. The competitors quality will improve slowly over time.
Installed Base	Number of units currently in use. There is loss due to wear.
Revenues	Each unit is assumed to sell for 12,000 dollars.
Cumulative Profits	Cumulative profits to date. There is assumed to be a learning curve to production that allows the cost to produce each unit to exponentially fall as the total number of units produced increases. This means the more units sold, the greater the profit on each unit.

Customer Perceived Quality	As soon as the product is released (time=scheduled release date), the customers obtain an initial perception of the quality. This perception of quality changes as the quality of the product changes, but more slowly as time passes after the product release date.
Cumulative Production	Total units produced to date.

3.2 Graphs & Tables

While weekly information on the progress of the project is available from the report pages, the behavior of the product development system over time is found in the graphs. The variables can also be displayed as tables by toggling the graph/ table bar in the microworld. Each graph displays several different variables. In the listing, the graph names are in bold while the variable names are in italics.

A Decision1

PE_Work_Week
PcE_Work_Week
Product_Engineers
Process_Engineers

A full description of each management decision can be found in the Management Decisions Section, page 5.

A Decision2

Coordination_Frac
Quality_Goal
Sheduled_PE_Finish
Sched_Release_Date

All Staffing

Total_Product_Engrs

Sum of the experienced and inexperienced product engineers.

Total_Process_Engrs

Sum of the inexperienced and experienced process engineers.

COP Team Costs

PE_Team_Costs

Total cumulative cost of the Product Engineer-weeks. Based on the number of Product engineers employed each week and the expense per engineer week. Includes overtime costs.

<i>PcE_Team_Costs</i>	Total cumulative cost of each Process Engineer-week
Cost of Project	
<i>Cum_Project_Cost</i>	Sum of the two Team Costs. The goal is to keep the cumulative project cost under or equal to the budget.
<i>Project_Budget</i>	The budget is calculated by finding number of engineer-weeks it would take to finish the project working at the base productivity rate and multiplying by the expense per engineer week. A 10% cushion is added to the estimate to make the budget. (Initial decision)
Cost of Overtime	
<i>PE_Overtime_Costs</i>	Cumulative overtime costs for the product engineering team.
<i>PcE_Overtime_Costs</i>	Cumulative overtime costs for the process engineering team.
Engineer Experience Mix	
<i>PcE_Rookie_Fract</i>	Fraction of Product Engineers that are inexperienced.
<i>PE_Rookie_Fract</i>	Fraction of Process Engineers that are inexperienced.
Process Engineer	
<i>Total_Process_Engrs</i>	The total process engineers is the sum of the inexperienced and the experienced engineers.
<i>Inexp_Process_Engrs</i>	Additions to the Process engineering team are assumed to be inexperienced. Inexperienced is defined as a fully trained engineer, but unfamiliar to the project. After a certain training period that is a function of the availability of experienced engineers for teaching and the coordination with the product engineers, the inexperienced engineer will become an experienced engineer. Inexperienced engineers are not as productive as experienced engineers.
<i>Exp_Process_Engrs</i>	The number of experienced Process Engineers.

Process Engineering Effort

<i>PcE_Tasks_Completed</i>	Current number of process engineers tasks remaining. There are 100 total tasks to complete over the course of the project.
<i>PcE_Tasks_Remaining</i>	Number of process engineering tasks remaining.
<i>Total_Disc_PcE_Rework</i>	A fraction of the process engineering tasks are found to be done incorrectly. Upon discovery, they are removed from the tasks completed stock and returned to the tasks remaining stock. The cumulative rework shows the current total number of tasks that were discovered to require rework. The fraction of the process engineering tasks incorrect is a function of the PcE quality of practice, their burnout, the percent of the product engineering tasks completed incorrectly (faulty designs sent over), the amount of the total product engineering tasks completed, and the coordination with the product engineers.

Process Engineering Turnover

<i>Inexp_ProcEngr_TO</i>	Number of inexperienced process engineers leaving the team. They will leave for voluntary reasons due to burnout and time pressure
<i>Exp_PcE_TO</i>	Experienced process engineers leaving the team.
<i>Avg_PcE_TO</i>	Average turnover of inexperienced and experienced engineers.

Product Engineer

<i>Total_Product_Engrs</i>	Sum of experienced and inexperienced product engineers.
<i>Inexp_Product_Engrs</i>	Engineers added through the product engineer decision are assumed to be inexperienced. Inexperienced is defined as a fully trained engineer, but one unfamiliar to the project. After a certain training period that is a function of the availability of experienced engineers for teaching, the inexperienced engineer will become an experienced engineer. Inexperienced engineers are not as productive as experienced engineers.

Exp_Product_Engrs Number of experienced product engineers.

Product Engineering Effort

PE_Tasks_Completed Number of product engineers tasks completed. There are a total of 100 product engineers tasks to be completed during the project.

PE_Tasks_Remaining Number of product engineers tasks remaining to be done.

Total_Disc_PE_Rework Total number of product engineering tasks that were discovered to require rework.

Product Engineering Turnover

Inexp_PE_TO Number of inexperienced product engineers leaving the team. They leave for voluntary reasons due to burnout and time pressure

Exp_PE_TO Experienced product engineers leaving the team.

Avg_PE_TO Average of experienced and inexperienced turnovers.

Productivity Estimated

Estimd_PE_Prod Estimated number of product engineering tasks competed per product engineer per week. The productivity is influenced by the time pressure from the scheduled finish date, the engineer to tasks ratio (too many engineers can be ineffective), the amount of coordination, and the fraction of each team that are inexperienced. This estimate does not account for the amount of time spent on each task and the time spent coordinating.

Estimd_PcE_Prod Estimated process engineering productivity. See explanation for product engineers.

Productivity Measured

Measured_Prodty_PE The measured product engineering productivity is the PE task completion rate divided by the total PE team size. This measure of productivity is much more accurate because it averages in the actual amount of time spent on each tasks and the time spent coordinating.

<i>Measured_Prodty_PcE</i>	The measured process engineering productivity is the PcE task completion rate divided by the total PcE team size.
Project Progress	
<i>PE Tasks Completed</i>	Number of PE tasks completed. Shown in the same graph as the PcE tasks completed to show the overall project progress and the relative product and process engineering phases of the project.
<i>PcE Tasks Completed</i>	Number of process engineering tasks completed.
Quality of Practice	
<i>Quality_Goal</i>	Desired final quality of the product. The competitor is assumed to have a quality of 1. A quality goal of 1.2 would indicate a desired final product quality 20% better than the competitor. (initial decision)
<i>PE_Quality_of_Practice</i>	Current product engineering quality of practice. The product engineers strive to work at the quality goal, but the quality will fall with increases in time pressure.
<i>PcE_Quality_of_Practice</i>	Current process engineering quality of practice.
Quality of Product	
<i>Quality_Goal</i>	Desired final quality of the product. There is a time delay between setting the quality goal and having that goal implemented in the engineer's quality of practice. (weekly decision)
<i>Product_Quality</i>	Cumulative quality of the product and process engineering tasks completed at their qualities of practice.
Rework Cumulative	
<i>Total_Disc_PE_Rework</i>	Cumulative product engineering tasks discovered to require rework.
<i>Total_Disc_PcE_Rework</i>	Cumulative process engineering tasks discovered to require rework.
Rework Discovery Rate	

<i>PE_Rework_Discovery</i>	A certain percentage of the tasks completed by the product engineers are incorrect and required rework. Each task discovered to require rework is taken out of the task completed pool and put back into the tasks remaining pool. The fraction of tasks done incorrectly is a function of the product engineering time pressure, their quality of practice, the amount of burnout from overwork, and the level of coordination with process engineering. The tasks requiring rework are not instantly known but rather are discovered each week as a function of how much of the project is completed and the level of coordination with process engineering.
<i>PcE_Rework_Discovery</i>	Process engineering tasks discovered to require rework each week.
Task Completion Rate	
<i>PE_Task_Completion_Rate</i>	Product engineering tasks completed each week is a function of the number of product engineers and their productivity, skill level, work week, and quality of practice.
<i>PcE_Task_Completion_Rate</i>	Process engineering tasks completed each week is a function of the number of product engineers and their productivity, skill level, work week, and quality of practice.
Work Week	
<i>PE_Work_Week</i>	Hours worked each week by the product engineering team. The work week directly affects the task completion rate.
<i>PE_Average_Work_Week</i>	The average work week for the product engineering team averaged over a month. When the average work week gets too high, burnout will set in leading to lower productivity and more errors.
<i>PcE_Work_Week</i>	Hours worked each week by the process engineering team.
<i>PcE_Average_Work_Week</i>	The average work week for the process engineering team averaged over a month.

Work Pressures*PE_Time_Pressure*

Product Engineering Time Pressure is the amount of pressure put on the engineers by the ratio of the estimated time required to finish the PE portion of the project to the time left in the PE schedule. As time pressure increases, productivity increases but eventually the quality of the work practice will decrease.

PcE_Time_Pressure

The Process Engineering Time pressure is similar to the Product Engineering Time Pressure except it is driven by the ratio of the estimated time required to time remaining in the scheduled release date.

The following graphs concern the sales/marketing results of the completed project. The project is "launched" when time is equal to the scheduled release date. This means that is the project is not completely finished at that time, the product will be released and sold without reaching its full quality.

z. Installed Base*Installed Base*

Number of units currently in the field.

z. Lead Time*LeadTime*

Number of weeks required to fill an order.

z. Orders*Orders*

Total number of unit ordered each week.

z. Perceived Quality*Actual_Quality*

Actual quality of the product.

*Customer_Percd
_Quality*

As soon as the product is released (time=scheduled release date), the customers obtain an initial perception of the quality. This perception of quality changes as the quality of the product changes, but more slowly as time passes after the product release date.

z. Revenues*Revenues*

Weekly revenues. Each unit is assumed to sell for 12,000.

Profits

Weekly profits. Revenues minus cost of production.

z. Revenues Cumulative

Cum_Revenues

Cumulated revenues for total sales.

Cum_Profits

Cumulated profits on total sales to date.