# *Cost Reduction Methodology and Management*

*by*
*Gary Tarpinian*

*B. S., Electrical and Computer Engineering*
*University of Wisconsin-Madison, 1990*

*Submitted to the Sloan School of Management and*
*the Department of Electrical Engineering and Computer Science*
*in Partial Fulfillment of the Requirements for the Degrees of*

*Master of Science in Management*
*And*
*Master of Science*

*at the*
*Massachusetts Institute of Technology*
*June 1999*

Signature of Author ______________________________
Sloan School of Management
Department of Electrical Engineering and Computer Science

Certified by ______________________________
Professor Elizabeth Eccher, Thesis Advisor
Sloan School of Management

Certified by Alvin W Drake
Professor Alvin Drake, Thesis Advisor
Department of Electrical Engineering and Computer Science

Accepted by ______________________________
Art Smith, Chairman
EECS Department Committee on Graduate Students

Accepted by ______________________________
Larry Abeln, MBA Program Director
Sloan School of Management

[redacted]

# Cost Reduction Methodology and Management

by
Gary Tarpinian

Submitted to the Sloan School of Management and
the Department of Electrical Engineering and Computer Science
in Partial Fulfillment of the Requirements for the Degrees of

Master of Science in Management
And
Master of Science

at the
Massachusetts Institute of Technology
June 1999

## ABSTRACT

Intel Corporation recognized the ability to perform continuous cost reduction was necessary to maintain competitiveness in manufacturing operations. Intel was concerned with not only reducing the variable costs of producing semiconductors, but also in doing so quickly in order to maximize profitability for products of limited life-cycle.

This thesis is based upon work that took place in 1998 at an Intel wafer fabrication facility near the corporate headquarters Santa Clara, CA. The objective of this thesis is to describe a cost reduction methodology used within Intel factories to achieve reductions in wafer manufacturing costs. Organization, information, project definition, and tools characterize the methodology. Cost reduction teams were organized around the key cash cost drivers in the factory: spares, gas/chemicals, test wafers, and labor. Information was used to continuously monitor spending, measure improvements, and benchmark against the virtual factory to identify opportunities for improvement. Cost reduction projects were identified, characterized, prioritized and tracked with the help of cost reduction management tools developed at Intel.

This work included interviews with over thirty participants in Intel's cost reduction effort at Santa Clara and other manufacturing sites. These interviews identified a need for an improved method for managing cost reduction projects. Feedback from the various stakeholders led to the development of a comprehensive project database that incorporated best practices from a variety of sources from within Intel as well as financial and operational analysis techniques from outside the organization. The project database allowed factory managers and technicians to perform economic analysis of an individual project, aggregate the results of a portfolio of projects, forecast manufacturing costs, and evaluate team and factory performance with respect to cost reduction. The research also addresses the role of risk management and organizational learning in a continuous cost improvement strategy.

## ACKNOWLEDGMENTS

The research presented in this thesis was performed under the auspices of the MIT Leaders for Manufacturing Program, a partnership between MIT and a number of major U.S. manufacturing companies. I would like to thank MIT and the Leaders For Manufacturing program for its support of this work and the opportunity to learn and grow as an individual over the past two years.

Special thanks go to Intel Corporation for its support of my internship and this paper. Many people played an important role in making my experience at the D2 facility in Santa Clara, CA a very rewarding one. In particular, two key individuals provided much appreciated time, support, and encouragement during my internship: Melinda Manente (LFM '95) and Chris Keith (LFM '96).

Special thanks go to my advisors, Elizabeth Eccher and Alvin Drake, for their support during my internship and the development of this paper.

I would also like to extend my appreciation to my classmates at LFM who contributed to my education over the past two years. I have learned from your ideas, values, and commitment. In particular, I would like to thank my fellow Intel LFM Interns (ILIs), Shafali Rastogi, Manish Bhatia, and Ajay Amar, who helped me to shape this work and make my internship a success. You all played an important part in making my LFM experience a very positive and fulfilling one.

Finally, I thank my wife Donna and my children Kelsey and Matthew for indulging me with your support and patience these last two years. I solemnly promise to refrain from further educational pursuits for at least two years.

## Table Of Contents

## Table Of Figures

# 1. Introduction

## 1.1. Chapter Description

The purpose of this thesis is two-fold. The first is to describe in detail tools and a methodology for identifying, prioritizing, and executing cost reduction projects in a manufacturing setting. The second is to present a framework for developing such a system so that it will appropriately fit the organization for which it is intended. This chapter includes an assessment of the manufacturing-business climate at Intel to provide the necessary context for the rest of the document. The cost reduction methodology described in Chapter 2 consists of the organization, information systems, processes, and tools used to conduct cost reduction at Intel. The processes used to develop a new cost management tool for Intel are described in Chapters 3-8. The development process is broken into five phases: data collection, definition of functional requirements, prototype development, pilot program, and implementation. For each phase of development, this thesis provides instructions and guidelines followed by specific examples and lessons learned from implementation of a system at an Intel semiconductor manufacturing facility[1].

## 1.2. Manufacturing-Business Climate at Intel in 1998

Intel, the undisputed world leader in microprocessor sales, was encountering a crisis in 1998, during what many would consider the worst downturn ever in the semiconductor industry. The market had softened for Intel's highest performing chips, and the competition had seriously eroded Intel's share of the low-end PC processor business.

Intel responded to the market changes by segmenting the microprocessor market into three performance levels (Celeron, Pentium, and Xeon), and by aggressively pursuing cost reduction in operations. The cost reduction mandate was passed down to the factories from the highest levels of management, and it took little time to trickle down to the factory floor.

### 1.2.1. Capital Intensive Industry

The semiconductor manufacturing industry was somewhat unique in that it was so overwhelmingly capital intensive. As an industry, depreciation of equipment and facilities were two of the greatest expenses. Factories typically cost more $1 Billion and had a useful life of two to four years. Once the factory was built, little could be done to reduce the rate at which the property depreciated.

[1] The data for all examples of cost reduction projects at Intel are fabricated to protect proprietary information.

Depreciation was driven by obsolescence, so the equipment depreciated at the same rate whether it was 80% utilized or 40% utilized. Another significant operating cost driver was labor. Factory technicians were highly skilled and highly paid. For every technician in the factory, there was an engineer or manager outside the fab working to maintain production, cycle time and Work In Process (WIP) turn goals. While labor costs are theoretically variable with production level, in practice it was difficult to quickly adjust headcount. Technicians required a great deal of training before they were productive, and equipment maintenance demanded technician support around the clock.

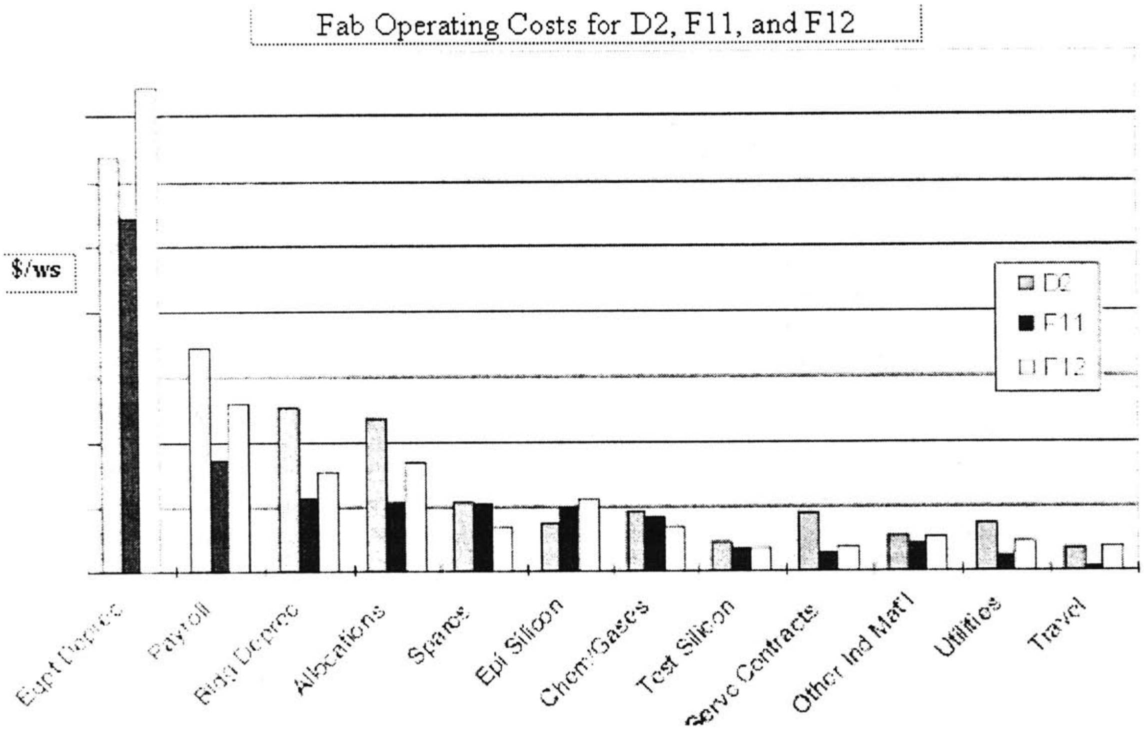


**Figure 1 – Cost Comparison of Three Intel factories**

The next largest costs were the consumables, or variable costs associated with manufacturing semiconductors. Variable costs included the raw silicon wafers used to make product, chemicals, gases, spare parts, and test wafers. The focus of the cost reduction efforts at Intel was on labor and these consumables, since they were largely within the control of the factory management. Figure 1 shows the costs that go into making a wafer, ordered by magnitude in a Pareto diagram.

### 1.2.2. Volume and Wafer Costs

The other variable that greatly affected wafer cost was production levels. A factory manager could significantly reduce his or her average wafer cost by increasing the wafer starts per week. Since much of the operating costs were fixed in the short term (depreciation and labor) increasing production levels spread the fixed costs out over more wafers and therefore lowered the average cost per wafer. This strategy could have lead to inefficiencies for the overall company if total production was not carefully managed. Had all the factories together overproduced, Intel would have exceeded market demand and created excess inventory. Ideally, each factory produced at a level where the marginal cost per wafer was the same as the other factories.

### 1.2.3. Copy EXACTLY! And the Virtual Factory

Two terms coined within Intel are necessary to describe their operations strategy. The first is *copy EXACTLY!*. Copy EXACTLY! refers to the practice used by Intel to copy manufacturing operations from one factory to another. Stated in its simplest form, "everything which might affect the process, or how it is run" is to be copied down to the finest detail, unless it is either physically impossible to do so, or there is an overwhelming competitive benefit to introducing a change. The second term is *virtual factory*. The virtual factory is the set of Intel factories tooled to produce the same products. As the name suggests, the virtual factory operated as a single entity with respect to manufacturing processes, capacity planning, engineering, and purchasing. For the purposes of this thesis, the virtual factory was made up of the following manufacturing sites:

- D2 in Santa Clara, CA
- F11 in Rio Rancho, NM
- F12 in Chandler, AZ
- F15 in Portland, OR
- IFO in Ireland

All members of the virtual factory produced 8-inch wafers containing logic or memory chips using one of several possible manufacturing processes.

## 1.3. Mission of the D2 Facility in Santa Clara, CA

D2 was one of two development factories operated by Intel. The development factory was responsible for developing the manufacturing process[2] by which a new generation of semiconductor products was manufactured. When the process at D2 reached a level of stability and yields were acceptable, the process

[2] A process is the combination of equipment and procedures that make up a recipe for manufacturing wafers.

was ported over to the high volume factories within the virtual factory. Engineers from the high volume factories temporarily relocated to D2 to learn the process and facilitate a smooth transfer back to their home fab.

In addition to operating as a development factory, D2 also produced products that were sold for revenue. Production levels were maintained to allow ample capacity for process development, but still utilize the equipment at reasonable levels. This ensured that the processes were capable within a production environment and not just in a laboratory setting. Production also subsidized the development mission by generating revenue.

# 2. Cost Reduction Methodology at Intel

This chapter describes the organization, information, and tools that make up the cost reduction methodology at Intel. An example of a cost reduction project from conception to execution is provided as a means to demonstrate how the methodology and tools were used in practice.

## 2.1. Organization

D2 had organized several cost teams made up of management, engineers, tool owners, and representatives from finance and purchasing. The purpose of the teams was to identify opportunities to reduce manufacturing costs, execute projects to achieve cost savings, and report results to management. The teams were organized around the following commodities:

- Spares: Spares are spare parts consumed by the manufacturing equipment in the factory. The majority of the spares were consumed as a result of planned maintenance. The remainder was spares used because of unplanned failure of parts. The planned maintenance work was scheduled according to a time-based or a wafer-based schedule.
- Chemicals and Gases: The semiconductor manufacturing process involves the consumption of various chemicals and gases. Consumption of chemicals and gases was approximately linear with production level.
- Test wafers: Test wafers are silicon wafers consumed in the manufacturing process to calibrate and confirm proper equipment operation. In some manufacturing processes, the ratio of test wafers to product wafers approached or exceeded one.
- Labor: Technicians and engineers working in and supporting the fab were classified as a commodity for cost measurement purposes.

These four commodities made up the majority of the cash costs of operating the D2 semiconductor fab. The commodity cost reduction team structure was duplicated throughout the virtual factory, and representatives from like teams communicated periodically.

In addition to the cost teams, D2 established a higher-level cost reduction forum made up of senior factory management and the leaders from each team. The forum was known as "Cost Ops", and it met once a week to discuss recent spending trends, review ongoing cost reduction efforts, and evaluate new initiatives.

## 2.2. Information

The cost teams had ample and timely data from which they would analyze detailed spending profiles for D2 as well as the virtual factory. Typically, the commodity team would break spending down by tool and by the specific commodity purchased. Information was typically presented to the team in Pareto style charts to focus attention on the highest costs. Figure 2 is an example of a typical Intel cost Pareto. Attention would be focused on Tools C and F first, since they represent the highest costs.

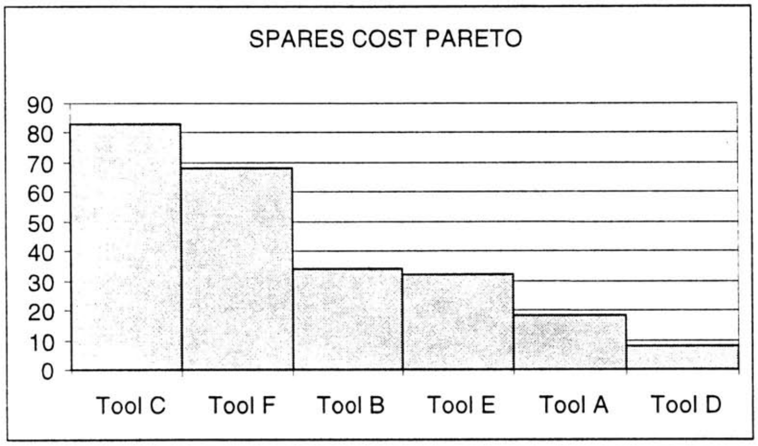


**Figure 2 - Tool Cost Pareto for Spare Parts**

Cost data for each fab were often used for benchmarking purposes. Cost was often presented normalized to $/wafer start to adjust for differences in production levels. Figure 3 is an example of a Pareto diagram for the spare parts used by Tool C. The spares team would focus its attention on the PM kit since D2's costs were considerably higher than either Fab 11 or Fab 12. Attention would also be focused on the holder, since D2's costs were lower than average. The virtual factory competed but also cooperated to help each other reach the lowest costs possible.

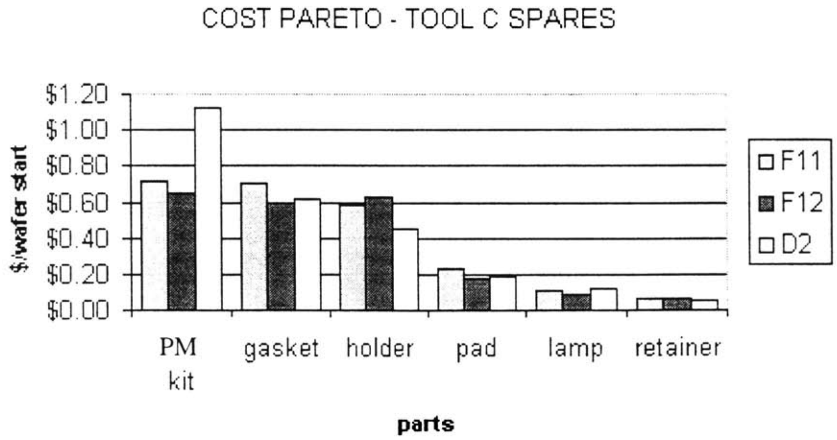


**Figure 3 - Pareto of Parts for Tool C**

## 2.3. Project definition

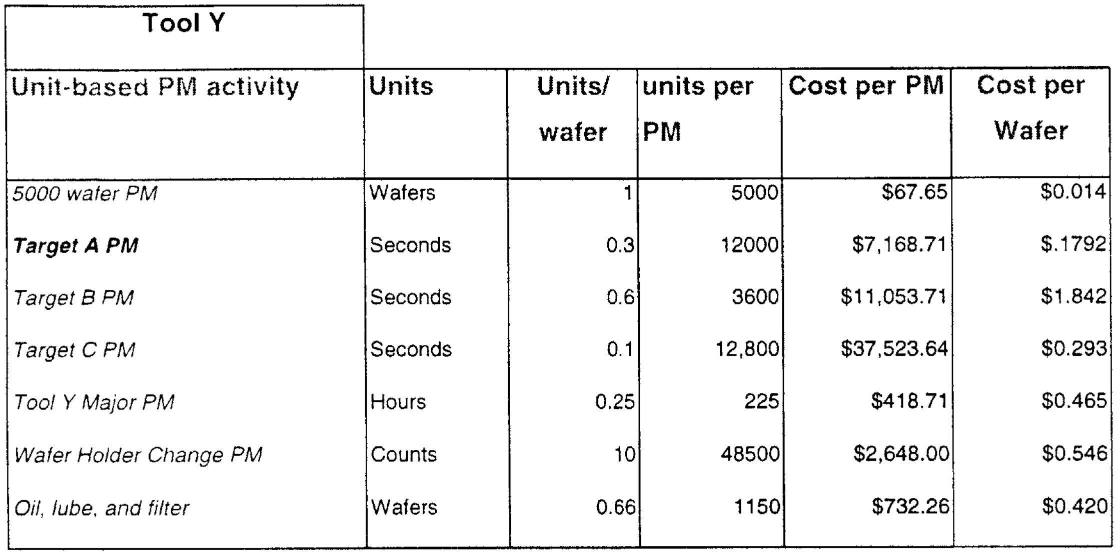
Using information from Pareto diagrams to focus their efforts, teams would then brainstorm to come up with cost reduction projects. When a potential project was identified, there was a formal process of review by the virtual factory to assess the risks and benefits of the project. Usually one factory would take the lead on the project and collect experimental data. The data would be reviewed by the virtual factory joint engineering team and the project either approved or disapproved. If approved, the project would be executed throughout the virtual factory, consistent with the Copy EXACTLY! Strategy.

## 2.4. Tools

To aid in the characterization, assessment, and tracking of projects, Intel developed several tools.

### 2.4.1. Bottoms-Up Estimate

The Bottoms-Up-Estimate (BUE) was a cost model that captured the cash costs associated with operating a fabrication tool, taking into consideration factory load, product mix, planned maintenance schedules and frequency, parts cost, and a parts list for each maintenance activity. Based on this information, tool owners developed detailed cost models that could then be used to create "should cost" estimates of spare parts consumption. Typically, the models were created using multi-page spreadsheet models and required detailed understanding of how the tool operated.



| Tool Y | | | | | |
|---|---|---|---|---|---|
| Unit-based PM activity | Units | Units/ wafer | units per PM | Cost per PM | Cost per Wafer |
| *5000 wafer PM* | Wafers | 1 | 5000 | $67.65 | $0.014 |
| ***Target A PM*** | Seconds | 0.3 | 12000 | $7,168.71 | $.1792 |
| *Target B PM* | Seconds | 0.6 | 3600 | $11,053.71 | $1.842 |
| *Target C PM* | Seconds | 0.1 | 12,800 | $37,523.64 | $0.293 |
| *Tool Y Major PM* | Hours | 0.25 | 225 | $418.71 | $0.465 |
| *Wafer Holder Change PM* | Counts | 10 | 48500 | $2,648.00 | $0.546 |
| *Oil, lube, and filter* | Wafers | 0.66 | 1150 | $732.26 | $0.420 |

**Figure 4 – Excerpt from the Bottoms-Up Estimate for Spares**

Figure 4 is an example from a BUE developed for a wafer-processing tool. In the figure, there are six different unit-based planned maintenance activities. The second PM (highlighted in bold) replaces a Target A after every 12000 seconds of use. Each wafer consumes 0.3 seconds of target life. Each Target A PM has a cost of $7168, resulting in a cost of $0.1792/wafer. This same calculation is performed for all PMs for this tool. The result is an estimate of the spares cost on a per/wafer basis, assuming that spares are consumed only from scheduled maintenance and that all scheduled maintenance occurs exactly when it is required. In reality, Non-PM spares made up a significant portion of the spares costs, and maintenance was performed before or after the exact PM scheduled time.

D2 realized the following benefits from developing the BUE:

- Collecting the needed cost data helped tool owners identify the largest spares cost drivers.
- Discrepancies between the predicted and actual consumption could be investigated.
- Tool owners could perform "what if" analyses to determine how changes to the PM schedule would influence spares spending.

The BUEs helped justify efforts to convert many time-based PMs to wafer-based. In other words, instead of replacing a part every 30 days, it might be replaced after some number of wafers are processed. This would tend to reduce spares consumption in an under-utilized factory as well as rationalize the useful life of the part.

### 2.4.2. The Cost Reduction Project Database

A second tool Intel used to manage and track cost reduction projects was the Cost Reduction Project Database. The project database allowed management and tool owners a way to keep track of all the cost reduction projects in D2 and the virtual factory. The project database provided a means to capture all relevant information about a project, keep track of the project status, generate reports and graphs, and aggregate the results.

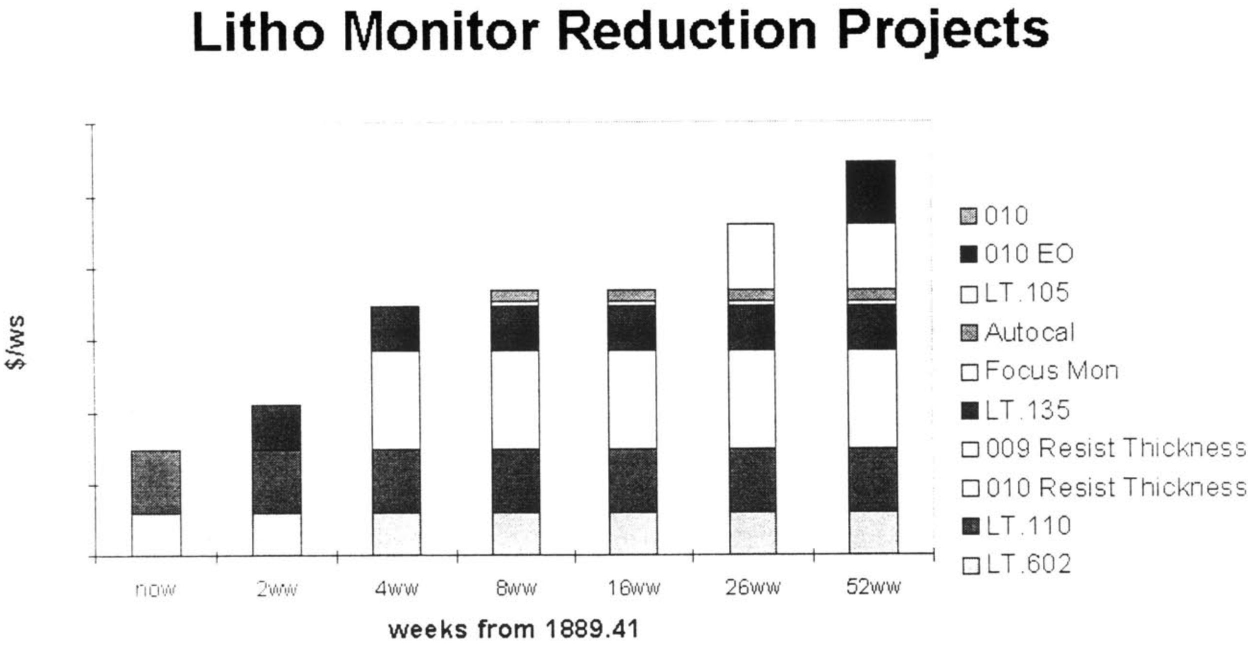


**Figure 5 - Sample Chart from Project Database**

The database captured information about the purpose of the project, the functional area (e.g., lithography), the specific tool affected, estimated completion dates, estimated savings, and impact on yield and cycle time. The database included tools to help calculate savings estimates and risks. Wherever possible, the database helped the user to use a common set of assumptions about factory production levels, product mix, and commodity costs. Figure 5 shows a sample chart from the project database. In the example, there are ten different projects each with unique estimated savings and completion dates. This kind of chart would help tool owners prioritize their efforts and forecast savings into the future based on expected gains from the planned projects.

The project database provided Intel the following benefits:

- Documentation for hundreds of cost reduction projects across functional areas and throughout the virtual factory.
- Accurate and complete descriptions of each project, including savings estimates.
- The ability to aggregate the results of a portfolio of projects.
- Forecasting capability to help predict future manufacturing costs.

The database was developed using best practices from within Intel and results of interviews with project stakeholders. The process of developing this project database is the focus of Chapters 3-7 of this document. A complete list of functional requirements can be found in Chapter 4.

## 2.5. Relationship between the BUE and Project Database

Figure 6 illustrates the relationship between the BUE and the project database. The two tools complimented each other. The BUE captured the estimated spending for an entire tool area. The project database helped manage the process of reducing spending.

For a given project, the BUE could be used to help estimate the savings from the project. By using the BUE as a cost model and performing "what-if" analysis, tool owners were able to quickly estimate the total savings of a cost reduction project by comparing before and after costs. The savings estimates could then be entered into the project database, which could be used to calculate the total benefit of the project over time and prioritize the project among other candidate projects.

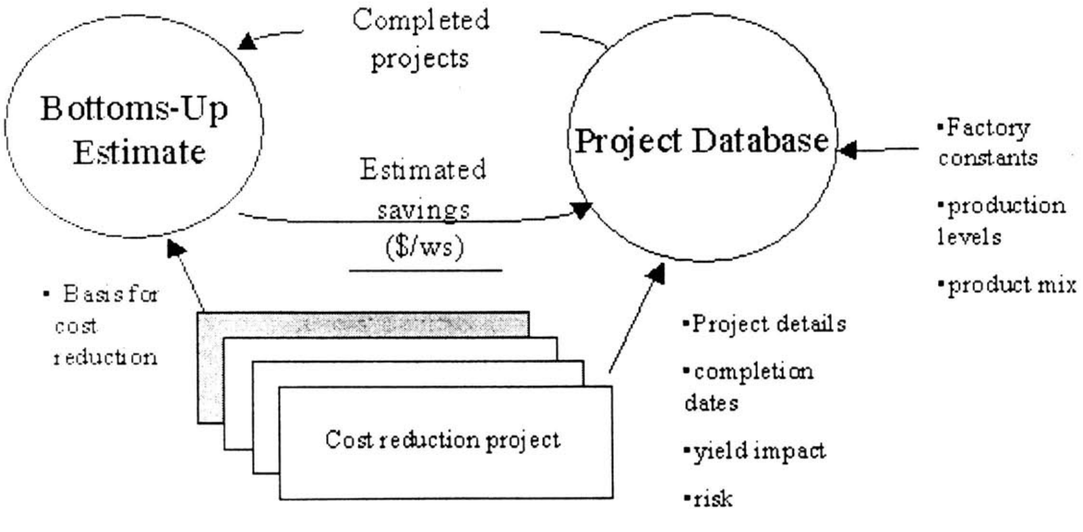


**Figure 6 - Relationship between BUE and Project Database**

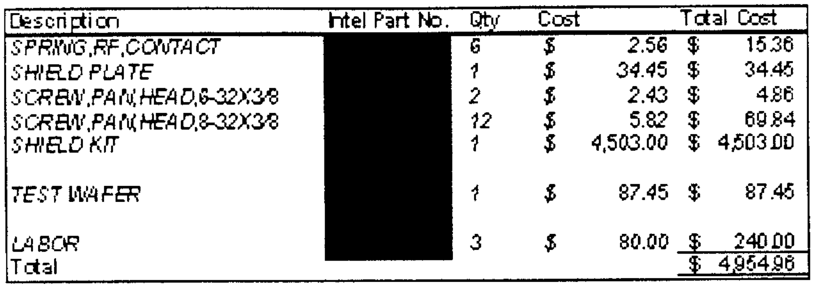
## 2.6. Cost management at Intel – A Example

The following example takes a project from conception through completion using the Intel cost reduction methodology used by the commodity teams, including use of the tool described above. The example takes place over the course of several weeks. The example project, factory loading assumptions, and all associated cost data are fictitious.

Week 1: At the weekly spares cost reduction team meeting, the team leader announces that according to the latest cost data, the spares spending at D2 was the highest in the virtual factory and 50% higher than the best factory (see Figure 1 on page 7). D2 has been tasked with reducing spares spending to match the virtual factory average. Figure 2 on page 11 shows spares spending for the top six tools, indicating that tools C and F have the highest spares spending. The owners of these two tools are asked to seek out opportunities to reduce spares costs for these tools.

Week 2: The owner of Tool C opens up the meeting by presenting spares consumption for the past six months, normalized by the number of wafers processed at each factory (Figure 3 on page 12). Based on her analysis, she has found that D2 is spending about $ 0.40 more per wafer than F11 or F12 for the PM kit. The PM kit is replaced by a monthly planned maintenance (PM) activity, and is the single most expensive part consumed. She has looked into why costs are higher at D2 by comparing notes with her counterparts at the other factories. It turns out that while all three factories are spending the same for the kits and replacing them monthly, D2 has lower utilization on tool C than either F11 or F12. Therefore, the cost *per wafer* is higher at D2. The team leader suggests that she look into why the PM is performed every month, and if it could be converted into a wafer-based PM instead of a time-based PM. For example, if all factories replaced the PM kit every 20,000 wafer passes instead of every month, then the per wafer cost would be the same for all factories. The tool owner agrees to look into the basis for the PM and to investigate the potential cost savings.

Week 3: The owner of Tool C has learned from the vendor that the recommended monthly replacement of the PM kit is driven by degradation of the shield as a function of wafers processed, and the one-month periodic was established by assuming 80% tool utilization. She recommends that the PM be converted to a wafer-based PM. Based on her calculations and conversations with the vendor about how the life of the PM kit could be extended, she recommends the PM be performed every 24,000 wafer passes.



| Description | Intel Part No. | Qty | Cost | Total Cost |
|---|---|---|---|---|
| SPRING,RF,CONTACT | [redacted] | 6 | $ 2.56 | $ 15.36 |
| SHIELD PLATE | [redacted] | 1 | $ 34.45 | $ 34.45 |
| SCREW,PAN,HEAD,6-32X3/8 | [redacted] | 2 | $ 2.43 | $ 4.86 |
| SCREW,PAN,HEAD,8-32X3/8 | [redacted] | 12 | $ 5.82 | $ 69.84 |
| SHIELD KIT | [redacted] | 1 | $ 4,503.00 | $ 4,503.00 |
| TEST WAFER | [redacted] | 1 | $ 87.45 | $ 87.45 |
| LABOR | [redacted] | 3 | $ 80.00 | $ 240.00 |
| Total | | | | $ 4,954.96 |

**Figure 7 – Costs for the Tool C PM kit PM**

She also presents data from her BUE to estimate the potential savings from extending the PM (see Figure 7). In addition to the $4,503 for the PM kit, there are additional parts, labor and a test wafer to re-qualify the tool after the PM. The total cost is $4,954.96 each time the PM is performed.

| PM | Cost | passes/PM | cost/pass | process A passes/wafer | cost/ Process A wafer | process B passes/wafer | cost/ Process B wafer |
|---|---|---|---|---|---|---|---|
| PM Kit | $ 4,954.96 | 24,000 | $ 0.21 | 3 | $ 0.62 | 5 | $ 1.03 |

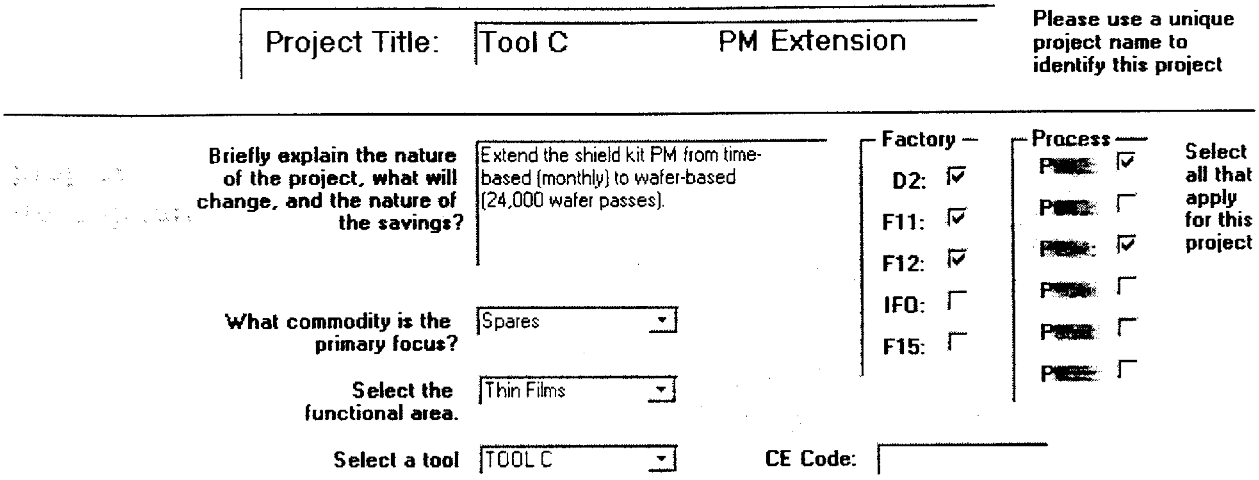
**Figure 8 – BUE $/wafer start Cost Estimate for PM kit PM**

Each PM should be good for 24,000 wafer passes, and each wafer passes through Tool C several times. Therefore, the cost per Process A and Process B[3] wafer is $0.62 and $1.03, respectively. Figure 8 shows the variables for determining the wafers costs. Based on the current process mix and factory loadings, extending the PM should reduce spares costs for Tool C by about $0.50 per wafer.

The team leader and the tool owner decide to pursue changing the PM kit PM frequency. This will require writing a white paper proposal to justify the change to the virtual factory, testing the change during a pilot phase, data evaluation, and virtual factory approval. The project is now ready to be entered into the Cost Reduction Project Database.

Week 4: The owner of Tool C has entered the preliminary project data into the project database.

[3] Process A and Process B are different recipes used to produce semiconductors at D2.



Project Title: Tool C PM Extension

Please use a unique project name to identify this project

Briefly explain the nature of the project, what will change, and the nature of the savings? Extend the shield kit PM from time-based (monthly) to wafer-based (24,000 wafer passes).

What commodity is the primary focus? Spares

Select the functional area. Thin Films

Select a tool TOOL C

CE Code:

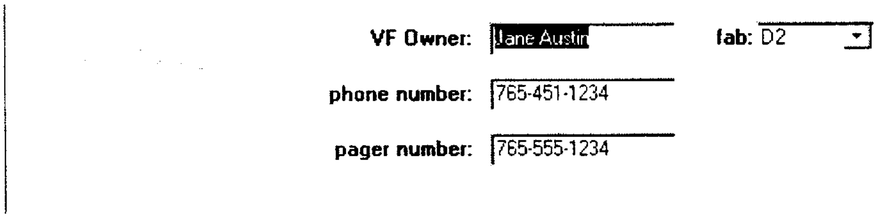
Factory
D2: ☑
F11: ☑
F12: ☑
IFO: □
F15: □

Process
[redacted] ☑
[redacted] □
[redacted]: ☑
[redacted] □
[redacted] □
[redacted] □

Select all that apply for this project

## Figure 9 - Project Description Screen

Figure 9 shows the project description screen, where the tool owner enters the name of the project, a brief description of what the project does, the factories and processes to which the project applies, tool group, tool, and commodity.



VF Owner: Jane Austin

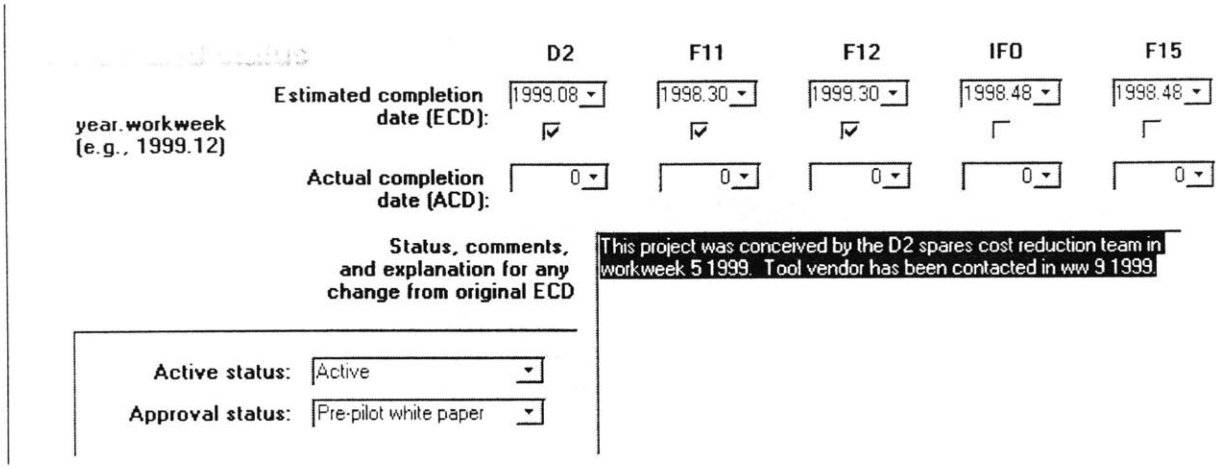
fab: D2

phone number: 765-451-1234

pager number: 765-555-1234

## Figure 10 - Ownership Screen

Figure 10 shows the ownership screen where the tool owner can enter her name and contact information. She also enters her home factory, establishing herself and D2 as the lead person and factory for this project.



year.workweek (e.g., 1999.12)

| | D2 | F11 | F12 | IFO | F15 |
|---|---|---|---|---|---|
| Estimated completion date (ECD): | 1999.08 | 1998.30 | 1999.30 | 1998.48 | 1998.48 |
| | ☑ | ☑ | ☑ | ☐ | ☐ |
| Actual completion date (ACD): | 0 | 0 | 0 | 0 | 0 |

Status, comments, and explanation for any change from original ECD: This project was conceived by the D2 spares cost reduction team in workweek 5 1999. Tool vendor has been contacted in ww 9 1999.

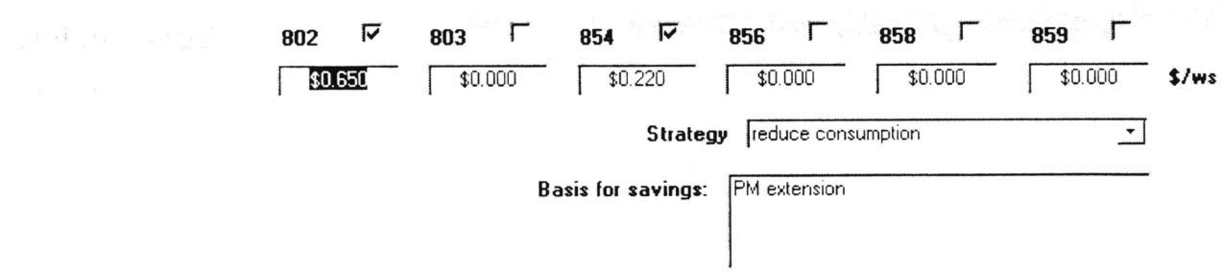
Active status: Active

Approval status: Pre-pilot white paper

**Figure 11 - Dates and Status Screen**

Figure 11 shows the dates and status screen. Each factory has two dates associated with each project: an estimated completion date (ECD) and an actual completion date (ACD). ECDs were established when the project was initiated, and ACDs were noted when the project is completed. All dates in the database use the year.workweek format. The status menus helped further classify the projects. The active status differentiated projects that were currently in process from those that had been completed or dropped. The approval status stated at what stage of the virtual factory approval process the project is in. The large text box is an area to type anything related to the project that others might need to know.

The tool owner has entered workweek 1999.08[4] as the estimated workweek for D2, and workweek 1999.30 for F11 and F12. Since IFO and F15 do not use Tool C, the project does not apply to them.



| 802 ☑ | 803 ☐ | 854 ☑ | 856 ☐ | 858 ☐ | 859 ☐ | |
|---|---|---|---|---|---|---|
| $0.650 | $0.000 | $0.220 | $0.000 | $0.000 | $0.000 | $/ws |

Strategy: reduce consumption

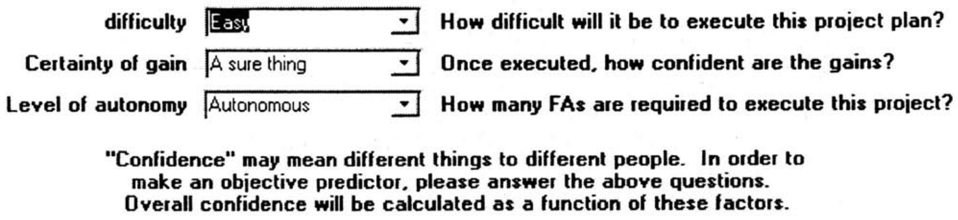
Basis for savings: PM extension

**Figure 12 - Savings Screen**

Figure 12 shows the savings screen where the tool owner enters in the expected savings from the project. Referring back to Figure 8 where she calculated the casts before and after the project, she estimates the

[4] Intel measures time in workweeks. 1999.30 means the 30th week of 1999.

savings to be $0.65 for process X wafers and $0.22 for P854 wafers. She also captures the basis and strategy for capturing the savings: reduce consumption of spares through PM extension.



difficulty: Easy — How difficult will it be to execute this project plan?

Certainty of gain: A sure thing — Once executed, how confident are the gains?

Level of autonomy: Autonomous — How many FAs are required to execute this project?

"Confidence" may mean different things to different people. In order to make an objective predictor, please answer the above questions. Overall confidence will be calculated as a function of these factors.

**Figure 13 - Confidence Screen**

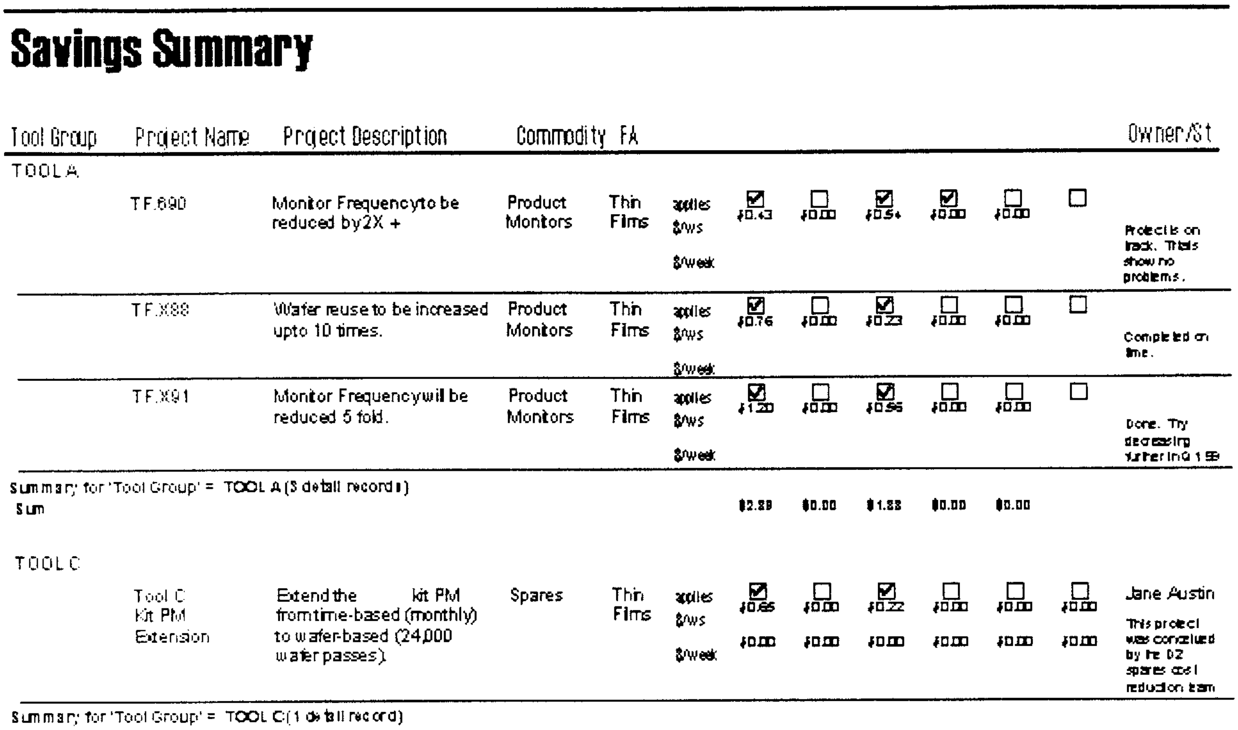
Figure 13 shows the confidence screen. This screen captures the certainty or uncertainty of the projects success using three different metrics: difficulty, certainty of gain, and level of autonomy. Difficulty captures how hard is to do what you say you are going to do. Certainty of gain means even if you execute the project as planned, how certain is the gain. Level of autonomy asks how many different stakeholders must be involved to complete the project. Based on these three metrics, the project database estimates the success of the project and discounts the expected value accordingly. For example, a project that is expected to save $200,000 with a confidence of 75% would be valued at $150,000.

Now that the project has been captured in the database, it will appear with other projects in various reports used by management, as well as her own reports for tracking projects in her area.

Week 6: The manager of the Thin Films area requests a summary of all projects currently being worked on in Thin Films. Figure 14 (next page) shows an excerpt from the savings summary report for thin films, including the Tool C PM kit project.

Week 8: Management is pleased with the results of the project and asks whether the PM could be extended further, to 36,000 wafer passes. The spares calculator shown in Figure 21 can be used to calculate the expected savings, assuming the entire cost of the PM, to be about $0.21/ws for process X wafers. However, the tool vendor warns that beyond 24,000 passes, die yield may be degraded. The yield impact can also be reduced to a $/ws, and weighed against the benefit of the lower PM cost.

Week 9: The engineering committee reviewing the project approves implementation for the entire virtual factory. The database and the BUE are updated to reflect the actual completion dates, status, and costs.



## Savings Summary

| Tool Group | Project Name | Project Description | Commodity | FA | | | | | | | | Owner/St |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| TOOL A | | | | | | | | | | | | |
| | TF.690 | Monitor Frequencyto be reduced by2X + | Product Monitors | Thin Films | applies<br>$/ws<br>$/week | ☑<br>$0.43 | □<br>$0.00 | ☑<br>$0.54 | ☑<br>$0.00 | □<br>$0.00 | □ | Project is on track. Trials show no problems. |
| | TF.X88 | Wafer reuse to be increased upto 10 times. | Product Monitors | Thin Films | applies<br>$/ws<br>$/week | ☑<br>$0.76 | □<br>$0.00 | ☑<br>$0.23 | □<br>$0.00 | □<br>$0.00 | □ | Completed on time. |
| | TF.X91 | Monitor Frequencywill be reduced 5 fold. | Product Monitors | Thin Films | applies<br>$/ws<br>$/week | ☑<br>$1.20 | □<br>$0.00 | ☑<br>$0.56 | □<br>$0.00 | □<br>$0.00 | □ | Done. Try decreasing further in Q1 99 |
| Summary for 'Tool Group' = TOOL A (3 detail records) | | | | | | | | | | | | |
| Sum | | | | | | $2.39 | $0.00 | $1.33 | $0.00 | $0.00 | | |
| TOOL C | | | | | | | | | | | | |
| | Tool C Kit PM Extension | Extend the kit PM from time-based (monthly) to wafer-based (24,000 wafer passes). | Spares | Thin Films | applies<br>$/ws<br>$/week | ☑<br>$0.65<br>$0.00 | □<br>$0.00<br>$0.00 | ☑<br>$0.22<br>$0.00 | □<br>$0.00<br>$0.00 | □<br>$0.00<br>$0.00 | □<br>$0.00<br>$0.00 | Jane Austin<br>This project was concluded by the 02 spares cost reduction team |
| Summary for 'Tool Group' = TOOL C (1 detail record) | | | | | | | | | | | | |

**Figure 14 - Savings Summary Report**

## 2.7. Chapter Summary

This chapter describes the methodology used within Intel factories to achieve reductions in wafer manufacturing costs. Organization, information, project definition, and tools characterize the methodology. Cost reduction teams were organized around the key cash cost drivers in the factory: spares, gas/chemicals, test wafers, and labor. Information was used to continuously monitor spending, measure improvements, and benchmark against the virtual factory to identify opportunities for improvement. Cost reduction projects were identified, characterized, prioritized and tracked with the help of two cost reduction management tools developed at Intel. The first tool was the bottoms up estimate, a cost model that reflects all the planned costs of operating a tool in the factory. The second tool was the project database, a system for characterizing and tracking cost reduction projects from conception to completion.

# 3. Data Collection

## 3.1. Chapter Overview

The cost reduction culture at D2 was studied as part of an overall process to develop a formal methodology for sustained cost competitiveness. The development process included data collection, project definition, prototype development, pilot program, and implementation. This chapter describes a two-stage data collection process used to develop an understanding of the strengths and weaknesses of the cost reduction environment at Intel. The first stage included a period of informal, passive data collection to get to know the organization. Informal data collection took the form of attending cost team meetings and getting involved in small projects. The second stage was a period of more active collection including formal interviews to document how engineers and managers performed cost reduction and what could be done to help them. Voice-of-the-Customer (VOC) interview techniques were used to collect customer needs. The VOC data was then organized in a useful format using the Strengths-Weaknesses-Opportunities-Threats framework. The VOC interviews also provided an opportunity to collect information on existing cost reduction project tracking tools. This chapter summarizes these tracking tools by commodity team and describes the strengths and weaknesses of each. The VOC data and the analysis of existing tools are organized for use in creating functional requirements in the next chapter.

## 3.2. Participating on Teams

During the passive phase, it was possible to learn a great deal about the organization by attending daily operations meetings, weekly factory cost meetings, and commodity cost team meetings. Each commodity group (spares, chems/gases, monitors/test wafers) met regularly to discuss the status of cost projects actively being pursued in various parts of the factory. These meetings provided many opportunities to see how cost reduction was being managed at the factory floor level.

## 3.3. Voice of the Customer Interviews

The next phase of data collection required a more comprehensive understanding of the needs of the cost reduction effort at D2. A useful technique for collecting information on existing cost reduction efforts was to conduct Voice-of-the-Customer interviews with many key players in the cost reduction effort. The Cost Ops forum was used as a starting point for identifying interview candidates. Figure 15 is an initial list of representatives from various functions and titles of Cost Ops participants targeted for interviews.

| Interviewees | |
|---|---|
| D2 and F17 Operations Manager | Product marketing |
| Finance Manager | D2 purchasing |
| Test Wafer Reduction Team | Finance Reps to Cost Teams |
| Chems/Gases Reduction Team | Buyers (Finance) |
| Spares Reduction Team | D2, F11, and F12 Cost Managers |
| Shift Mangers | Tool Group Leaders |

**Figure 15 -Voice of the customer Interviewees at D2**

Representatives were targeted from each of the commodity teams tasked with cost reduction. Where possible, counterparts from other members of the virtual factory were included. The list of interviewees started at about 20, but grew with time as one person would recommend that someone else be added.

The following lessons resulted from the interviews and could be consider best practices when conducting the VOC interviews:

- Approximately one hour was adequate for the VOC interviews. Most people have very full schedules and may be reluctant to give up time if they were not getting something in return.
- Providing a list of topics or questions before the meeting allowed the interviewees to consider issues in advance. Figure 16 shows the interview topics used.
- Greater insight resulted from the interviews when they were conducted as open-ended discussions, rather than going through a standard list of questions.
- A good mix of time was to spend the first 1/2 of the interview dealing with functional aspects of cost reduction such as organization and decision making methods, and the second half focusing on managerial aspects; how they manage or are managed, how they keep track of their efforts, and how they report up and down the chain.
- Every interview began with an explanation of the scope and purpose of the cost reduction methodology project and the reason they were chosen for the interview. Often this lead to specific examples of cost projects with which they were involved.
- Interviewees examples of successful and unsuccessful cost reduction efforts, best practices they could pass along to other teams, and needs that they could identify that would help them perform cost reduction more effectively were very useful.

- It was useful to transcribe notes into a structured framework (SWOT) within 48 hours while the interview was still fresh in memory. Direct quotes and images were captured whenever possible.

| **Voice of the Customer Interview Guideline Questions** | |
|---|---|
| Functional | |
| | How/to what extent are you involved in cost cutting? |
| | How is your cost effort organized? (team/individual, representation) |
| | How do you decide what projects to take on? (risk, net savings, payback period, low hanging fruit) |
| | What kinds of tools or skills did you use? What have you lacked? |
| | What methods have worked well for you? What has not worked well? |
| Managerial | |
| | How was the "cost" message delivered to you? Is the message consistent? |
| | How do you track progress? What metrics do you use? |
| | How do you communicate the results to management? |
| | What roadblocks have you encountered? |
| | What would you change with respect to the cost reduction effort? |
| | Who else should I talk to? |

**Figure 16 VOC Interview Questions**

## 3.4. Interview Data Analysis

Below is a summary of the strengths, weaknesses, opportunities, and threats (SWOT) extracted from the interviews of D2 and virtual factory cost team members. The SWOT analysis proved to be an effective framework for organizing the results of the interviews. Based on the SWOT framework, broad generalizations were created and used as basis for developing the functional requirements of a cost management system.

#### 3.4.1.1. Strengths

D2 appeared to be very focused on cost reduction. All levels and functions were well aware of the cost goals. The following specific examples were mentioned repeatedly throughout the interviews as strengths of the ongoing cost reduction effort:

- *Healthy competition among fabs*: Each week, D2 compared itself to the virtual factory average and the virtual factory best across many cost areas. This created a healthy sense of urgency and competition, and motivated the cost teams to seek out improvements.

- *Consistent message*: There appeared to be a consistent message delivered to and by all levels of management that cost reduction was a priority. By explicitly targeting cost reduction, the Quarterly Business Unit Meeting (BUM) delivered by top management to the entire company helped reinforce the significance of cost reduction to company performance.

- *Thinking globally-Acting locally*: Cost targets were set at the virtual factory level, yet commitments were assigned at lowest level possible. Individual technicians and tool owners were assigned and held accountable for identifying and executing cost reduction opportunities. This sense of ownership helped to ensure that the projects were accomplished.

- *Bottoms-Up-Estimates*: The BUEs were widely considered valuable tools for managing cost. Some believed that the exercise of building the BUE was equally as useful as actually using it afterwards because they learned about the costs associated with their tool.

- *X-functional teams*: Technicians, engineering, finance, and purchasing were all represented and actively participated on the cost teams.

- *White papers*: The White paper process, a formal review at the virtual factory level of significant process or equipment modifications, was considered an effective mechanism for monitoring and implementing change.

#### 3.4.1.2. Weaknesses

The majority of perceived weaknesses associated with the cost reduction effort were cultural. Risk aversion, many believed, was the single greatest barrier to cost reduction. The low hanging fruit had been picked, and the next generation of projects would require greater risk-taking. The following items were mentioned as specific problems within the cost effort.

- *Yield culture is unyielding*: Despite the need to reduce costs, the culture within the factory was still strongly tied to obtaining high yields and low cycle time. There was little room for compromise in these two areas, even when significant cost savings were possible.

- *D2 too risk averse*: Risk was generally not quantified when evaluating a change. Elements of risk included issues such as "can we execute the change?", "will the change deliver the desired result?" or "will there be unanticipated side effects of change?" Instead of quantifying the risk, people tended to gravitate toward projects that contained little risk.
- *VF team structure lacked leadership*: Many decisions that affect cost were made at the virtual factory level and by consensus. At times this meant a single dissenting opinion could suppress a good idea.
- *Weak organizational learning*: While cost teams had made progress, there was no system of continuous learning and informational exchange to help foster future improvements. There was a lack of infrastructure to grow and maintain a cost-conscious culture within the factory. To illustrate the point, cost reduction topics made up only a few percent of the 1998 papers submitted at a company-wide learning conference despite cost reduction being a top priority for the company that year. Cost reduction was considered a one-time event rather than a permanent way of thinking and working.

#### 3.4.1.3. Opportunities

The following comments describe activities that were suggested as opportunities to make significant improvements to the cost reduction effort.

- *Introduce cost improvements earlier in the ramp*: Many felt that by the time cost is considered in the process development, many variables had been fixed. If process development and cost reduction were pursued more concurrently, greater improvements could have been achieved.
- *Use excess capacity to experiment*: Excess capacity in the virtual factory was a significant opportunity to experiment with cost reduction ideas. Experiments with labor, for example, had been carried out with great success.
- *Use D2's stock of PhDs to reduce costs*: There was great brain-power within D2 that could have been applied to cost reduction. D2 should have been the leading exporter of cost reduction ideas, but in fact it often lagged the high volume factories.
- *Free up technical staff with administrative help*: Cost team leaders and members spent many hours each week creating materials to present at cost meetings. An administrative assistant dedicated to cost reduction could collect and disseminate cost project information, prepare presentation slides, etc., freeing technical people to spend time working on projects.

#### 3.4.1.4. Threats

The following items were considered significant threats or barriers to the cost reduction effort:

- The capacity-constrained mindset: D2 made decisions as though it was capacity constrained when the virtual factory actually had excess capacity. When a factory is capacity constrained, one can apply constraint-theory to bottlenecks to justify high production costs. In a market-constrained environment, however, one should consider the marginal cost of a wafer, not revenue, when justifying production costs, such as monitors.
- *Suppliers and vendors had no incentive to reduce*: Vendors were an excellent source of knowledge and ideas for reducing costs. However, Intel did not provide the vendors an incentive to share the information. In many cases, it was contrary to the vendor's interest to help Intel cut costs. Vendors also claimed that Intel has been resistant to adopt their ideas.
- *There were no daily cost indicators at the factory floor level*: There was a perceived lack of cost information in some tool groups. This appeared to be an organizational flaw, rather than a real lack of information, since other tool groups had ample access to real-time cost data.
- *Scarce low hanging fruit*: It is widely believed all of the "low risk, high reward" projects would be completed by the forth quarter of 1998. After the current round of projects, there would not be enough good projects to keep the momentum going and costs would plateau.
- *False perception of improvements*: The bulk of the projected wafer cost savings resulted from increased production at D2 rather than cost reduction efforts. While increased utilization helped make D2 more competitive with the virtual factory, it artificially inflated the effectiveness of cost reduction efforts and may also have inhibited the development mission of D2.

## 3.5. Cost Project Tracking Systems

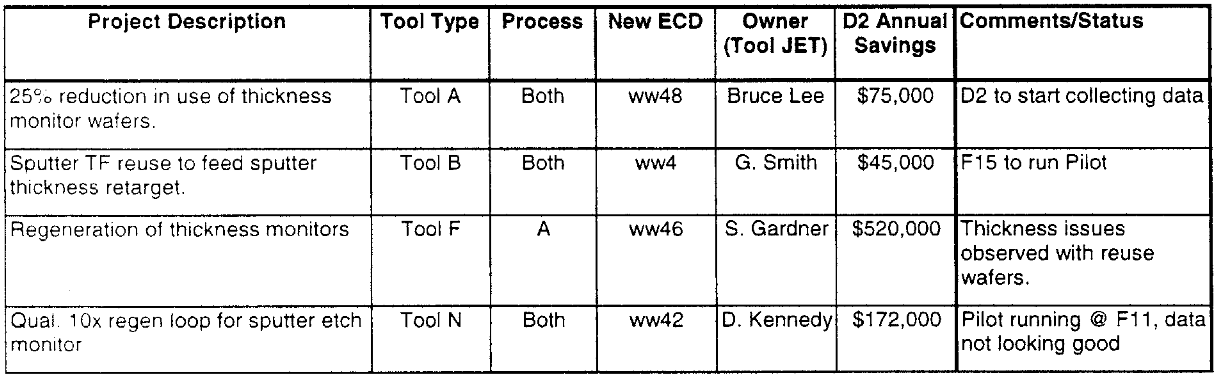
A by-product of the interviews was a collection of examples of how people managed their cost reduction efforts. Methods varied from pencil-and-paper lists to detailed databases containing hundreds of projects. For some groups, creating the list was a one-time event and the list was rarely if ever revisited, while others reflected on their list daily or weekly. There were many independent project databases and project evaluation tools in circulation. Each database, spreadsheet, and project evaluation tool required considerable time to create and even more to maintain.

### 3.5.1. Commodity Team Solutions

Each commodity team had its own method of tracking projects. While there were differences in style and format, they all contained the same basic components and served the same purpose.

#### 3.5.1.1. Monitors

The monitor reduction team had a spreadsheet format that all tool groups started out using, but then modified to meet their specific naming conventions and format needs. For example, the projects for an in-line monitor reduction project were primarily focussed on labor savings, and measured effectiveness in technician hours saved. Test-wafer projects, like those shown in Figure 17, might involve reduced consumption of test wafers through increased reuse or elimination of monitor steps. The original monitor team format did not work for both kinds of projects.



| Project Description | Tool Type | Process | New ECD | Owner (Tool JET) | D2 Annual Savings | Comments/Status |
|---|---|---|---|---|---|---|
| 25% reduction in use of thickness monitor wafers. | Tool A | Both | ww48 | Bruce Lee | $75,000 | D2 to start collecting data |
| Sputter TF reuse to feed sputter thickness retarget. | Tool B | Both | ww4 | G. Smith | $45,000 | F15 to run Pilot |
| Regeneration of thickness monitors | Tool F | A | ww46 | S. Gardner | $520,000 | Thickness issues observed with reuse wafers. |
| Qual. 10x regen loop for sputter etch monitor | Tool N | Both | ww42 | D. Kennedy | $172,000 | Pilot running @ F11, data not looking good |

**Figure 17 Sample Project Tracking Document**

#### 3.5.1.2. Spares

The spares teams were using an MS Access 2.0 database borrowed from F11 to maintain all their projects. The F11 database offered substantial search, filter, and report capability; however, few project owners were comfortable enough with Access to use the database effectively. Most kept separate lists that they used for routine tracking of projects.

#### 3.5.1.3. Chems/Gases

Chemicals and gases projects were generally less complicated, and were captured in a single list maintained by the team leader. This allowed the team leader the opportunity to add up the net effect of many projects, in whichever combination she needed. This ability to aggregate results of many projects made her list useful for forecasting future costs.

#### 3.5.1.4. Labor

As a result of an all-day labor reduction meeting and subsequent brainstorming sessions, D2 had in excess of 700 different labor projects on record. Of the 700, about 400 were considered active, at some level. These projects were maintained in five or more separate Excel spreadsheet files, but all used the same format. Separate files were created to avoid conflicts between groups needed to access the data at the same time.

### 3.5.2. Common Weaknesses

All or most of the existing cost project tracking systems shared common weaknesses.

#### 3.5.2.1. Redundant data

Many people were expected to keep track of the same projects in more than one place. For example, one member of the monitor reduction team had a spreadsheet of her own that she used to document projects for her tool group meetings, and another spreadsheet that was supposed to be used for the monitor reduction team.

#### 3.5.2.2. Lack of Consistency and data integrity

There was no consistency among the cost teams, no way to aggregate the benefits of many projects, and no way to forecast savings into the future. Some projects measured savings in total dollars, others in dollars/wafer, and others in terms of labor hours. A lack of consistency in data formats made the information difficult to manipulate. For example, all projects have an estimated completion date. Sometimes the completion date was in a standard date format (2/15/99); other times in a workweek format (WW 43); other times listed by quarter (Q2 1999). Often times projects would be recorded without key pieces of information like estimated savings or completion dates. Without data consistency, the project information could not be manipulated, sorted, or aggregated. The data was only useful as a static list.

#### 3.5.2.3. No link between projects and projections

There were tables and charts containing cost projects and other tables and charts showing wafer cost-cutting goals, but no logical link between the two. Long range wafer cost goals were established and committed to without the benefit of knowing what projects were planned, what the benefit would be, or the when the savings would be realized. As a first cut, management might say, "everyone cut costs by 30%" without tying the goals to the existing project roadmaps.

#### 3.5.2.4. The process X Cost scorecard

An interesting exception was the process X Cost Scorecard. A team was formed specifically to track the costs associated with the process X process used to make FLASH memory. Intel had to drastically cut costs to meet a price set by the open market. The process X Scorecard team developed a baseline cost made up of ten or more detailed cost categories (e.g., the cost of clean-room gowns). From this baseline, the process X Scorecard team established goals for each cost team that would allow Intel to meet the market price plus margin. The process X Scorecard team then tasked the cost teams to provide project roadmaps that would allow the teams to make the cost target. If the cost team could not meet their targets, they would appeal to the Scorecard team, and the savings allocated elsewhere. This was an excellent example of target costing.

#### 3.5.2.5. Arbitrary assumptions

The cost reduction teams estimated economic benefit for each project without a common set of assumptions and guidelines. Assumptions regarding factory loadings, wafer costs, labor rates, and product mix were at the discretion of the person planning the project. Therefore, comparing the relative benefits of two projects was often difficult.

## 3.6. Data collection conclusions

Data collection occurred in two forms. The first form was passive, by observing, participating on teams, and by testing reactions to new ideas and tools. The other form of data collection was active, through surveys or interviews.

Conclusions from the interviews were that Intel was very focused on cost reduction, but there were aspects of the organization and culture that may have been preventing greater improvements. There was a variety of tools in place to help with cost reduction, but there was no single solution that applied to the entire factory and satisfied the needs of the many stakeholders.

The purpose of the data collection was to help in the next step, which was to define a project and the functional requirements for a system to provide sustained cost competitiveness at Intel. Functional requirements are covered in the next chapter.

# 4. Project Definition and Functional Requirements

## 4.1. Chapter Overview

During the Project Definition phase, preliminary recommendations were formulated and presented to the factory management. Feedback from the recommendations established a direction for the remainder of the cost reduction methodology project. Considered while defining the project were aggregated findings from research including Voice of the Customer data, best practices, existing tools such as cost models and project tracking databases, and examples of effective cost reduction management from outside literature.

## 4.2. Midstream recommendations

The following preliminary recommendations were presented to D2 management following the data collection phase. The presentation occurred at approximately the midpoint of the six-month work.

Strategic recommendations:

- *D2 should take a leadership role in cost reduction*: deliver a low cost process to the high volume factories.
- *Take risks at D2: Isolate failures, leverage successes.* Create a metric to encourage calculated risk-taking.
- *Introduce cost reduction in process development*: create a cost scorecard for process development.

Tactical recommendations:

- *Develop BUEs at the virtual factory level.*
- *Evaluate projects across functional areas using NPV analysis*: include startup cost, yield, WIP, confidence.
- *Balance priorities between yield, cycle time, cost.*
- *Consolidate and integrate factory tools such as the project databases, BUEs, and cost scorecard.*

## 4.3. Project Definition

Most cost teams had some sort of cost project tracking system that they were currently using. The concept of a list of ongoing projects with due dates, savings estimates, and status seemed to be common across many cost teams. The cost reduction project database (henceforth referred to as Project99) was conceived to provide a master tracking system that incorporated the best practices of all the existing tools.

Project99 would also serve as a medium to demonstrate the midstream recommendations in a tangible way.

The biggest complaint from those administering these lists was that they spent too much of their time managing the list instead of working on the projects themselves. In most cases, there were at least two meetings each week that required substantial effort to prepare status reports, charts, and other supporting material. Another complaint from the cost teams was that there seemed to be little correlation between the projects that they agreed to work on and the cost goals that were mandated for them. Those setting the targets were equally frustrated because they would have preferred to create realistic goals based on a project roadmap that they were confident the teams could achieve, but they did not have accurate savings forecasts from the teams. Management wanted a mechanism for monitoring project performance from a higher level. For example, the operations manager requested that all groups report projects performance using a single report format, and a metric for monitoring which groups were completing projects on time.

## 4.4. Functional Requirements

The following list was a set of functional requirements for Project99:

### 4.4.1. Ease of use

In order to get buy-in from the target users, Project99 had to be very easy to learn. While most people at Intel were very computer literate, they were also very busy and could not afford to take half a day to learn a new project database to replace one with which they were already satisfied. Furthermore, Project99 had to make their life easier on an ongoing basis by saving them time preparing for meetings and tracking projects. Ease of use features included:

- A menu driven user interface.
- User-friendly data entry screens.
- On-line help screens.
- Pre-formatted reports.

### 4.4.2. Consistency of assumptions

Project99 must provide a common set of assumptions for:

- Factory loadings: Approximate production level by week or by quarter in wafer starts per week were required. Loadings were to be itemized by process (e.g., process X, process Y) since not all projects applied to all processes, or they applied but to different degrees. For example, a project reduced cost on a certain processing step. process Y wafers passed through this step six times and process X wafers pass through 3 times. The process Y savings was be 2X the process X savings.

- Labor rates: A consistent policy for tracking and valuing labor savings was required. Many projects reduced technician labor. In order to compare a labor in terms of $/ws the labor rate had to be converted to dollars. There was disagreement among groups about how to value labor savings. Some groups did not consider labor savings true savings unless an entire technician, or "head", was eliminated. Others valued fractional labor savings, but tracked the savings separate from other cash savings.
- Wafer costs: Assumptions about the value of a wafer made a significant difference in the economic value assigned to a project, or whether a project was even beneficial. Project99 required a consistent policy for establishing the value of a wafer that distinguishes between processes (process X vs. process Y) and considers how much value has been added to the wafer.
- Factory constants: Other factory constants included the discount rate (for NPV analysis), a dollar value for increments of cycle time or WIP-turn improvements, and yield impact.

### 4.4.3. Features to ensure accuracy

Project99 had to contain tools to help calculate project savings using before and after scenarios. Where possible, Project99 had to connect to other databases, such as lists containing Intel part numbers and prices, to assure that cost savings were calculated using accurate data. A link to a BUE cost model was desirable, so savings could be derived by comparing costs before and after a proposed project.

### 4.4.4. Data integrity

Controls needed to be in place to ensure that data is entered into the system in a consistent format. This ensured that the project data could be sorted and manipulated. Also, certain pieces of information were absolutely required, such as the project completion date, to ensure the project is fully characterized.

### 4.4.5. Normalization

Project99 needed to provide the tools to estimate all projects in terms of $/wafer start savings. Since few projects can be directly reduced to a $/ws savings, conversion factors may be required to normalize the savings. Examples might include converting savings per lot or batch, savings per PM, or savings per month. Converting time-based savings to wafer based savings is particularly tricky. A project may save a fixed amount per unit time, so the $/ws savings varies with factory loadings.

### 4.4.6. Forecasting ability

Project99 needed to consider estimated savings, factory loadings and estimated completion dates to project \$/ws savings into the future. The result should be an expected cash flow resulting from the project. This cash flow would be used to perform NPV analysis to prioritize projects and to create project roadmaps (see Aggregate results below).

### 4.4.7. Risk-management

Project99 needed to quantify and manage project risk in three ways:

- Yield impact: A project may have a positive or negative impact on line yield[5] or die yield[6]. For example, eliminating a monitor step may save labor but decrease line-yield as a result of having to scrap wafers further downstream. To effectively manage risk and make quantitative risk-benefit decisions, a dollar value was established for a % of line yield and an ISO point of die yield.
- Effect of an excursion: An excursion can be defined as a rare, catastrophic, expensive event. Risk of an excursion, such as a pump breaking down or a large loss of wafers, might be enough of a reason not to pursue a project. A simple probabilistic risk assessment[7] helped quantify the risk in order to make a rational decision.
- Certainty of the project outcome: Some projects have a greater probability of success than others. Various project tracking tools used "confidence" ratings to classify projects. Medium or low confidence projects were avoided as a result, even though they had a high potential payoff. A methodology to assess confidence and to discount the project value appropriately allowed for a well-balanced portfolio of projects.

### 4.4.8. Prioritization

Project99 had to estimate the net present value of the expected savings of a project to help estimate the cash value of a project and prioritize projects accordingly. Projects could be prioritized by \$/ws savings

[5] Line yield is the measure of wafers that successfully complete the manufacturing process, measured in %.

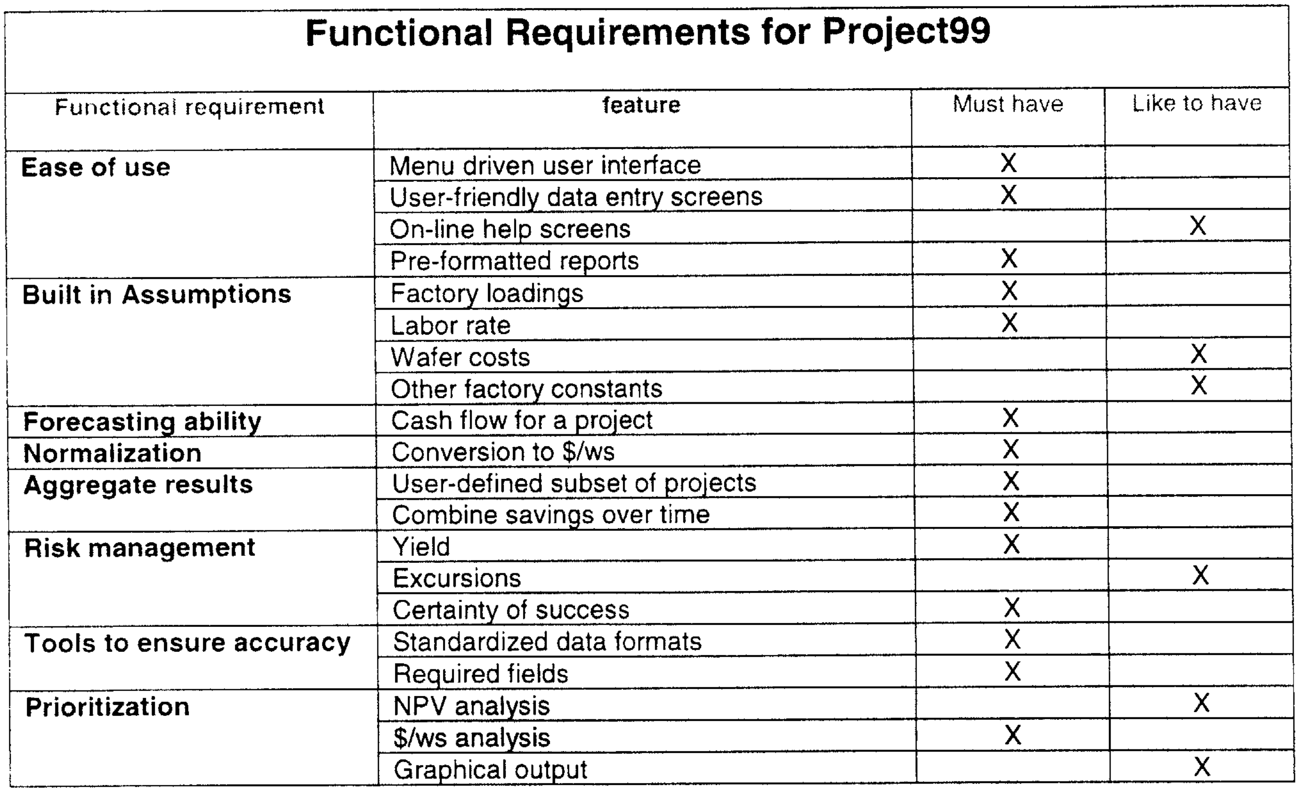
[6] Die yield is a measure of die, or individual chips, that function satisfactorily on a wafer, measured in ISO points.

[7] For example, an excursion may have a probility of occuring once every 100,000 wafers and have an economic impact of \$25,000. The expected cost is \$0.25/ws.

or by NPV. Graphical output of project prioritization would be helpful. At Intel, Pareto diagrams are widely understood and were ideal for presenting the prioritization results.

### 4.4.9. Aggregate results

Project99 had to allow management to aggregate a portfolio of projects. This would allow management to determine if a given set of projects will result in the target wafer cost at the desired time. The user had to be able to define and select a subset of projects and show the combined impact as a function of time.



**Functional Requirements for Project99**

| Functional requirement | feature | Must have | Like to have |
|---|---|---|---|
| **Ease of use** | Menu driven user interface | X | |
| | User-friendly data entry screens | X | |
| | On-line help screens | | X |
| | Pre-formatted reports | X | |
| **Built in Assumptions** | Factory loadings | X | |
| | Labor rate | X | |
| | Wafer costs | | X |
| | Other factory constants | | X |
| **Forecasting ability** | Cash flow for a project | X | |
| **Normalization** | Conversion to $/ws | X | |
| **Aggregate results** | User-defined subset of projects | X | |
| | Combine savings over time | X | |
| **Risk management** | Yield | X | |
| | Excursions | | X |
| | Certainty of success | X | |
| **Tools to ensure accuracy** | Standardized data formats | X | |
| | Required fields | X | |
| **Prioritization** | NPV analysis | | X |
| | $/ws analysis | X | |
| | Graphical output | | X |

**Figure 18 - Project Database Functional Requirements**

Figure 18 is a summary table of functional requirements, prioritized as "must have" or "like to have". The functional requirements were reviewed with several key stakeholders involved with the project and adjustments made. Requirement issues such as multi-user capabilities and database security were documented but not included in the initial scope. As the basis for a thesis, serious tradeoffs were required between academic needs (e.g., research content) and requirements as an Intel resource. The goal was to develop a tool primarily as a prototype and as a mechanism to present ideas, and secondarily as a functional and useful management tool.

## 4.5. Project definition conclusions

The project definition phase of the project required a great deal of communication with all of the stakeholders. It was essential to establish precisely what the problem was before trying to propose a solution. Figure 6 was used to show the existing tools and relationships to illustrate what was believed to be the problem along with a proposed solution. In hindsight, it might have been more effective to break this into two discrete steps; spending more time working with the stakeholders to develop a more insightful problem statement before proposing a solution.

Given that an accurate and meaningful problem statement has been defined, a set of functional requirements was then established. Voice-of-the-Customer data was referred to often when developing the functional requirements. Implementation considerations were largely ignored when defining requirements. Instead, there was a focus first on satisfying a need. Implementation challenges are addressed in the next chapter.

# 5. Prototype Development

## 5.1. Chapter Overview

The Project99 prototype was a working model that demonstrated ideas, concepts, functionality, and relationships within the Intel cost reduction methodology. Functional requirements detailed in the previous chapter were used as a roadmap for developing the prototype computer application. Prototype development included choosing an environment, creating a database structure, developing report formats, and creating a user interface for entering and retrieving information.

## 5.2. Computing Environment

The prototype was created using the latest version of MS Access as the computing environment. MS Access has features that allow a non-programmer to create a relational database with menu screens, data-entry forms, and reports that are customized, easy to use, and easy to modify. MS Excel was also considered, since nearly everyone in the organization was experienced with Excel and the transition would be less of an event. However, the existing project lists that were maintained in Excel were difficult to understand if you were not the original creator and not appropriate for a general use tool.

One problem encountered was that the version (2.0) of MS Access that was currently installed on most of the desktops at D2 was obsolete and scheduled to be discontinued at the end of 1998. The new version (Access 97) was available, but required technical support from Intel's Information Technology Support Group to install. Database applications from the old version of Access required conversion to work with the new version, and not all features ported from one to the other. This was an advantage in that it provided a good excuse for those using Access to track projects to consider my project. It was a disadvantage in that the prototype had to be ported from the old version half way to completion. Another disadvantage was that at the end of implementation in December of 1998, very few people had Access 97 installed on their computers.

## 5.3. Database Structure

The database is the backbone for the system. It was very important to carefully design a database structure from the beginning to ensure the necessary relationships were in place. Careful design up-front saved considerable time later because rework and debugging were reduced.

Project99 contained a table of all the unique information pertaining to a cost project. A review of all the existing cost project databases generated an inclusive list of all of the different types of information, or fields[8], that had been captured. Examples of project database fields are the name of the project, the completion date, the estimated savings, or the name of the project owner. Any redundant fields were eliminated. The remaining list became the basis for the project table. Each field was classified into one of several groups. The groups are shown in Figure 19.

**Project99 Table Field Classifications**

| Field Classification | Description |
|---|---|
| project description | name, functional area, commodity, applicable factories and processes, etc. |
| ownership | project owner name, phone number, pager number |
| dates | estimated and actual completion dates |
| startup costs | one-time labor, parts, and qualification costs required for project |
| yield/cycle time impact | pos/neg impact on line yield, die yield, or cycle time |
| wafer-based savings | per unit (wafer) savings as a result of project (by commodity) |
| time-based savings | per unit time savings as a result project (by commodity) |
| excursion costs | probability and expected cost of an excursion |
| confidence | multi-dimensional measure of project confidence that the project can be executed as planned and the projected gains realized |

**Figure 19 - Project Database Field Classifications**

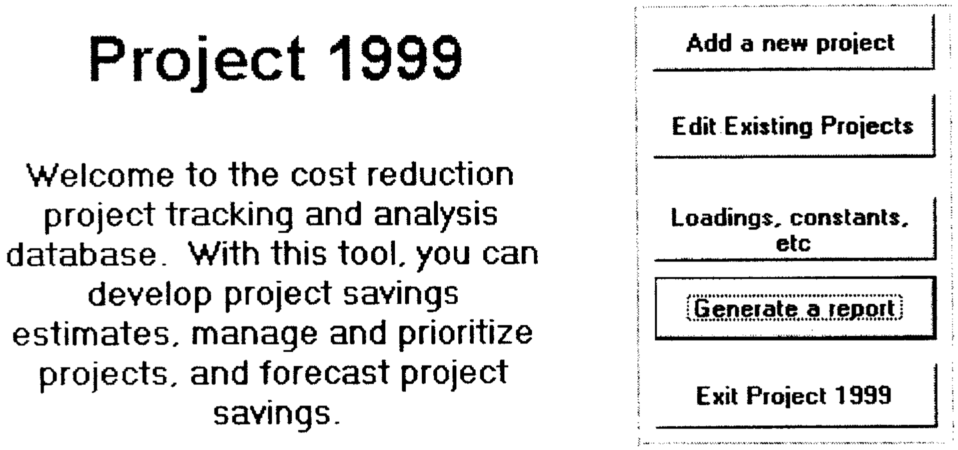
In addition to the project table, the database required tables to store other information not directly related to the project, such as loadings (itemized by process and factory); prices for consumables such as spare parts, chemicals, gases; and factory constants such as labor costs, wafer costs, and yield impact. The finance group was an excellent resource for this kind of information.

Once tables were created, relationships among the tables were required. The combination of tables and relationships is called a relational database. Relational databases offer several advantages over a "flat" database, such as those used at Intel to track projects. A flat database is generally a two dimensional table. A relational database could be made up of many tables linked together through key fields, such as a zip code, part number, or student ID. A relational database environment was ideal for this project, since the table of projects had to link with other tables containing related data for spare parts, factory loadings, product mix, etc.

[8] Database fields can contain text, numbers, dates, Boolean values, or other specific kind of information.

## 5.4. User interface

A thoughtfully-designed user interface made Project99 easy to learn and fast to use. The prototype was completely menu-driven and contained helpful data entry screens to ensure complete and accurate project descriptions. Once the program was launched, anyone who was accustomed to using common computer applications could use the database with little or no training or assistance.



Project 1999

Welcome to the cost reduction project tracking and analysis database. With this tool, you can develop project savings estimates, manage and prioritize projects, and forecast project savings.

**Figure 20 - Project Database Main Screen**

The main welcome screen (shown in Figure 20) welcomed the user and contained buttons to click for all of the major functions: add a new project, edit existing project, edit factory constants, generate a report, and exit. A navigation bar with menu buttons helped users find a specific project or set of projects.

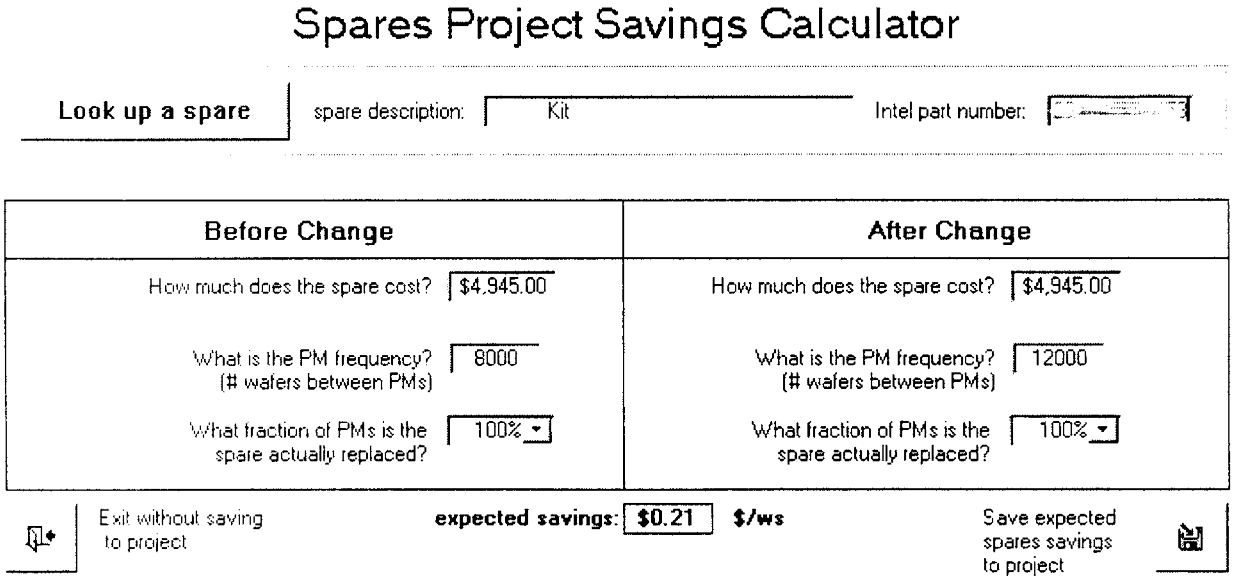
## 5.5. Data Entry Screens

The prototype used a series of data entry screens to help the user characterize their project. There were many pieces of information required for each project, so it was helpful to break the data fields into groups. The prototype data entry screens used the project field classifications described in Figure 19 to segregate the data. Each classification group had its own screen so the user could concentrate on a single topic.

Wherever possible, the prototype used list boxes that let the user choose from a list of entries. This was easier for the user since less typing was required. Controlled data entry was extremely helpful to ensure that the appropriate data was entered and the right format was used. The prototype used list boxes to enter tool groups, tools, functional areas, and dates. The prototype used check boxes to enter Boolean (yes/no) data. Again, they are easier for the user and they maintain data integrity by not allowing someone to make an invalid entry or to mistype something. Figure 9 through 13 show five Project99 data entry screens, and contain examples of check boxes and a list boxes.

## 5.6. Project Savings Calculators

In order to help ensure that users followed a common methodology for determining the project savings, the prototype contained calculators for each commodity. The calculators helped quantify projects in the language and units of the commodity and normalized the project savings into $/ws. This was an important functional requirement to the commodity cost team leaders, who were concerned that the savings estimates were inaccurate and inconsistent among projects.



Spares Project Savings Calculator

Look up a spare

spare description: Kit

Intel part number:

| Before Change | | After Change | |
|---|---|---|---|
| How much does the spare cost? | $4,945.00 | How much does the spare cost? | $4,945.00 |
| What is the PM frequency? (# wafers between PMs) | 8000 | What is the PM frequency? (# wafers between PMs) | 12000 |
| What fraction of PMs is the spare actually replaced? | 100% | What fraction of PMs is the spare actually replaced? | 100% |

Exit without saving to project

expected savings: $0.21 $/ws

Save expected spares savings to project

**Figure 21 - Spares Calculator**

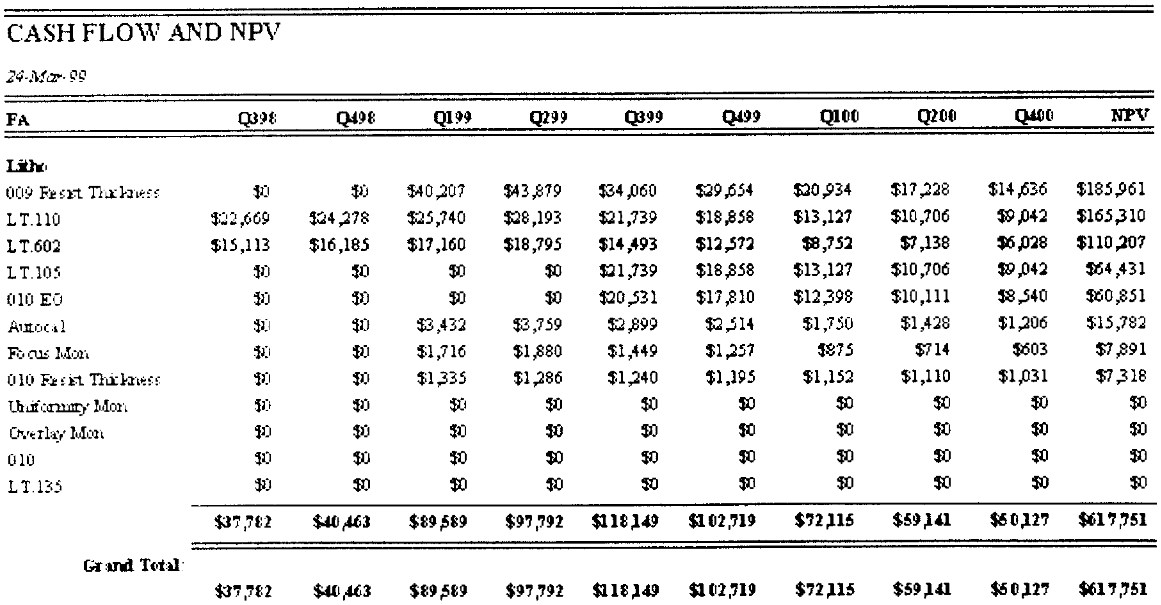
Figure 21 shows the prototype calculator for spares projects. In this fictitious example, a PM kit that costs $4.945 is replaced as part of a PM every 8000 wafers. The project involves extending the PM to replace the PM kit every 12000 wafers. The calculator determines the wafer savings by determining the difference between before and after wafer costs for this PM. Other PM project scenarios that could be handled by the calculator were a change in the price or a change in the fractional time that the part was consumed (e.g., if the part is inspected and only replaced if worn). The spares and chems/gases calculators were also linked to databases of prices for thousands of parts used by Intel to ensure that the savings is based on an accurate price. Use of the calculators in the prototype was optional and the values could be over-ridden. They were provided as a back-of-the-envelope calculation aid. Savings could also be calculated using the BUE or other tools. As discussed in the implementation chapter, the calculators became a point of contention if they were perceived as inaccurate or unsophisticated. For the purposes of a prototype, however, the calculators demonstrated the idea of using a consistent methodology for characterizing a project.

## 5.7. Data Manipulation, Calculations, and Reports

An important requirement of Project99 was the ability to manipulate and present the project information in useful ways. This included performing calculations with the data, sorting, grouping and filtering projects, and generating reports and graphs.

### 5.7.1.1. Mathematical Operations

To understand what kinds of data manipulation and reports were useful, consider how the information was used with the previously existing project databases. For example, the Cost Ops managers preferred a report that showed the name and status of each project, the name of the project owner, the expected completion date, and any roadblocks the project may be facing. This kind of report did not require any calculations, but might require the ability to sort projects by completion date and to filter projects to show only those involving lithography, for example.



CASH FLOW AND NPV

*24-Mar-99*

| FA | Q398 | Q498 | Q199 | Q299 | Q399 | Q499 | Q100 | Q200 | Q400 | NPV |
|---|---|---|---|---|---|---|---|---|---|---|
| **Litho** | | | | | | | | | | |
| 009 Resist Thickness | $0 | $0 | $40,207 | $43,879 | $34,060 | $29,654 | $20,934 | $17,228 | $14,636 | $185,961 |
| LT.110 | $22,669 | $24,278 | $25,740 | $28,193 | $21,739 | $18,858 | $13,127 | $10,706 | $9,042 | $165,310 |
| LT.602 | $15,113 | $16,185 | $17,160 | $18,795 | $14,493 | $12,572 | $8,752 | $7,138 | $6,028 | $110,207 |
| LT.105 | $0 | $0 | $0 | $0 | $21,739 | $18,858 | $13,127 | $10,706 | $9,042 | $64,431 |
| 010 EO | $0 | $0 | $0 | $0 | $20,531 | $17,810 | $12,398 | $10,111 | $8,540 | $60,851 |
| Autocal | $0 | $0 | $3,432 | $3,759 | $2,899 | $2,514 | $1,750 | $1,428 | $1,206 | $15,782 |
| Focus Mon | $0 | $0 | $1,716 | $1,880 | $1,449 | $1,257 | $875 | $714 | $603 | $7,891 |
| 010 Resist Thickness | $0 | $0 | $1,335 | $1,286 | $1,240 | $1,195 | $1,152 | $1,110 | $1,031 | $7,318 |
| Uniformity Mon | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 |
| Overlay Mon | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 |
| 010 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 |
| LT.135 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 | $0 |
| | **$37,782** | **$40,463** | **$89,589** | **$97,792** | **$118,149** | **$102,719** | **$72,115** | **$59,141** | **$50,127** | **$617,751** |
| **Grand Total:** | **$37,782** | **$40,463** | **$89,589** | **$97,792** | **$118,149** | **$102,719** | **$72,115** | **$59,141** | **$50,127** | **$617,751** |

**Figure 22 - Sample Cash Flow and NPV Graph**

Interviews suggested that there was also a need for standardized assumptions for factory loadings in order to project total project savings. This required the ability to associate the per-wafer savings of an individual project to a separate table containing the factory loadings. The resulting product was a project cash flow that could be used by itself to show the effectiveness of the project, aggregated with other projects, or converted into an NPV. Figure 22 shows a report that displays both the cash flow and the NPV for a subset of projects for the lithography group.

A third resource for defining functionality was the collection of project evaluation tools already in circulation. For example, F12 created a tool to help prioritize projects according to the expected savings weighted by a confidence factor. Project99 implemented this feature by providing the ability to calculate an NPV for a project adjusted for confidence, startup costs, yield impact, etc. Figure 23 shows a Pareto diagram of the adjusted NPV for seven projects.

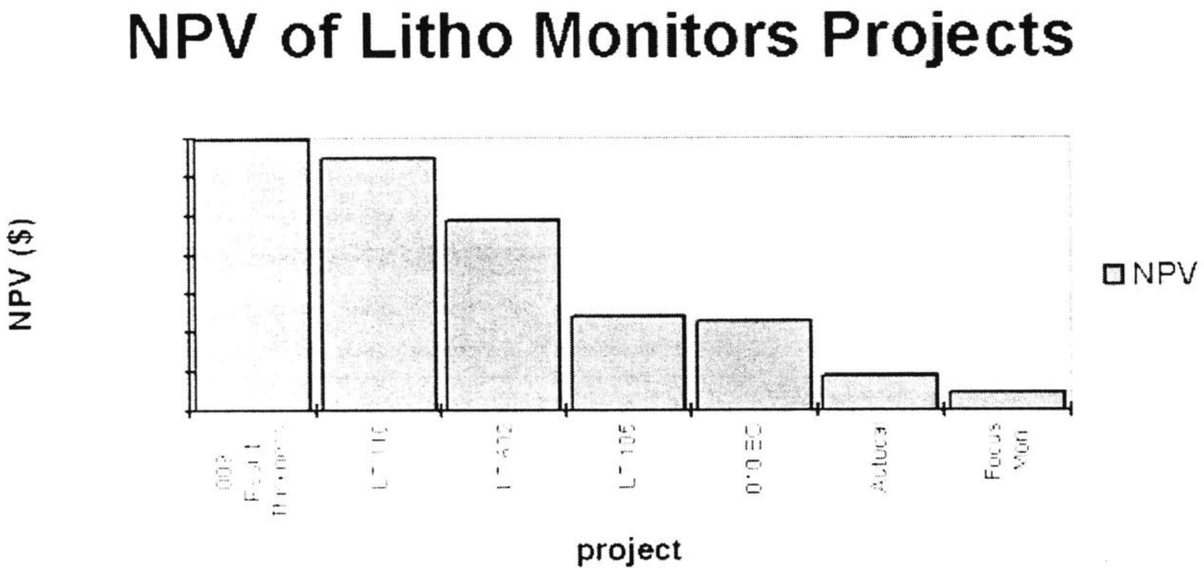
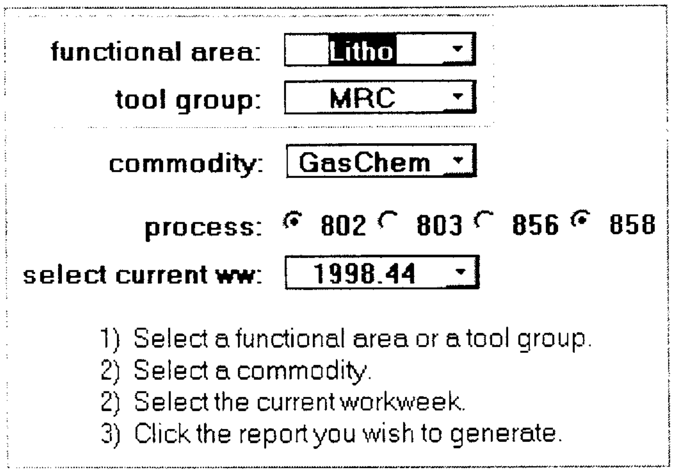


**Figure 23 - Pareto Diagram of Project Net Present Value**

#### 5.7.1.2. Sorting, Grouping, and Filtering

Project99 had to allow users to sort, group and filter projects according to their requirements. For example, factory managers wanted to show all projects grouped by functional area, team leaders wanted only projects for their commodity (e.g., show all spare parts projects), and tool owners wanted to display only those projects for which they were responsible. Figure 24 shows a screen that a user could use to filter projects by functional area, tool group, commodity, process, estimated completion date, or any combination of these criteria. This kind of simple filter screen let the user quickly narrow the scope of a report to include only the projects of interest.



functional area: Litho

tool group: MRC

commodity: GasChem

process: ● 802 ○ 803 ○ 856 ● 858

select current ww: 1998.44

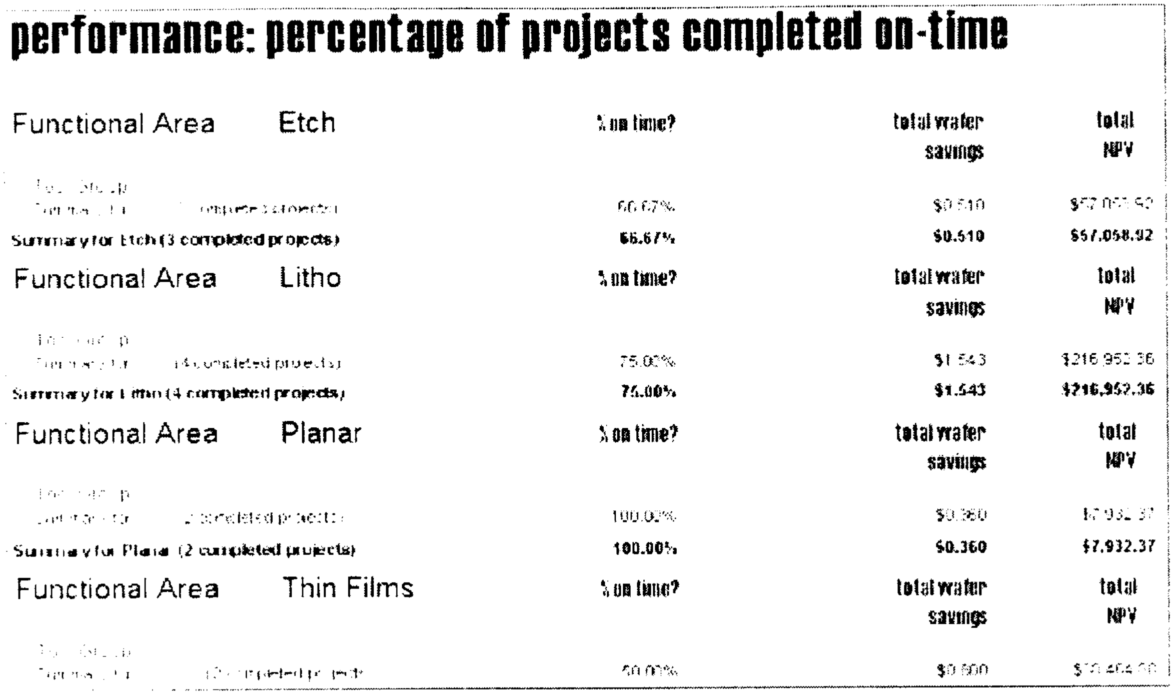
1) Select a functional area or a tool group.
2) Select a commodity.
2) Select the current workweek.
3) Click the report you wish to generate.

**Figure 24 - Project Selection Criteria Screen**

### 5.7.1.3. Reports and Graphs

Users needed to be able to retrieve project data in a meaningful way. MS Access allows users to export data to MS Excel. However, one of the ease-of-use customer requirements was to provide pre-formatted reports that would save people time in preparing reports and to provide a uniform format for all teams. Project99 contained a small number of standard reports that were useful to a broad audience, but also demonstrated the report capability so users could develop their own reports

In order for the prototype to appeal to a broad audience, reports spanned several levels of detail. For example, the $/ws report shown in Figure 22 was well suited for the team working on the projects, but contained more detail than the factory manager wanted or needed. After seeing the project database for the first time, the operations manager at D2 requested a report that only showed the project completion performance grouped by functional area. Figure 25 is a sample of the on-time performance report. Note that there are no specific projects in the report, only summary information.



**performance: percentage of projects completed on-time**

| Functional Area | | % on time? | total water savings | total NPV |
|---|---|---|---|---|
| Functional Area | Etch | % on time? | total water savings | total NPV |
| [illegible] | | | | |
| Summary for [illegible] (3 completed projects) | | 66.67% | $0.510 | $57,058.92 |
| Summary for Etch (3 completed projects) | | 66.67% | $0.510 | $57,058.92 |
| Functional Area | Litho | % on time? | total water savings | total NPV |
| [illegible] | | | | |
| Summary for [illegible] (4 completed projects) | | 75.00% | $1.543 | $216,952.36 |
| Summary for Litho (4 completed projects) | | 75.00% | $1.543 | $216,952.36 |
| Functional Area | Planar | % on time? | total water savings | total NPV |
| [illegible] | | | | |
| Summary for [illegible] (2 completed projects) | | 100.00% | $0.360 | $7,932.37 |
| Summary for Planar (2 completed projects) | | 100.00% | $0.360 | $7,932.37 |
| Functional Area | Thin Films | % on time? | total water savings | total NPV |
| [illegible] | | | | |
| Summary for [illegible] (2 completed projects) | | 50.00% | $0.600 | [illegible] |

Figure 25 - Sample Report

Graphs were particularly helpful tools to present project savings data. One shortcoming of MS Access 97 is that it has very limited graphing capability. The good news was that users could export report data from Access to Excel, which had extensive graphing capability. Figure 5 is an example of a graph created from the report data from Figure 22. This report shows a portfolio of cost reduction projects, with the projected savings accumulated over time. The X-axis is time, measured in workweeks, until the estimated completion date. The Y-axis shows the $/ws savings for this set of projects. At a glance, a project team could see which projects were coming due soon as well as the absolute and relative savings. This helped the team manage resources and forecast project savings into the future (another functional requirement).

## 5.8. Prototype Development Conclusions

This chapter describes prototype development as a discrete step, but in fact it was an iterative process. As parts of the prototype were developed, data collection and functional requirements were revisited many times. It was important to keep in perspective the goal of a prototype of this sort. The prototype was a mechanism to demonstrate and test ideas, and not necessarily a perfectly finished product the first time through (not unlike this document).

The prototype development process was an opportunity to gain understanding of how pieces of information were related and if necessary pieces were missing. The calculators, for example, required additional information in the form of part lists that were not part of the original database design. The multi-dimensional relationship among project savings, processes, factories, and time became overwhelming, so understanding the relationships through good database design made the database much faster to develop.

Data integrity was necessary to make the project database useful. Creating the data entry systems with tools that restricted input (like the list boxes) and guided estimates (like the calculators) when the data was going in helped to ensure the information could be manipulated correctly on the way back out.

In order to be adopted by the organization, the prototype had to show that the new system would make users' lives easier rather than harder. Entering data, navigating around the database, and generating reports had to be overwhelmingly better and faster than whatever they were using before, otherwise they would not change.

# 6. Pilot Program

## 6.1. Chapter Overview

When a suitable prototype for a cost management system had been developed, it was time to test in a "live" environment. A pilot program was preferable to a full-scale implementation so the system could be tested in a controlled environment and problems fixed before introduction to an entire factory. This chapter explores the steps required to run the pilot program and discuss some lessons learned from the D2 pilot program. The pilot program included installing a working prototype on a computer network, selecting a pilot group, populating the prototype database with cost reduction projects, training users, and incorporating their feedback.

## 6.2. Installing the prototype

In order for the pilot program to effectively flush out as many potential problems and design issues as possible, the prototype had to be accessible to those who needed to use it in as realistic environment as possible. The prototype was installed on an internal computer network. Everyone in the virtual factory could be given access to the prototype directory on an as-needed basis. However, this took some effort on the part of the network administrator to manage a list of users who needed access. Ideally, the prototype would have been installed such that everyone could *access* it easily without any special permission, but the owner would manage a list of who could *make changes*. An intranet web-based application would have been ideal.

## 6.3. Selecting a Target Group

The pilot group was a subset of D2 whose members became the lead users of the prototype. Rather than randomly selecting 50 unrelated projects, it made the most sense to focus on a group or team of users that have like projects. By piloting with a group, not only were the individual project analysis tools useful, but also the benefits of being able to study a portfolio of similar projects became more apparent. Also, any disruption in work practices was contained within a single area. The downside was that the experience, feedback, and subsequent modifications were biased toward the pilot group, possibly missing requirements unique to other user groups. At D2, the choices were to introduce the system to a functional area such as lithography, or to a commodity group such as chems/gases.

#### 6.3.1.1. Benefits of Piloting with the Monitor Reduction Team

The monitor reduction team was selected as the pilot for Project99. A commodity team was preferable over a functional area because the cost teams were organized around cost teams. There were already project lists established for monitors that could be ported over to a new system.

Monitor reduction had two main focus areas. The first area was reducing test wafers. Test wafers were wafers sent through a tool for the sole purpose of ensuring the tool was working properly. While they could be reused as test wafers, they did not result in a finished product that could be sold for revenue. An eight-inch test wafer, depending on the formulation and quality, cost on the order of $100. Figure 17 on page 28 is a table containing examples of test wafer reduction projects. The second focus area of monitor reduction was product monitors. These were in-line measurements made on actual product wafers to ensure the manufacturing process was in control. The primary cost for product monitors was labor, since the wafer itself continued on the manufacturing process if it passed inspection.

Monitor reduction had several advantages that potentially made it the best test group. The following elements were considered when planning the pilot:

- Test wafers and in-line monitors were areas rich with cost reduction opportunities. D2's costs for monitors were much higher than the virtual factory average and factory management was anxious to see improvements in this area. A successful pilot program in the monitors area would greatly enhance D2 management support for a large-scale implementation.

- Monitors had an aggressive team working on a variety of projects. Monitors had a well organized, well managed, and well staffed team working on projects. The team members were far enough along in their efforts to have valuable project data, but not so far along that they would not benefit from the prototype.

- Monitor reduction involved calculated risk taking. Monitor projects contain many interesting risk factors addressed by the prototype. Line yield and die yield could be affected by reducing the frequency or eliminating a monitor. Out-of-specification process steps may take longer to be corrected resulting in more bad wafers. Experimental projects introduced risk in the form of failed or limited success. For example, projects involving increased reuse or "regen" of test wafers were sometimes unsuccessful because the wafers degraded over time. These risks factors of the monitor area helped to highlight the features of the prototype.

- The monitor reduction team had many fragmented tracking systems and could benefit from consolidation. As discussed in section 2.5.1.1, the monitor reduction team had several different project lists that were all slightly different. The team leader welcomed an opportunity to consolidate the projects.

## 6.4. Populating the Database with Projects

Populating the database required entering detailed information into the database for all projects. This could have been done one of two ways: entering each piece of information individually, or porting the data in from an existing spreadsheet or database. There were advantages and disadvantages to each.

Entering project data by could be extremely time consuming. Each project may contain hundreds of pieces of discrete information. If the user knew all the required information, just entering the information might only take 15-20 minutes per project. If research was required, characterizing a project could take much longer. It was not unusual for one tool owner to have responsibility for several projects. Since there was no guarantee that the pilot program would become permanent, there was some resistance on the part of the project owners to invest many hours of effort. The benefit from entering project data manually was that the information would be more up-to-date and the users become familiar with the features and functionality of the prototype.

Porting the data over from an existing database was an efficient way of populating the database, since less information had to be retyped. Many pieces of information such as project names, savings estimates, completion dates, and tool names were already recorded and could simply be copied and pasted into the prototype. Only new information needed to be added. Porting project information was especially good if the pilot team was time constrained. A third party, such as the person who developed the prototype, could do the porting for them and the project owners would need only review the project information after it had been entered. The most significant drawback of this approach was that there was less direct input from the project owner. If the information from the old database was inaccurate, out-dated, or inconsistent with the norms of the prototype, the value of Project99 was greatly diminished.

The monitors team chose to have the projects ported over by an ambitious intern associated with the project. The leader of the monitor reduction team was very supportive of the pilot program, to the extent that it was not a burden on the team. An agreement was reached where the existing projects were ported over and then the team took over to clean up any inaccuracies that may have occurred during the transition and provide any additional information that might be required. Porting the data proved to be a very difficult task because each project owner had his or her own way of describing projects. Notation was often very cryptic and the quantitative information was often sketchy or missing altogether. While this made the task more difficult, it also reinforced the need for a better system.

## 6.5. Collecting and Incorporating Customer Feedback

Once the database was populated with projects, users started to see and use the benefits of the cost project management system. The process of entering and verifying project data generated the first bits of customer feedback, followed by feedback about the reports and the user interface. Classifications were used to keep the feedback organized in a useful and productive way. For example, feedback was classified by prototype feature (data entry, filter menu, report X, etc) and by the nature of the comment (feature does not work, suggested features and functions, cosmetic changes, etc.

To demonstrate the features of the prototype to the pilot team, the introduction was broken into steps. First, reports and graphs of interest were created and presented to the entire team. Second, one-on-one time with the managers and project owners showed them how to use the prototype themselves.

Feedback from the D2 monitor reduction team pilot was very useful to help refine and enhance the prototype. The pilot was an opportunity to test report formats on a live audience with real project data. The pilot also provided valuable insight on erroneous assumptions about how they would be using the database and the level of detail required. The following examples demonstrate the kinds of customer feedback that resulted from the pilot program.

- The prototype provided forecasting capabilities by creating tables and graphs indicating projected savings quarterly, three years into the future. The monitor reduction team pointed out that its project horizon rarely extended past four or five months, and that forecasts should be incremented in weeks rather than quarters.

- The Test Wafer Savings Calculator was a huge failure. The test wafer team already had a much more elaborate tool for measuring test wafer costs. If anything, including the calculator made the entire prototype less credible in the eyes of the monitor team.

- The NPV analysis for prioritizing projects was of some interest, but not essential. Not everyone understood the concept of NPV, so some additional background was required to explain discounted cash flows. There was agreement that being able to prioritize projects was valuable, but some felt that if a project saved any money it should be implemented independent of the economic analysis provided by Project99.

- The team appreciated having all of its projects consolidated into a single database, especially since the team did not have to do the work. It was not at all clear that they would have been willing to enter all the projects themselves.

Many of the monitor reduction team comments and suggestions were incorporated in the prototype. Team members were very willing to offer suggestions for enhancements, but what they really expected and wanted was very basic functionality without a great deal of complexity beyond what they were currently using. The most popular features were the easy-to-use data entry screens and the forecasting capability.

## 6.6. Pilot Program Conclusions

The pilot program was an excellent opportunity to test a cost management system in a controlled environment so that problems could be identified and fixed before introduction to an entire factory. The pilot program includes installing a working prototype on a computer network, selecting a pilot group, populating the prototype database with cost reduction projects, training users, and incorporating their feedback.

Selecting the pilot team was very important for a successful pilot program. The pilot area had high visibility to management, leaders and staff that were willing to participate, ample data that showcased features of the prototype, and potential to benefit significantly from the pilot program.

A key lesson learned was that it was probably not a good idea to attempt to port data from an old project list directly into the new database. The result was that many of the same flaws and inaccuracies from the old system were introduced into the new one.

Following a successful pilot program, the Project99 system was implemented into the factory for general use. Implementation also included propagation of Project99 to other sites with similar project management needs. Implementation is described in the next chapter.

# 7. Implementation

## 7.1. Chapter Overview

The Project99 pilot program provided feedback about necessary changes, demonstrated capability to the organization, and built credibility with management. Following the successful pilot, the system was ready for implementation throughout the organization. This chapter describes the steps involved in system implementation and discusses lessons learned from implementation within D2 and the virtual factory. Steps included delivering release versions of tools, informing stakeholders, training users, providing technical support, collecting feedback, and finally releasing the system to the organization.

## 7.2. Delivering Tools

The tools developed in the prototype phase and tested in the pilot phase required modifications and repairs before release to the entire organization. Issues like data security, multi-user capability, scalability, and technical support that were not resolved during the pilot program were revisited before a larger-scale implementation. If a prototype lacks the necessary robustness and the information technology resources exist, it may be appropriate to completely rewrite the tools using the prototype as a model only. The upside is that the finished product will be robust while retaining the functionality of the prototype. The downside is that a complete rewrite may take a long time and the program will lose the momentum gained during the pilot. Whether using the improved prototype or starting from scratch, there will always be demand for more improvements, enhancements, and changes. At some point, the design must be frozen and a firm launch date set. Additional features can always be added later.

At Intel, implementation was an iterative process. In the first iteration, the lessons learned from the pilot were incorporated into the prototype for release into the D2 environment. In the second iteration, modifications were made to accommodate virtual factory requirements.

## 7.3. Informing Stakeholders

Getting information out to the organization can be the most challenging part of implementation. In the pilot phase, the audience was much smaller and was more motivated to learn and adopt a new system. In the larger implementation, the audience had a more diverse set of needs and there was less opportunity for one-on-one conversations. Chain of command was a good way to reach a large audience.

At Intel, the bottom up approach seemed to work the best to communicate the goals of the project. First, the individual cost teams and their leaders had valuable insight beyond that which came out of the pilot program. This feedback was incorporated, making the presentation to management that much more sound. Second, the factory management was very consensus driven and respected the opinions of the cost team leaders. Factory management would likely have rejected the proposal immediately without the success of the pilot program and the endorsement of the cost team leaders. Experience suggested that unless there was nearly unanimous agreement from the members of the cost team, it would be difficult getting buy-in from the factory management.

#### 7.3.1.1. Managing Customer Expectations

After seeing the prototype demonstration, there was considerable resistance from the some of the commodity team leaders. They were expecting much different functionality than the proposed cost management system offered. Fundamentally, they were looking for a spending model rather than a savings model. In other words, they wanted a general-purpose tool that modeled all commodity costs for all tools, that would reflect cost reduction projects by lower spending for the tool. To their dismay, the system was based solely on projected savings, not actual spending. They argued that without an accurate model of how resources were consumed, there could not be an accurate measure of savings or reduced consumption. While the value of such a tool was undisputed, it was well beyond the Project99. Clearly, customer expectations were not met.

A compromise was reached whereby new features were added that helped the users develop a more accurate estimate of project savings based on consumption data (e.g., costs of spares, chemicals, test wafers, and labor) by tying in information from other Intel databases.

After incorporating the new features not only did the factory management approve of the work, but also they recommended it to the other fabs within the virtual factory. By deploying the system to the virtual factory, management argued, fabs would share the same project database and the impact of the projects would be pooled. A virtual factory project database offered the additional advantage of sharing cost-reduction success and failure among factories. The Copy EXACTLY! Strategy greatly enhanced the ability to transfer ideas from one factory to another.

## 7.4. Training Users and Providing Technical Support

Training was needed for the technicians, engineers, and managers using the system. Training sessions resulted in the following lessons learned:

- Build adequate time for training into the implementation schedule.
- Be flexible with training sessions. Accommodate the schedules of the audience.

- Use project examples that are relevant to the group when describing the system features.
- Make training interactive. Let the users interact with the system if possible.
- Listen to the feedback during training.

## 7.5. Collecting and Incorporating Feedback

Initially, the D2 cost manager planned an extended trip to visit Fab 11 in New Mexico, Fab 12 in Arizona, and Fab 15 in Oregon. The goal was to gain virtual factory acceptance for Project99, enabling improved information sharing across all factories. The most favorable reaction came from Fab 11, where a similar project was already planned. They were happy to discuss a ready-made solution that would save them many hours of development. An initial meeting with the Fab 11 cost team identified many differences of opinion on the value or functionality of certain features. There was, however, a broad agreement regarding the need for and usefulness of a shared project database.

After settling on some system design changes, Fab 11 agreed to take ownership of the database after it was completed. The remaining weeks of the internship were spent incorporating changes to make Project99 suitable for virtual factory use. In a final videoconference meeting with F11 in mid-December 1998, Project99 was officially turned over to the tool to the virtual factory.

## 7.6. Implementation Conclusions

Implementation was the ultimate test of acceptance of Project99. There was only a short window of opportunity to succeed in having the implementation last, and so great care was taken to ensure a smooth transition. The following lessons from the implementation at D2 may be of interest:

- This bottoms-up approach of delivering the message of the cost management system to the factory might not have worked in all organizations, especially if the teams had been resistant to change. It is important to match the approach to the organization.
- Communication with the cost team leaders throughout the prototype and pilot phase is absolutely necessary. By the time implementation is discusses, stakeholders should be on-board with the scope of the system.
- Plan meetings and training sessions for implementation well in advance. Book conference rooms, get on meeting agendas, and plan meetings as needed to meet the stakeholder group needs. Create and execute a detailed schedule of implementation milestones.

# 8. Recommendations

The project database described in this thesis can a useful part of an overall manufacturing cost management strategy. Used and maintained correctly, it can help an organization to:

- Develop a roadmap to meet overall cost targets.
- Manage effective risk-taking.
- Quantify and prioritize projects and resource allocation.
- Track and manage the effective execution of projects.
- Convey progress in a meaningful way to technicians, engineers, and management.

The following recommendations are presented in the context of the D2 factory and the Intel virtual factory organization, but the concepts can be applied to any manufacturing organization. The recommendations progress from the more tactical to the more strategic:

- *Develop accurate models that reflect the fixed and variable costs associated with operating a factory.* The first step toward creating an effective cost management system should be to create a cost model for the major cost drivers within an organization. Given the appropriate level of resources and management attention, a model could be created that reflects actual spending. This model would provide a more meaningful baseline to which a cost project could be measured against. The Bottoms Up Estimate created to estimate spares consumption on a tool area was an excellent first-order cost model. These models could be extended to capture all commodity consumption (e.g., chemicals, labor) and extended to cover all tools in a functional area (e.g., cover all tools in the lithography area under a single model).
- *Develop a cross-functional cost reduction organization.* The cost reduction teams in the D2 was organized by commodity, while the management of the factory was organized by functional area. This provided an excellent opportunity to share learnings from one functional area to another and one commodity group to another. Care was taken to ensure that information flows freely throughout the matrix. For example, weekly functional area teem meetings could feature a success story from a different commodity team member each week.
- *Establish cost metrics to which everyone can relate.* Intel chose to use an average $/wafer start as the primary metric by which it measured cost performance. Costs were broken out by product, by functional area, and by commodity consumption in terms of a single, well-understood metric. When

there is not an obvious relationship between consumption of a resource (such as labor) and the cost metric, create a logical conversion and apply it universally.

- *Balance cost with other factory metrics.* A key feature of the Project99 tool was the ability to capture cost savings as well as yield and cycle time in an NPV analysis of a project. Shift managers at Intel had other metrics to consider as well, such as the percentage of machine up-time, the amount of overtime consumed, and training. An effective cost management system balances cost with other factory metrics in a fair and consistent manner.
- *Align the incentives of all employees.* Care should be taken to ensure that incentives should be aligned across the organization, and employees do not work against each other in order to sub-optimize to their own individual incentive structure.
- *Encourage and practice calculated risk-taking.* From the perspective of many engineers, Intel was leaving many great cost reduction opportunities untapped due to a risk-averse mentality. By creating tools and frameworks for performing a quantitative risk-benefit analysis and making them available to those closest to the manufacturing process, great gains may be achieved. Some organizations celebrate and reward projects that were great failures as long as the reasoning for undertaking the project was sound.
- *Introduce cost reduction early in the product life-cycle.* By the time a semiconductor manufacturing process reaches high-volume production, many of the key cost drivers are already firmly established. Nevertheless, D2 was generally trailing the rest of the virtual factory in terms of $/ws in nearly every category. This was in part do to the economies of scale at the larger facilities and the development burden at D2. Still, much of the cost-cutting innovation flowed from the high-volume fabs back to D2. Cost should be considered early in the manufacturing process development, ensuring that the process that it exported to the high-volume factories is cost effective before it arrives. Cost should be weighted appropriately along with yield as a precursor to production release.

To one degree or another, all of the above recommendations were reflected in the Project99 system. Every effort was made to strike a balance between providing a practical tool that could be applied immediately in the factory environment and creating a gentle lever to encourage a change in the way decisions were made to better match the long-term goals of the organization.