Radu Prodan Thomas Fahringer

Grid Computing

Experiment Management, Tool Integration, and Scientific Workflows

Springer

Contents

1 Introduction 1
- 1.1 Motivation 2
  - 1.1.1 Performance Tuning 2
  - 1.1.2 Parameter Studies 3
  - 1.1.3 Optimisation 3
  - 1.1.4 Scheduling 3
  - 1.1.5 Parametrisation Language 4
  - 1.1.6 Instrumentation 4
  - 1.1.7 Portability 5
  - 1.1.8 Tool Interoperability 5
  - 1.1.9 Grid Services 5
  - 1.1.10 Scientific Workflows 6
- 1.2 Goals 6
  - 1.2.1 Experiment Specification Language 6
  - 1.2.2 Experiment Management Tool 7
  - 1.2.3 Optimisation 8
  - 1.2.4 Scientific Workflows 8
  - 1.2.5 Service-Oriented Grid Architecture 9
  - 1.2.6 Grid Services 10
- 1.3 Outline 10

2 Model 13
- 2.1 Introduction 13
- 2.2 Distributed Technology History 14
- 2.3 Web Services 15
  - 2.3.1 Web Services Stack 16
  - 2.3.2 Web Services Runtime Environment 18
- 2.4 Grid Security Infrastructure 19
- 2.5 Globus Toolkit 20
- 2.6 Grid Architectural Model 22
  - 2.6.1 Machine Layer 22
  - 2.6.2 Grid Services Layer 28
  - 2.6.3 Application Layer 30
- 2.7 Summary 35

3 The ZEN Experiment Specification Language 37
3.1 Functionality and Use Cases 37
3.1.1 Shared Memory Application Scalability 38
3.1.2 ZEN Transformation System 39
3.1.3 Shared Memory Loop Scheduling 40
3.1.4 Distributed Processor Arrays 41
3.1.5 Distributed Memory Arrays 41
3.1.6 Work Distribution 43
3.1.7 Parameter Studies 43
3.2 Formal Language Specification 44
3.2.1 ZEN Set 44
3.2.2 ZEN Directives 49
3.2.3 ZEN Substitute Directive 50
3.2.4 Local Substitute Directive 51
3.2.5 Homonym ZEN Variables 51
3.2.6 ZEN Assignment Directive 53
3.2.7 Multi-dimensional Value Set 54
3.2.8 ZEN Constraint Directive 55
3.2.9 ZEN Performance Directive 59
3.2.10 Parameter Study Experiment 62
3.2.11 Experiment Generation Algorithm 62
3.2.12 Online Monitoring and Analysis 65
3.3 Summary 68

4 ZENTURIO Experiment Management Tool 69
4.1 User Portal Functionality 69
4.1.1 ZEN Editor 70
4.1.2 Experiment Preparation 71
4.1.3 Experiment Monitor 73
4.1.4 Application Data Visualiser 73
4.2 Performance Studies 76
4.2.1 Ocean Simulation 76
4.2.2 Linearised Augmented Plane Wave 79
4.2.3 Three-Dimensional Particle-in-Cell 84
4.2.4 Benders Decomposition 86
4.2.5 Three-Dimensional FFT Benchmarks 89
4.3 Parameter Studies 94
4.3.1 Backward Pricing 94
4.4 Architecture 105
4.4.1 Experiment Generator 107
4.4.2 Experiment Executor 108
4.4.3 Experiment State Transition Diagram 110
4.4.4 Experiment Data Repository 110
4.5 Summary 111

5 Tool Integration 113
5.1 Architecture 114
5.2 Interoperable Tool Set 116
5.2.1 Object Code Browser 117
5.2.2 Function Profiler (Z_prof) 117
5.2.3 Function Tracer (Z_trace) 118
5.2.4 Function Coverager (Z_cov) 119
5.2.5 Sequential Debugger (Z_debug) 121
5.2.6 Memory Allocation Tool (Z_MAT) 121
5.2.7 Resource Tracker (Z_RT^2) 122
5.2.8 Deadlock Detector (Z_deadlock) 122
5.3 Tool Interoperability 122
5.3.1 Classification 122
5.3.2 Interaction with a Browser 123
5.3.3 Performance Steering 124
5.3.4 Just-in-Time Debugging 126
5.3.5 Interaction with a Debugger 127
5.4 The Monitoring Layer 128
5.4.1 Dynamic Instrumentation 128
5.4.2 The Process Manager 130
5.4.3 Dynamic Instrumentation of MPI Applications 134
5.5 The Grid Services Layer 136
5.5.1 Web Application and Services Platform (WASP) 138
5.5.2 Service Repository 139
5.5.3 Abstract Grid Service 140
5.5.4 Factory 142
5.5.5 Registry 143
5.5.6 WSDL Compatibility 144
5.5.7 Dynamic Instrumentor 144
5.5.8 Aggregator 145
5.6 Event Framework 146
5.6.1 Representation 146
5.6.2 Implementation 149
5.6.3 Filters 151
5.7 Firewall Management 151
5.8 WASP Versus GT3 Technology Evaluation 152
5.8.1 Stub Management 153
5.8.2 Service Lifecycle 154
5.8.3 UDDI-Based Service Repository 155
5.8.4 Service Data 155
5.8.5 Events 155
5.8.6 Registry 156
5.8.7 Security 159

5.8.8 Grid Service Throughput 160
5.8.9 Comparison 163
5.9 Summary 164

6 Optimisation Framework 165
6.1 Workflow Scheduling 167
6.1.1 Schedule Dependencies 169
6.1.2 Objective Function 170
6.2 Genetic Search Engine 174
6.2.1 Initial Population 175
6.2.2 Selection 177
6.2.3 Crossover 177
6.2.4 Mutation 178
6.2.5 Elitist Model 178
6.2.6 Fitness Scaling 179
6.2.7 Convergence Criterion 180
6.3 Genetic Workflow Scheduling 180
6.3.1 WIEN2k 180
6.4 Throughput Scheduling 192
6.5 Performance Tuning of Parallel Applications 194
6.5.1 Parallel Applications on the Grid 195
6.6 Summary 201

7 Scientific Grid Workflows 203
7.1 Workflow Model 204
7.1.1 Computational Activity 205
7.1.2 Control Flow Dependencies 206
7.1.3 Data Flow Dependencies 207
7.1.4 Conditional Activity 207
7.1.5 Parallel Loop Activity 208
7.1.6 Sequential Loop Activity 211
7.1.7 Workflow Activity 213
7.2 Scheduler 214
7.2.1 Workflow Converter 214
7.2.2 Scheduling Engine 220
7.2.3 Layered Partitioning 226
7.2.4 WIEN2k 227
7.2.5 Invmod 231
7.3 Enactment Engine 235
7.3.1 Workflow Partitioning 236
7.3.2 Control Flow Management 241
7.3.3 Data Flow Management 242
7.3.4 Virtual Single Execution Environment 243
7.3.5 Workflow Steering 244
7.3.6 Fault Tolerance 248

7.3.7 WIEN2k Execution Experiments 253
7.3.8 Steering Experiments 255
7.4 Overhead Analysis 260
7.4.1 Experiments 263
7.5 Summary 269

8 Related Work 271
8.1 Experiment Management 271
8.2 Performance Study 271
8.3 Parameter Study 273
8.4 Optimisation and Scheduling 273
8.5 Tool Integration 274
8.5.1 Scientific Workflows 276

9 Conclusions 279
9.1 Contributions 279
9.1.1 Experiment Specification 279
9.1.2 Experiment Management 280
9.1.3 Optimisation 281
9.1.4 Tool Integration Design 281
9.1.5 Web Services for the Grid 283
9.1.6 Scientific Workflows 283

10 Appendix 285
10.1 Notations 285
10.2 Code Regions 288
10.3 Abbreviations 289
10.4 Performance Metrics 292

References 297

Index 311