Configuration Management
Principles and Practice
The Agile Software Development Series
Alistair Cockburn and Jim Highsmith, Series Editors
For more information check out http://www.awprofessional.com/

Agile software development centers on four values identified in the Agile Alliance’s Manifesto:

- Individuals and interactions over processes and tools
- Working software over comprehensive documentation
- Customer collaboration over contract negotiation
- Responding to change over following a plan

The development of Agile software requires innovation and responsiveness, based on generating and sharing knowledge within a development team and with the customer. Agile software developers draw on the strengths of customers, users, and developers, finding just enough process to balance quality and agility.

The books in The Agile Software Development Series focus on sharing the experiences of such Agile developers. Individual books address individual techniques (such as Use Cases), group techniques (such as collaborative decision making), and proven solutions to different problems from a variety of organizational cultures. The result is a core of Agile best practices that will enrich your experience and improve your work.

Titles in the Series:

Alistair Cockburn, Surviving Object-Oriented Projects, ISBN 0-201-49834-0
Alistair Cockburn, Writing Effective Use Cases, ISBN 0-201-70225-8
Steve Adolph, Paul Bramble, Alistair Cockburn, and Andy Pols, Patterns for Effective Use Cases, ISBN 0-201-72184-8
Contents

List of Figures xxv
List of Tables xxix
Foreword by Kim Caputo xxxi
Foreword by Alistair Cockburn xxxiii
Preface xxxv
Introduction xli

Part I What Is Configuration Management? ................. 1

Chapter 1 Definition of Configuration Management Used in This Book ......................... 3

1.1 Configuration Management Activities ....................... 4
   Metadata ................................................. 5
   Configuration Management Is Cyclic—or Is It? .................. 5
   Quality Assurance Process .................................. 6
   Audit ..................................................... 7

1.2 Identification ............................................. 7
   Inputs ................................................... 8
   Outputs .................................................. 8
   Process Descriptions ..................................... 9
   Unique Identification .................................... 9
<table>
<thead>
<tr>
<th>Section</th>
<th>Pages</th>
</tr>
</thead>
<tbody>
<tr>
<td>Examples</td>
<td>9</td>
</tr>
<tr>
<td>Authorization</td>
<td>11</td>
</tr>
<tr>
<td>Roles</td>
<td>11</td>
</tr>
<tr>
<td>Connection with Other Activities</td>
<td>11</td>
</tr>
<tr>
<td>1.3 Storage</td>
<td>12</td>
</tr>
<tr>
<td>Library</td>
<td>13</td>
</tr>
<tr>
<td>Main Processes</td>
<td>15</td>
</tr>
<tr>
<td>Process Descriptions</td>
<td>16</td>
</tr>
<tr>
<td>Roles</td>
<td>16</td>
</tr>
<tr>
<td>Connection with Other Activities</td>
<td>17</td>
</tr>
<tr>
<td>Example</td>
<td>17</td>
</tr>
<tr>
<td>1.4 Change Control</td>
<td>19</td>
</tr>
<tr>
<td>Inputs</td>
<td>20</td>
</tr>
<tr>
<td>Outputs</td>
<td>20</td>
</tr>
<tr>
<td>Change Control Activities</td>
<td>20</td>
</tr>
<tr>
<td>Usage of Metadata</td>
<td>21</td>
</tr>
<tr>
<td>Consequence Analysis</td>
<td>22</td>
</tr>
<tr>
<td>Roles</td>
<td>22</td>
</tr>
<tr>
<td>Process Descriptions</td>
<td>23</td>
</tr>
<tr>
<td>Connection with Other Activities</td>
<td>23</td>
</tr>
<tr>
<td>Example</td>
<td>23</td>
</tr>
<tr>
<td>1.5 Status Reporting</td>
<td>23</td>
</tr>
<tr>
<td>Inputs</td>
<td>25</td>
</tr>
<tr>
<td>Outputs</td>
<td>25</td>
</tr>
<tr>
<td>Process Descriptions</td>
<td>25</td>
</tr>
<tr>
<td>Roles</td>
<td>26</td>
</tr>
<tr>
<td>Connection with Other Activities</td>
<td>26</td>
</tr>
<tr>
<td>1.6 False Friends: Version Control and Baselines</td>
<td>26</td>
</tr>
<tr>
<td>Version Control</td>
<td>27</td>
</tr>
<tr>
<td>Baseline</td>
<td>27</td>
</tr>
</tbody>
</table>

**Chapter 2**  
*Configuration Management in Maturity Models*  

2.1 CMM Version 1.1 | 29 |
| CMM Maturity Levels | 30 |
| Definition | 31 |
| Activities | 31 |
## 2.2 CMMI

- CMMI Process Areas .............................................................. 34
- Definition .............................................................................. 35
- Goals ...................................................................................... 35
- Practice-to-Goal Relationships ............................................ 35
- Capability and Maturity Levels ............................................. 36
- Achieving Capability Levels .................................................. 37
- Level 2 for All Process Areas .................................................. 37
- Raising the Capability of the Configuration Management Process ................................................................................. 38

## 2.3 ISO 15504 (SPICE) and BOOTSTRAP 3.2

- SPICE Process Model .............................................................. 40
- Definition .............................................................................. 40
- Goals ...................................................................................... 40
- Best Practices ........................................................................ 42
- Maturity Levels ...................................................................... 42
- Maturity of Configuration Management ................................ 43

## Chapter 3 Configuration Management in International Standards ......................................................... 45

### 3.1 Overview of Related Standards ................................................. 45

- BS6488, DoD, IEEE ................................................................. 47
- BS6488 .................................................................................... 47
- DoD Mil-Std-973 ..................................................................... 48
- IEEE-Std-610.12-1990 ............................................................. 48

### 3.2 BS6488, DoD, IEEE ................................................................. 47

- BS6488 .................................................................................... 47
- DoD Mil-Std-973 ..................................................................... 48
- IEEE-Std-610.12-1990 ............................................................. 48

### 3.3 ESA PSS-05-09 ................................................................. 48

- Introduction from the Guide ...................................................... 49

### 3.4 GAMP ................................................................. 49

- Description from the Guide ...................................................... 50


- ISO 9001:1994 ........................................................................ 50
- ISO 9000-3 ............................................................................. 51
- ISO 9001:2000 ....................................................................... 52

## Chapter 4 Organizations Working with Configuration Management ......................................................... 55

### 4.1 Institutions and Companies ................................................. 55

- CM Today Yellow Pages .......................................................... 55
## Part II  Configuration Management Data  .............. 75

### Chapter 6  What Can Be Placed under Configuration Management  . . . . . . . . . . . . . . . . . . . . . . . 77

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>6.1 Physical or Electronic Objects</td>
<td>77</td>
</tr>
<tr>
<td>Configuration Item Class Hierarchy</td>
<td>77</td>
</tr>
<tr>
<td>Physical Objects</td>
<td>78</td>
</tr>
<tr>
<td>Electronic Objects</td>
<td>79</td>
</tr>
<tr>
<td>6.2 Types of Objects in Product Perspective</td>
<td>79</td>
</tr>
<tr>
<td>Software</td>
<td>79</td>
</tr>
<tr>
<td>Hardware</td>
<td>80</td>
</tr>
<tr>
<td>Network</td>
<td>80</td>
</tr>
<tr>
<td>Data</td>
<td>80</td>
</tr>
<tr>
<td>Services</td>
<td>80</td>
</tr>
<tr>
<td>Tools</td>
<td>81</td>
</tr>
<tr>
<td>6.3 Types of Objects in Project Perspective</td>
<td>81</td>
</tr>
<tr>
<td>Life Cycle Activities</td>
<td>81</td>
</tr>
<tr>
<td>Support Functions</td>
<td>82</td>
</tr>
<tr>
<td>Tools</td>
<td>82</td>
</tr>
<tr>
<td>6.4 Types of Objects in Cross-Organizational Perspective</td>
<td>82</td>
</tr>
<tr>
<td>Cross-Organizational Perspective</td>
<td>82</td>
</tr>
<tr>
<td>Administrative Documents</td>
<td>82</td>
</tr>
<tr>
<td>Company Product Assets</td>
<td>82</td>
</tr>
<tr>
<td>Infrastructure</td>
<td>83</td>
</tr>
<tr>
<td>Quality System</td>
<td>83</td>
</tr>
<tr>
<td>6.5 Deliveries under Configuration Management</td>
<td>83</td>
</tr>
<tr>
<td>Examples</td>
<td>84</td>
</tr>
<tr>
<td>Project Relationships</td>
<td>84</td>
</tr>
<tr>
<td>6.6 Deliveries for Planned Events Like Milestones</td>
<td>85</td>
</tr>
<tr>
<td>Development Model</td>
<td>85</td>
</tr>
<tr>
<td>Milestones</td>
<td>85</td>
</tr>
</tbody>
</table>

### Chapter 7  What One Needs to Know about a Configuration Item  . . . 89

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>7.1 Overview of Metadata for a Configuration Item</td>
<td>89</td>
</tr>
<tr>
<td>Data Elements</td>
<td>89</td>
</tr>
<tr>
<td>Metadatabase Medium</td>
<td>90</td>
</tr>
</tbody>
</table>
Contents

Other Data Elements .................................................. 91

7.2 Metadata for Unique Identification .................................. 91
Belongs To ................................................................. 91
Name ................................................................. 91
Version ................................................................. 93
Status ................................................................. 93
Date ................................................................. 93
Storage Location ...................................................... 94
Storage Medium ....................................................... 94
Example of States for a Document .................................. 94
Example of States for a Source Code Unit, Including in Build ......... 94

7.3 Metadata for Authorization ........................................... 96
Producer ................................................................. 96
Person Holding Overall Responsibility ................................ 97
Person Responsible for Approval ..................................... 97
Ownership ............................................................. 97

7.4 Metadata for Relations to Other Configuration Items .............. 97
Traces To (and From!) .................................................... 98
Tracing Registration ..................................................... 99
Importance of Tracing .................................................. 99
Produced With .......................................................... 100
Derived From .......................................................... 100
Consists Of .............................................................. 100

7.5 Metadata for Distribution ............................................. 101
May Be Distributed To .................................................. 101
Has Been Distributed To ................................................ 101

Chapter 8 What One Must Register for a Configuration Item ......... 103

8.1 Item Approval .......................................................... 103
Quality Approval ........................................................ 103
Medium ................................................................. 104
Content ................................................................. 104
Examples .............................................................. 105

8.2 Release Request ........................................................ 106
Medium ................................................................. 106
Content ................................................................. 108
Stock Control .......................................................... 109
8.3 **Event Registration** ........................................ 111
   Life Cycle and Responsibility ................................ 111
   Content .................................................................. 112
   Created ................................................................ 113
   For Evaluation ..................................................... 113
   Under Decision ..................................................... 113
   Under Change ....................................................... 115
   Closed .................................................................. 115
   Classification ....................................................... 115
   Examples ............................................................. 115

8.4 **Change Request** ............................................. 116
   Life Cycle and Responsibility .................................. 119
   Content .................................................................. 119
   Created ................................................................ 120
   Implemented ....................................................... 120
   Approved ............................................................ 121
   New Events ......................................................... 121
   Examples ............................................................. 122

**Chapter 9**  *What Information Is Available for Configuration Items*  ........................................ 125

9.1 **Examples** ..................................................... 125
   Release Note ........................................................ 125
   Item Status List ...................................................... 127
   Item History List .................................................... 127
   Item Composition List ............................................ 127
   Trace Report ........................................................ 130

9.2 **Configuration Management as Supplier of Measurements** ........................................... 132
   Ideas for Process Improvement ................................. 133

**Part III**  *Roles in Configuration Management* ......................................................... 135

**Chapter 10**  *People and Configuration Management* .................................................... 137

10.1 **Configuration Management as a Career** .............................................................. 137
   Qualifications ........................................................ 138
10.2 **Managing Configurations Is Everyone’s Job** ....................................................... 138
Chapter 11  Configuration Management Roles .......................... 141

11.1  Configuration Control Board ............................. 141
Skills and Knowledge ........................................ 142
Multiple Boards ............................................. 143
Managing Configurations of CCB Work Products ............. 143
References ................................................... 143

11.2  Librarian ..................................................... 144
Tools ............................................................. 146
Managing Configuration of Library Work Products .......... 146
References ................................................... 146

11.3  Person Responsible for Configuration Management ........ 146
Planning Configuration Management ................................ 148
Managing Configuration Management Work Products ...... 148
References ................................................... 148

Chapter 12  Organizational Roles .................................. 149

12.1  Management .................................................. 149
Defining and Tracking Goals .................................. 150
Benefits ....................................................... 150
References ................................................... 150

12.2  Person Responsible for Assets .............................. 151
Different Process Descriptions ................................ 151
References ................................................... 152

12.3  Person Responsible for Operation .......................... 152
Configuration Management Responsibility .................... 152
References ................................................... 153

12.4  Person Responsible for Process Management ............... 153
Managing Configurations of Process Management Work Products 154
References ................................................... 155

12.5  Person Responsible for Environments and Tools ............ 155
Managing Configurations of Environments and Tools ....... 156
References ................................................... 156
## Chapter 13  Project-Related Roles

12.6 Support/Helpdesk .................................................. 156
References .......................................................... 157

### 13.1 Analyst
Benefits ................................................................. 159
References ........................................................ 160

### 13.2 Designer
Benefits ................................................................. 160
References ........................................................ 161

### 13.3 Programmer
Benefits ................................................................. 161
References ........................................................ 162

### 13.4 Integrator
Benefits ................................................................. 162
References ........................................................ 163

### 13.5 Tester
Benefits ................................................................. 163
References ........................................................ 164

### 13.6 Project Manager
Benefits ................................................................. 164
Managing Configurations of Project Management Work Products 165
References ........................................................ 165

### 13.7 Person Responsible for Quality
Managing Configurations of Quality Assurance Work Products. 166
References ........................................................ 166

### 13.8 Person Responsible for Customer Contact
References .......................................................... 167

### 13.9 Person Responsible for Subcontractor Contact
References .......................................................... 168

## Chapter 14  External Roles

14.1 Customer .......................................................... 169
References .......................................................... 170

14.2 Subcontractor ...................................................... 171
References .......................................................... 171
Part IV  Configuration Management in Practice ........ 173

Chapter 15  General Principles ......................... 175

15.1 Milestones ............................................. 175
  Identification ........................................... 175
  Generic Content Lists ................................. 177
  Storage .................................................. 177
  Change Control ........................................ 178
  Status Reporting ...................................... 179

15.2 Document Handling .................................... 179
  Configuration Items or Deliveries .................... 179
  Identification ........................................... 180
  Authorization .......................................... 182
  Tracing .................................................. 182
  Storage .................................................. 183
  Change Control ........................................ 183
  Status Reporting ...................................... 183

15.3 Emergency Changes .................................... 183
  Examples ............................................... 184
  Principles for “Cheating” ............................. 184
  Avoid Cheating ........................................ 185
  Examples Again ....................................... 185

Chapter 16  Configuration Management in Development Activities ... 187

16.1 Documentation Activities (Specifications and Design) ............... 187
  Identification ........................................... 188

16.2 Coding ................................................. 189
  Unique Identification .................................. 189
  Authorization .......................................... 190
  Tracing .................................................. 190
  Storage .................................................. 190
  Change Control ........................................ 190

16.3 Integration ............................................. 191
  Production Time ....................................... 191
  Unique Identification .................................. 191
  Tracing .................................................. 192
Chapter 18  Managing Configurations in Different Development Models  ..........  207

18.1 Agile Development ............................................. 207
Configuration Management in Agile Development ................. 208
Empowered Teams .............................................. 208
Process Handling .............................................. 209
Environment and Support ........................................... 209
Requirements Management ........................................... 210
Working Together ............................................. 210
Frequent Delivery of Working Software ................................. 210
Communication and Documentation .................................. 211
Status Reporting .................................................. 211

18.2 Frequent-Build Technique ........................................... 211
Planning Considerations ............................................. 212
Configuration Management Considerations ......................... 212
Frequent Builds Are Not Frequent Storage ......................... 212
Identification .................................................. 213
Building .................................................. 213
Storage .................................................. 213
Backtracking ............................................... 213
Change Control ............................................... 214
Example .................................................. 214

18.3 Integrated Product Development ................................... 216
Organizational Considerations ........................................ 217
Configuration Management Considerations ......................... 217
Approach .................................................. 217

18.4 Iterative Development ............................................... 218
Configuration Management Considerations ......................... 218
Requirements Management ........................................... 219
Identification .................................................. 220
Storage .................................................. 220
## 18.5 Sequential Development

- Change Control: 220
- Status Reporting: 220

### W-Model

- Configuration Management Considerations: 222
- Identification: 224
- Change Control: 224
- Status Reporting: 224

## Chapter 19 Managing Configurations for Different Product Types

- **19.1 Composite Systems**
  - Design Considerations: 225
  - Configuration Management Considerations: 226
  - Identification: 227
  - Storage: 227
  - Change Control: 227
  - Status Reporting: 227

- **19.2 Multiplatform**
  - Configuration Management Considerations: 227
  - Identification: 228

- **19.3 Multivariants**
  - Examples: 228
  - Requirements Considerations: 229
  - Design Considerations: 229
  - Configuration Management Considerations: 230
  - Identification: 230
  - Storage: 231
  - Change Control: 231
  - Status Reporting: 231

- **19.4 Safety-Critical Products**
  - Examples: 232
  - Configuration Management Considerations: 233

- **19.5 Size of Product (Large and Small)**
  - Small Systems: 233
  - Large Systems: 233
  - Identification: 234
  - Storage: 234
20.4 Tool Support ................................................. 249
  Configuration Management Considerations .......... 249

Chapter 21 Managing Configurations for
Cross-Organizational Functions ......................... 251

21.1 Company Infrastructure .................................. 251
  Organizational Considerations ......................... 251
  Identification ............................................ 252
  Storage .................................................. 252
  Change Control ....................................... 252
  Status Reporting ..................................... 253

21.2 Cross-Organizational Objects ......................... 253
  Configuration Management Considerations .......... 253
  Identification ........................................ 253
  Storage ................................................. 254
  Change Control ..................................... 254
  Status Reporting ................................... 254

21.3 External Reuse Component Development ............ 254
  Examples .............................................. 255
  Configuration Management Considerations .......... 255
  Identification ........................................ 255
  Storage ............................................... 255
  Change Control .................................... 256
  Status Reporting ................................... 256

21.4 Internal Asset Development (Product-Line Approach) 257
  Examples .............................................. 258
  Central Ownership of Components ................... 258
  Configuration Management Considerations .......... 258
  Identification ........................................ 259
  Storage ............................................... 259
  Change Control .................................... 260
  Status Reporting ................................... 261

21.5 Quality System, Including Process Management .... 261
  Configuration Management Considerations .......... 262
  Responsibility ....................................... 262
  Identification ....................................... 262
  Storage ............................................... 262
## Contents

Change Control ................................................................. 263
Status Reporting ............................................................ 263

**Part V  Improving Configuration Management ........ 265**

Capability Levels .......................................................... 265

**Chapter 22  Getting Started on Configuration Management—
up to Capability Level 1 ........................................ 267**

22.1 How to Get Started from Nothing ......................... 267
  Getting the Right People ............................................ 267
  Collecting Best Practices Internally .............................. 268
  Looking at the Outside World ..................................... 268
  Focus .................................................................. 268
  Look Ahead ............................................................... 269

22.2 First Steps Toward Configuration Management ......... 269
  Establish Baselines ...................................................... 269
  Track and Control Changes .......................................... 270
  Minimum Documentation ............................................ 271
  Establish Integrity ....................................................... 271

22.3 Experiences in Implementing Configuration Management ........ 272
  Overall Conclusion .................................................... 272
  Datamat Ingegneria dei Sistemi .................................... 272
  S.I.A. S.p.A. .............................................................. 273
  Istiservice, S.p.a. ....................................................... 274
  Event A/S ................................................................ 275
  Sysdeco A/S .............................................................. 276

**Chapter 23  Planning Configuration Management—
up to Capability Level 2 ......................... 277**

23.1 General Planning Advice ........................................... 277
  The Plan Itself ............................................................ 278
  Connection to the Project ............................................ 278
  Template ................................................................. 279

23.2 Table of Contents for a Configuration Management Plan ........ 279
## 23.3 Configuration Management Plan: Introduction
- **Purpose**
- **Scope**
- **Vocabulary and Reference Lists**

## 23.4 Configuration Management Plan: Management and Relations to the Environment
- **Organization**
- **Responsibilities**
- **Interface Control**
- **Subcontractor Management**
- **Relevant Standards**

## 23.5 Configuration Management Plan: Activities
- **Identification**
- **Storage**
- **Change Control**
- **Status Reporting**

## 23.6 Configuration Management Plan: Schedule
- **Tasks**
- **Phases and Milestones**
- **Diagrams and Charts**

## 23.7 Configuration Management Plan: Tools, Techniques, and Methods
- **Tools**
- **Techniques and Methods**

## Chapter 24  Processes for Configuration Management—up to Capability Level 3
- **24.1 Processes in General**
- **Connection with Maturity Models**
- **Definitions**
- **A Process Is Like a Recipe**
- **Process Model**
- **24.2 Configuration Management Processes—Overview**
- **Special Requirements for Configuration Management Processes**
- **24.3 Configuration Management Process—Model Examples**
### Chapter 25  Continuous Improvement of Configuration Management—up to Capability Level 4 and 5

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>25.1 General Software Process Improvement Advice</td>
<td>295</td>
</tr>
<tr>
<td>Processes in Use</td>
<td>296</td>
</tr>
<tr>
<td>Dissemination and Adaptation</td>
<td>296</td>
</tr>
<tr>
<td>Companies at Capability Levels 4 and 5</td>
<td>297</td>
</tr>
<tr>
<td>25.2 Metrics for Controlling Configuration Management Performance</td>
<td>297</td>
</tr>
<tr>
<td>Metrics in General</td>
<td>298</td>
</tr>
<tr>
<td>Measuring Methods</td>
<td>300</td>
</tr>
<tr>
<td>Measurement Plan</td>
<td>300</td>
</tr>
<tr>
<td>Examples</td>
<td>300</td>
</tr>
<tr>
<td>25.3 Analyzing Metrics for Control and Improvement</td>
<td>301</td>
</tr>
<tr>
<td>Statistics</td>
<td>302</td>
</tr>
<tr>
<td>Balance Point</td>
<td>302</td>
</tr>
<tr>
<td>Variation—What Is Normal</td>
<td>303</td>
</tr>
<tr>
<td>Control Charts</td>
<td>303</td>
</tr>
</tbody>
</table>

### Chapter 26  Tool Support for Configuration Management

<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>26.1 Classes of Tools for Configuration Management</td>
<td>306</td>
</tr>
<tr>
<td>Individual Support</td>
<td>306</td>
</tr>
<tr>
<td>Project-Related Support</td>
<td>307</td>
</tr>
<tr>
<td>Full, Company-Wide Process Support</td>
<td>307</td>
</tr>
<tr>
<td>Who Should Use Which Tool?</td>
<td>308</td>
</tr>
<tr>
<td>26.2 Organizational Considerations</td>
<td>308</td>
</tr>
<tr>
<td>Business Goals</td>
<td>308</td>
</tr>
<tr>
<td>Buy It or Do It Yourself</td>
<td>309</td>
</tr>
<tr>
<td>Environmental Constraints</td>
<td>309</td>
</tr>
<tr>
<td>Legacy from the Past</td>
<td>309</td>
</tr>
<tr>
<td>Financing</td>
<td>310</td>
</tr>
<tr>
<td>Organizational Scope</td>
<td>310</td>
</tr>
<tr>
<td>Ownership</td>
<td>310</td>
</tr>
<tr>
<td>Planning for the Future</td>
<td>310</td>
</tr>
<tr>
<td>Willingness to Change</td>
<td>311</td>
</tr>
<tr>
<td>26.3 Selecting a Configuration Management Tool</td>
<td>311</td>
</tr>
<tr>
<td>Evaluation Group</td>
<td>311</td>
</tr>
</tbody>
</table>
Contents

Evaluation Method ......................................................... 311
Requirements ................................................................. 312
Detailed Evaluation ....................................................... 312
Nomination of the Winner ............................................. 312

26.4 Requirements for Configuration Management Tools .......... 312
Integration with Other Tools ............................................ 313
Performance ................................................................. 313
Scalability ........................................................................ 313
Usability ......................................................................... 314
Web Access ..................................................................... 314

26.5 Requirements for the Tool Supplier ................................ 314
Acquaintances .................................................................. 314
Employees ...................................................................... 314
Financial Status .............................................................. 315
Focus .............................................................................. 315
Tool Use ......................................................................... 315
Reputation ..................................................................... 315
Support Facilities ............................................................ 316

26.6 Customizing Configuration Management Tools ................. 316
One Tool or More ............................................................ 316
Changing Tools or Processes .......................................... 317
From Class to Class ........................................................ 317

Appendix A Configuration Management Process Model:
A Software Code Example ............................................ 319

Appendix B Configuration Management Process Model:
A Tracing Example ....................................................... 333

Appendix C Agile SCM ...................................................... 343

Glossary ......................................................................... 349
Bibliography .................................................................... 351
Index ............................................................................. 357
List of Figures

Figure I–1  Generic Development Model ........................................ xliii
Figure 1–1  Overview of Configuration Management Activities .......... 4
Figure 1–2  Configuration Item Class and Instantiations ................. 6
Figure 1–3  The Life of a Configuration Item Class ......................... 6
Figure 1–4  Identification in Context ........................................... 8
Figure 1–5  Document Front Page .............................................. 10
Figure 1–6  Test Cases .......................................................... 11
Figure 1–7  Storage in Context ................................................ 12
Figure 1–8  Example Library Structure ...................................... 18
Figure 1–9  Change Control in Context ..................................... 19
Figure 1–10 Change Control Process Diagram ............................ 24
Figure 1–11 Status Reporting in Context .................................... 25

Figure 2–1  CMM Version 1.1 Maturity Levels ............................. 30
Figure 2–2  SPICE Process Area Model ................................... 41
Figure 2–3  SPICE Maturity Levels .......................................... 43

Figure 5–1  Configuration Management Cost for One Item ............ 60
Figure 5–2  Configuration Management Total Cost ...................... 63

Figure 6–1  Configuration Item Class Hierarchy .......................... 78
Figure 6–2  Requirement Specification Delivery ........................... 84
Figure 6–3  Hardware-Related Delivery .................................... 84
List of Figures

<table>
<thead>
<tr>
<th>Figure</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Figure 7–1</td>
<td>Overview of Metadata</td>
<td>90</td>
</tr>
<tr>
<td>Figure 7–2</td>
<td>Metadata for Unique Identification</td>
<td>92</td>
</tr>
<tr>
<td>Figure 7–3</td>
<td>Metadata for Authorization</td>
<td>96</td>
</tr>
<tr>
<td>Figure 7–4</td>
<td>Metadata for Relations to Other Configuration Items</td>
<td>98</td>
</tr>
<tr>
<td>Figure 7–5</td>
<td>Example of Tracing</td>
<td>99</td>
</tr>
<tr>
<td>Figure 7–6</td>
<td>Metadata for Distribution</td>
<td>101</td>
</tr>
<tr>
<td>Figure 8–1</td>
<td>Item Approval in Context</td>
<td>104</td>
</tr>
<tr>
<td>Figure 8–2</td>
<td>Mail Message Used as Approval Registration</td>
<td>106</td>
</tr>
<tr>
<td>Figure 8–3</td>
<td>Unit Approval Form—High Degree of Formalism</td>
<td>107</td>
</tr>
<tr>
<td>Figure 8–4</td>
<td>Release Request in Context</td>
<td>108</td>
</tr>
<tr>
<td>Figure 8–5</td>
<td>Mail Message Used as Release Registration</td>
<td>109</td>
</tr>
<tr>
<td>Figure 8–6</td>
<td>Release Request Form—High Degree of Formalism</td>
<td>110</td>
</tr>
<tr>
<td>Figure 8–7</td>
<td>Excel Workbook Used for Event Registrations—Low Degree of Formalism</td>
<td>116</td>
</tr>
<tr>
<td>Figure 8–8</td>
<td>Full Event Life Cycle Registration Form—High Degree of Formalism</td>
<td>117</td>
</tr>
<tr>
<td>Figure 8–9</td>
<td>Change Requests Derived from an Event Registration</td>
<td>119</td>
</tr>
<tr>
<td>Figure 8–10</td>
<td>Excel Workbook Used for Change Request Registrations</td>
<td>122</td>
</tr>
<tr>
<td>Figure 8–11</td>
<td>Change Request Life Cycle Registration Form—High Degree of Formalism</td>
<td>123</td>
</tr>
<tr>
<td>Figure 9–1</td>
<td>Release Note Example 1</td>
<td>126</td>
</tr>
<tr>
<td>Figure 9–2</td>
<td>Release Note Example 2</td>
<td>126</td>
</tr>
<tr>
<td>Figure 9–3</td>
<td>Item Status List</td>
<td>127</td>
</tr>
<tr>
<td>Figure 9–4</td>
<td>Item History List</td>
<td>128</td>
</tr>
<tr>
<td>Figure 9–5</td>
<td>Item Composition List</td>
<td>129</td>
</tr>
<tr>
<td>Figure 9–6</td>
<td>Trace Report, Part a</td>
<td>130</td>
</tr>
<tr>
<td>Figure 9–7</td>
<td>Trace Report, Part b</td>
<td>131</td>
</tr>
<tr>
<td>Figure 11–1</td>
<td>Multiple Configuration Control Boards</td>
<td>144</td>
</tr>
<tr>
<td>Figure 14–1</td>
<td>Customer, Contractor, and Subcontractor</td>
<td>170</td>
</tr>
<tr>
<td>Figure IV–1</td>
<td>Configuration Management Complexity for One Item</td>
<td>174</td>
</tr>
<tr>
<td>Figure 15–1</td>
<td>Document Delivery</td>
<td>180</td>
</tr>
<tr>
<td>Figure 16–1</td>
<td>Single Requirement</td>
<td>188</td>
</tr>
<tr>
<td>Figure 16–2</td>
<td>Test-Related Tracings</td>
<td>194</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td>Page</td>
</tr>
<tr>
<td>--------</td>
<td>------------------------------------------------------------------------------</td>
<td>------</td>
</tr>
<tr>
<td>Figure 17–1</td>
<td>Release Note for Full Delivery (Excerpt)</td>
<td>203</td>
</tr>
<tr>
<td>Figure 18–1</td>
<td>Initial Build Success, All Subsystems</td>
<td>215</td>
</tr>
<tr>
<td>Figure 18–2</td>
<td>Promotion of Code</td>
<td>215</td>
</tr>
<tr>
<td>Figure 18–3</td>
<td>Configurator Subsystem Beta Release</td>
<td>216</td>
</tr>
<tr>
<td>Figure 18–4</td>
<td>Salespoint Subsystem Beta Release</td>
<td>216</td>
</tr>
<tr>
<td>Figure 18–5</td>
<td>Stages in Iterative Development</td>
<td>218</td>
</tr>
<tr>
<td>Figure 18–6</td>
<td>Pure Waterfall Development Model</td>
<td>221</td>
</tr>
<tr>
<td>Figure 18–7</td>
<td>W-Model</td>
<td>223</td>
</tr>
<tr>
<td>Figure 19–1</td>
<td>Composite System—Example</td>
<td>226</td>
</tr>
<tr>
<td>Figure 19–2</td>
<td>Web Usage</td>
<td>236</td>
</tr>
<tr>
<td>Figure 20–1</td>
<td>Multisite Development—Sharing Items</td>
<td>240</td>
</tr>
<tr>
<td>Figure 20–2</td>
<td>Synchronization of Multisites</td>
<td>242</td>
</tr>
<tr>
<td>Figure 20–3</td>
<td>Synchronization</td>
<td>243</td>
</tr>
<tr>
<td>Figure 20–4</td>
<td>Multisite Build</td>
<td>244</td>
</tr>
<tr>
<td>Figure 20–5</td>
<td>Parallel Development and Merge</td>
<td>247</td>
</tr>
<tr>
<td>Figure 20–6</td>
<td>Naming Conventions in Parallel Development</td>
<td>248</td>
</tr>
<tr>
<td>Figure 21.1</td>
<td>Usage Matrix</td>
<td>257</td>
</tr>
<tr>
<td>Figure 21–2</td>
<td>Use of Reuse Components in Projects</td>
<td>260</td>
</tr>
<tr>
<td>Figure 21–3</td>
<td>Event Registration Involving a Component</td>
<td>261</td>
</tr>
<tr>
<td>Figure V–1</td>
<td>CMMI Capability Levels</td>
<td>266</td>
</tr>
<tr>
<td>Figure 23–1</td>
<td>RASIC Chart Example</td>
<td>281</td>
</tr>
<tr>
<td>Figure 24–1</td>
<td>Simple Identification Process</td>
<td>291</td>
</tr>
<tr>
<td>Figure 24–2</td>
<td>Flowchart Conventions</td>
<td>294</td>
</tr>
<tr>
<td>Figure 25–1</td>
<td>Process Deployment and Improvement</td>
<td>296</td>
</tr>
<tr>
<td>Figure 25–2</td>
<td>Average and Range Control Chart</td>
<td>304</td>
</tr>
<tr>
<td>Figure 26–1</td>
<td>Configuration Management Tool Classes</td>
<td>306</td>
</tr>
</tbody>
</table>
# List of Tables

Table I–1 Improvement Recommendations .................................. xliv
Table 1–1 Overview of Change Control Phases ............................. 21
Table 2–1 Mapping from CMM v.1.1 Activities. ............................ 31
Table 2–2 Discipline Description in CMMI. ................................. 33
Table 2–3 Definition of Capability and Maturity Levels in CMMI .... 36
Table 2–4 Mapping from CMMI Activities ................................. 39
Table 2–5 Mapping from SPICE Best Practices ......................... 44
Table 3–1 Standards Overview. ................................................. 46
Table 3–2 Mapping from ISO 9001:2000 Sections ....................... 53
Table 5–1 Activities with a Low Degree of Formalism ................. 64
Table 5–2 Activities with a High Degree of Formalism ............... 65
Table 5–3 Examples of Possible Savings Using a Configuration Management System .................. 69
Table 6–1 Contents of Milestone Deliveries .............................. 87
Table 7–1 Configuration Item Name Parts and Their Functions .... 92
Table 7–2 Source Code Unit State Examples .............................. 95
### List of Tables

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>8–1</td>
<td>Data Elements for Event Registration in the Creation Phase</td>
<td>113</td>
</tr>
<tr>
<td>8–2</td>
<td>Data Elements for Event Registration in the Evaluation Phase</td>
<td>114</td>
</tr>
<tr>
<td>8–3</td>
<td>Data Elements for Event Registration in the Decision Phase</td>
<td>114</td>
</tr>
<tr>
<td>8–4</td>
<td>Data Elements for a Change Request in the Creation Phase</td>
<td>121</td>
</tr>
<tr>
<td>8–5</td>
<td>Data Elements for a Change Request in the Implementation Phase</td>
<td>121</td>
</tr>
<tr>
<td>10–1</td>
<td>Belbin Team Roles Descriptions</td>
<td>139</td>
</tr>
<tr>
<td>15–1</td>
<td>Development in Milestone Deliveries</td>
<td>176</td>
</tr>
<tr>
<td>15–2</td>
<td>Unique Identification of Milestone Deliveries</td>
<td>177</td>
</tr>
<tr>
<td>15–3</td>
<td>Unique Identification for Documents</td>
<td>181</td>
</tr>
<tr>
<td>15–4</td>
<td>Unique Identification of Subitems</td>
<td>182</td>
</tr>
<tr>
<td>16–1</td>
<td>Document Types</td>
<td>188</td>
</tr>
<tr>
<td>16–2</td>
<td>Unique Identification for Coding Items</td>
<td>189</td>
</tr>
<tr>
<td>16–3</td>
<td>Unique Identification for Integration Items</td>
<td>192</td>
</tr>
<tr>
<td>17–1</td>
<td>Project Management Document Types</td>
<td>201</td>
</tr>
<tr>
<td>19–1</td>
<td>Safety Criticality Classification Example</td>
<td>232</td>
</tr>
<tr>
<td>25–1</td>
<td>Metric and Measurement Terms</td>
<td>298</td>
</tr>
<tr>
<td>25–2</td>
<td>Data Types</td>
<td>298</td>
</tr>
<tr>
<td>25–3</td>
<td>Aspects of Metrics</td>
<td>299</td>
</tr>
<tr>
<td>25–4</td>
<td>Scale Definitions</td>
<td>299</td>
</tr>
<tr>
<td>25–5</td>
<td>Requirements for Metrics</td>
<td>300</td>
</tr>
<tr>
<td>26.1</td>
<td>Buy It or Do It Yourself Considerations</td>
<td>309</td>
</tr>
</tbody>
</table>
Solving the problems in configuration management can dramatically reduce the cost of rework, not to mention reduce the number of programmer headaches. I was fortunate to work in a company that did very well with configuration management practices on their proprietary systems. However, when they began software development on open systems, it was not so easy. Things that were second nature, that were so internalized that we didn’t have to think about them anymore, suddenly became the things we didn’t have the foresight to think about on the new systems. We began to have problems again. We had to relearn things that we thought we had learned before, and it was difficult to go back and learn them all over again. The explanations of the concepts, definitions, roles, and responsibilities in this book would have helped us then.

This book will also help those who have never had the appropriate level of discipline in their workplace for configuration management, especially those who have experienced horror stories like these:

- The Lost Software: “I know I wrote it, but I don’t know where I put it.”
- The Missing Links: “This used to work, but now it points to code that isn’t there anymore.”
- Stepping on each other’s code: Developers doing different fixes in the same code area, overwriting each other.
- You Can’t Go Back Again: New fixes are worse, and there’s no “undo” button.
- You Can’t Put It Together Again: Dropped a document with no page numbers, or dropped two documents, no titles on pages, which was which?
Who’s on First? What’s on Second? Bug reported by customer, but don’t know what version they have, don’t know what fix to give them.

“But I Know I Fixed It!”
— Customer calls and says, “It’s broken.”
— Programmer makes the fix, but forgets to check-in the change.
— Software build is done without the fix (No one audited the baseline).
— Exact same software shipped to the customer.
— Customer calls and says, “It’s still broken.”
— Programmer says, “But I know I fixed it!”

Configuration management is a cornerstone of software process improvement. (After all, if you can’t manage your stuff, how can you tell whether your stuff has improved?) In *CMM Implementation Guide*, I wrote: “In the software industry, many of us have taken steps in software process improvement and made the steps our own, but perhaps many of us have not yet taken the more difficult steps of allowing ourselves to learn from each other and change under cross-cultural influence. It won’t happen unless we share our experiences and our techniques. I am sharing my experiences and techniques, not to tell people to do it my way but rather to open the door for us to learn from each other throughout the industry and throughout the world. Perhaps I am not the first to open this door, and I hope that I am not the last. This is an invitation to the dance.”

Across the world, nine time zones away, Anne Mette Jonassen Hass has answered the invitation and come through with a wonderful contribution. Here she shares her experiences and techniques for successful configuration management, with several possibilities for solutions that readers can take and make the steps their own. She also includes a wealth of references to reach more information for further learning. I am delighted with this contribution that takes up the call to influence our industry and our world.

—Kim Caputo
Mission Viejo, California
Software configuration management and automated regression testing tools are the two development tools most critical to the success of the agile project. Over the last ten years, the version control and configuration management system was consistently cited to me as the top priority tool to install, both for agility-focused and plan-driven projects. No other tool even came close. (The editor and compiler are so integral that they don’t get named.) Teams used to working with a version control and configuration management system refuse to operate without one.

Many teams find that once they have a satisfactory configuration management system in place they can do something more important to their project than merely coordinate their check-ins: They start experimenting with shorter and shorter periods between builds. (This is when the automated regression testing tool becomes important.)

Some teams run fully automated builds every half hour; these also run the suite of unit and system regression tests, post the results on a Web page, and email the owners of any failed code their failing test results! People on these teams report an increase in speed, agility, quality, and personal comfort, knowing they’ll learn of unexpected errors within a half-hour of checking in their code.

One company is even experimenting with using such a continuous-build system to synchronize the work between India and the United States. They report that it is helping the two teams stay synchronized with each other across nine time zones.

It is therefore astonishing to see how many teams try to work without a configuration management system. Moreover, it can be frustratingly difficult to find information on the topic.
Anne Mette Hass manages in this book to capture both the heart of the subject and the variations needed in widely varying circumstances—a rare accomplishment. She knows, as you do, that some organizations run with heavy bureaucracy, some with little bureaucracy, some with little formality, some with great formality—and all need configuration management to smooth their collective work. She presents the topic from several angles: the work products, the job roles involved, the organizational issues, the tools, and various levels of formality and bureaucracy. In addition to her insights, Steve Berczuk and Brad Appleton describe, in their appendix, how the terms and practices can be used on the lightest of agile projects.

I have always found this subject daunting, and was pleased to find this text well presented and easy to digest. I could never have written this book; I’m glad that Anne Mette Hass has done it for us.

—Alistair Cockburn
Salt Lake City, Utah
My Life as a Software Professional

I have two—well, three really—passions in my professional life: test, configuration management, and process improvement.

I started my career as an all-around developer—a little requirements elicitation, a little analysis, a lot of coding and recoding, and some test—more than 20 years ago. During these first professional years, I always loved testing most—making my work run on the computer and enjoying the satisfaction of being told, in a factual and precise way, that something was wrong. This enabled me to carry out the correction and then finally enjoy the privilege of knowing that at least this error was a secret between me and the computer.

My experience grew, and my working teams grew. The problems grew. I wasn’t always certain I had produced what I was supposed to and that I had tested everything. And sometimes an error would recur!

I got a job in which I was responsible for system and acceptance test in a company making software for the European Space Agency. For the first time in my then 12-year career, I heard the words configuration management. I had no clue as to what it was, but as I spent hours and hours trying to figure it out, discussing it with the person responsible for quality assurance and actually using parts of it in my daily work, I came to understand what a wonderful tool I had.

For the first time, I was able to trace my test cases to the requirements. I was able to tell, at any point, how many requirements I had covered in my test specification and how many were outstanding. I didn’t have to encounter the frustration of having made test cases for requirements that weren’t going to be implemented. Where I had
forgotten the reason for a turn in the work, I was able to find a previous version of my test specification and see why I had changed it. I loved it!

The last seven years, I’ve worked as a consultant, spending a good deal of my time on testing assignments of many types in many companies. One of the things I’ve learned from these assignments is that there is often a difference between what a customer asks for and what he really wants, what he needs (what you want to give him), and what you’re able to give him.

Test consultants are often presented with a system to test without the right conditions for performing a professional test. The requirements may be in any state from nonexistent to brilliantly documented, with a pronounced bias toward the former. If requirements are present, they are most often not up to date. This is partly a requirement specification problem and partly a configuration management problem.

Testing requires resources in terms of time and people to perform the test. These resources are often all too scarce. This is a project management problem.

When test consultants plan and perform a test, they need to establish an overview not only of what has to be tested but also how the test is progressing, what errors have been found, and what the state of error correction is. These are configuration management issues.

It’s tempting for a consultant to try to deliver what the customer really needs. However, this approach has some limitations and drawbacks. The art is to strike the right balance between what’s needed and what’s feasible. One of the things to keep in mind as a consultant is to keep up the standards but keep it light. So I try to keep up the configuration management standards as I solve the test assignment—hoping my customer will get an idea of what configuration management is and maybe ask for some assistance in that direction too.

Another part of my time is spent assessing software-producing companies using the BOOTSTRAP maturity model and method. Like the related Capability Maturity Model (CMM), this model includes configuration management. As an assessor in more than 40 assessments, I have time and again seen the blank look in people’s eyes when I ask how they perform configuration management. The eyes are rarely less blank if I elaborate and ask about tracing between work products, production of error reports, or other detailed configuration management disciplines.

On the other hand, people are more than willing to talk about problems they’ve experienced due to lack of control over what is being implemented and tested—and when—and lack of control over what errors have occurred and which ones are being corrected and which are not.

Although configuration management is one of the basic disciplines for sound development (in CMM it is a key process area at level 2), many people go through a
considerable part of their careers without any idea of what it is and how it can ease their everyday tasks, just as I did. So I keep emphasizing its importance and very often recommend it as one of the first disciplines a company should work on when embarking on structured process improvement.

Creation of This Book

In 1999, the Danish organization Datateknisk Forum, an association of about 70 software-producing companies, asked me to write a book on configuration management. This was the result of a survey among the members as to what topic they needed a book on. Some of the comments and requirements that came back from the survey were

- How do you incorporate configuration management in the development process?
- How do you handle the fact that different kinds of work products, like documents and code, are treated differently?
- How do you obtain integration between different configuration management tools?
- How do you handle multisite development?
- How do you handle configuration management in relation to object-oriented development—component-based development?

I took on the assignment because in my own experience, configuration management has been of great value, not because I felt I knew much about it theoretically. I know much more now, and I hope I’ve conveyed some of the understanding, knowledge, and appreciation I’ve gained during my work on this book. If readers try at least some of the detailed disciplines, I hope they will experience the same enthusiasm about its usefulness that I did.

The book is based on literature as well as experience—and also on attitudes and opinions. It contains a lot of examples, advice, and recommendations that are not to be regarded as The Truth but primarily as the sum of a lot of experience—negative as well as positive.

When I learned that the book was to be published in the Agile Series, I knew little about agile development. But as I studied the values and principles, I found that I had practiced it in parts for years. Agile development is a wonderful idea, and one of the cornerstones of its success is configuration management, so it was a pleasure to be able to contribute to the series with one of my favorite disciplines.
The book may seem a bit heavy to some agilists, but I think it’s better to discard some formality and detailed activities deliberately, knowing what one hasn’t performed, than to just not perform it out of ignorance. So, agilists and others, read and choose!

**Purpose of the Book**

This book is not supposed to be a primer in configuration management. It does, however, start with an introduction to fundamental principles, to establish a basic understanding of the concepts used. The main part of the book discusses more advanced issues encountered when configuration management has to be implemented. The overall purpose of the book is twofold:

- To scare those who are engaging in configuration management! The book will give the reader an understanding of the complexity and comprehensiveness of the discipline. *Configuration management is not easy!* If you think it is, you’ll be unable to solve its tasks in a professional way.

- To assuage the fear of those who are engaging in configuration management! The book will provide a fundamental understanding of the principles of the discipline, their interrelations and usage. *Configuration management is not difficult!* All you have to do is do it. If you understand it, it’s much easier to specify and plan so it fulfills its purpose and becomes manageable.

It’s assumed that the reader has some knowledge of other disciplines within software development, such as planning, design, test, and quality assurance.

**Thanks**

A lot of people have supported the creation of this book. I have no way of mentioning them all. First, I would like to thank the members and the board of Datateknisk Forum and my managers, Mr. Jørn Johansen and Mr. Ole Andersen, for believing in the idea and contributing to the contents.

I would also like to thank my colleagues (especially Ms. Elisabeth Broe Christensen and Mr. Robert Olesen), Mr. Lars Bendix of the University of Lund, Sweden, and not least my husband, Finn, for providing many pieces of good advice and good ideas, and for the interest and patience they have shown during my work on the book. My husband’s wry way of looking at things is sometimes annoying but always enlightening—thanks, Finn, for being who you are!
The publisher and my editor, Mr. Ross Venables, deserve lots of thanks for their enthusiasm and encouragement, all the way from my first approach through the development of the manuscript to the complete book.

Last but not least, a big thanks goes to my longtime friend Ms. Pernille Lemvig-Fog and my father, Mr. Birger Jonassen, for their great help with the translation of the text into understandable English.
Introduction

I.1 CONFIGURATION MANAGEMENT IN COMPANY PERSPECTIVE

Every company or organizational unit in a company that develops products should consider configuration management. Configuration management becomes part of the general culture. This means it should be adjusted to the company culture, whether loose, rigorous, or in between. Configuration management may be viewed from different perspectives: people, product, project, cross-organizational, process, and tools. Each is briefly introduced below and discussed at greater length in the book.

People Perspective

Many people affect and are affected by configuration management by fulfilling the roles it involves. These may be categorized as configuration management roles, organizational roles, project-related roles, and external roles.

Product Perspective

Configuration management to be performed for a product depends on the nature of the product. Today, we find more and more complex products composed of different types of subproducts, such as software (applications), hardware (boxes, PCs, peripherals), networks (LAN, Internet), data (system data, parameter values), services (intangible
deliveries such as training and maintenance). Any product may have more or less—
even no—emphasis on subproducts. A product may, for example, be

- A pure software product, delivered on a CD-ROM with no hardware, no initial
data, no support or any other service, and no network connection
- A large control system, including
  - Software embedded in some hardware and in the network
  - PC software with a graphical user interface
  - Network connections for remote surveillance and support
  - Initial data and parameters set
  - Training courses and maintenance services included in the delivery

Products may be simple, complex, or somewhere in between. They may be harm-
less, with no great impact on human lives or other companies, like games or house-
hold equipment, or they may be safety-critical, like flight control systems or medical
equipment. They may be developed as shrink-wrapped products, like a test tool, or as
bespoke software, like a control system for a factory. Any product has a combination
of these attributes.

**Project Perspective**

The work of developing and maintaining a product may be organized in one project
or in a number of projects under different management during the product’s life-
time. The project perspective is concerned with performing configuration manage-
ment for a product in the project or projects during its life cycle. A product goes
through a number of life cycle activities, for which configuration management
should be considered. These may be preparation, requirements specification, design,
production (e.g., coding and/or manufacturing), integration, testing, and operation
and maintenance, as illustrated in Figure I–1.

The activities mentioned above are just building blocks that are arranged accord-
ing to the chosen development model. A number of development models exist, such as
the waterfall model (similar to Figure I–1), agile development, incremental develop-
ment, and iterative development. Each subproduct may follow its own development
model—for example, the software subproduct may follow an iterative development
model, while the hardware subproduct follows a waterfall model.

As Figure I–1 also shows, a number of support functions exist for preparing, devel-
oping, operating, and maintaining a product. These functions, which may include
project management, quality assurance, and configuration management, should be performed during a product’s entire lifetime. Performing these support functions produces objects, which must also be considered for configuration management. The development activities and support functions included in this book are based on the activities and support functions defined in maturity models.

**Cross-Organizational Perspective**

All companies have cross-organizational objects or assets for which configuration management should be considered: infrastructure, company product assets (such as components for reuse developed using a product-line approach), and company documentation (sales material, plans, quality system, process descriptions, and so on).

**Process Perspective**

Configuration management may well be the subject for process improvement. In fact, as soon as a company starts to consider configuration management, the process perspective needs to be taken into account. To sustain the work, processes must be understood and implemented and must continuously undergo improvement.

Process improvement and the concept of maturity models to support it, especially in software development, are becoming more and more common in the industry. In the Capability Maturity Model (CMM), configuration management plays a prominent part as a key process area at level 2. Another maturity model, used mostly
(and most) in Europe, is the BOOTSTRAP model. As part of a BOOTSTRAP assessment, a company is given a list of its five processes that most require improvement. As of early 2001, more than 50 BOOTSTRAP assessments had been performed in Denmark. Table I–1 shows the three most frequently appearing processes.

<table>
<thead>
<tr>
<th>Number</th>
<th>Process</th>
<th>Appearances (%)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.</td>
<td>Project management</td>
<td>75</td>
</tr>
<tr>
<td>2.</td>
<td>Configuration management</td>
<td>55</td>
</tr>
<tr>
<td>3.</td>
<td>Test</td>
<td>51</td>
</tr>
</tbody>
</table>

More than half the projects had problems in the way they implemented configuration management and needed to improve their practices. This made configuration management the second most frequent process.

**Tools Perspective**

It is virtually impossible to manage configuration management without one or more tools. Many tools are available, but many companies prefer to develop their own.

**I.2 CONFIGURATION MANAGEMENT BETWEEN COMPANIES**

**Customers**

Some companies have other companies as direct customers. In such cases, the customer’s demand for configuration management may influence how the discipline is carried out in the delivering company. Typically, delivered products form part of components in other products the customers take care of themselves, or the customer takes over responsibility for the finished product.

A company is a purchaser of products, but it may also act as a supplier—for instance, of a requirement specification or components. The customer’s attitude toward configuration management must be clarified where relevant.
Subcontractors

In some cases, subcontractors work for product-producing companies. The way the subcontractors perform configuration management may influence the way it’s done in the producing company. Control of the subcontractors is a support function within software development and a discipline or process that ought to be present during a product’s whole lifetime. It may be defined for the entire company, if the company has standard procedures for handling subcontractors. (This is not often the case.)

I.3 CONFIGURATION MANAGEMENT IN A BROADER PERSPECTIVE

The World at Large

Universities, research institutes, and companies work with configuration management at several levels. Standards within software development include configuration management as a discipline or a process. During recent years, work in connection with process improvement, including maturity models, has been augmented considerably. Configuration management is included as a process in the best-known maturity models. Furthermore, various institutions and large international projects work with configuration management. These aspects of research into configuration management have been included in the book to provide a larger perspective for what may sometimes seem like an isolated struggle.

A Little Philosophy

Configuration management is the existentialism of software development, because it answers the following questions for individual components or entire products:

- Who am I?
- Why am I here?
- Why am I who I am?
- Where do I belong?

Just as in “real life,” a certain amount of leisure is necessary for that kind of consideration, but if you have the leisure and use it in a reasonable way, it’s possible to increase your quality of life—or, in this case, the quality of your products.