

Software Engineering

COEN 174

Ron Danielson

Configuration Management

Chapter 11

Objectives

- Understand basics of configuration management and how they support software development
- Know what artifacts need to be managed
- Understand issues to consider in choosing configuration management solutions

Configuration Management

- Managing SW artifacts
 - Documents, code, test cases, ...
 - Identify, track, store
- Coordination of change control
 - Checkout, versioning, ...
- Build activities
- Controlling access to and protection of each artifact

CM Systems Provide

- Baseline safety
 - New or changed artifacts are safely stored and can be retrieved
- Overwrite safety
 - Overlapping changes to same artifact are merged correctly
- Reversion
- Disaster recovery

What Volume of Artifacts

- Depends on organization and development process
- Government organizations and contractors keep a lot
 - Well-defined and rigid development processes
 - Versions of all documents at all stages
- Smaller companies may keep much less
 - More ad-hoc development processes
 - Latest version of requirements and code?

Artifact Examples

- Requirement specifications
- Design documentation
- Source code
- Test scenarios, data, and expected outputs
- Executable code
- DB tables
- Initialization data
- Problem reports
- Problem fixes
- End user documentation
- Training materials

Relationships Among Artifacts

- Inter-artifact relationships
 - Usage or dependency
 - Release packages
 - Test levels
- Intra-artifact relationships
 - Different versions of artifact
 - For different industries
 - Globalization

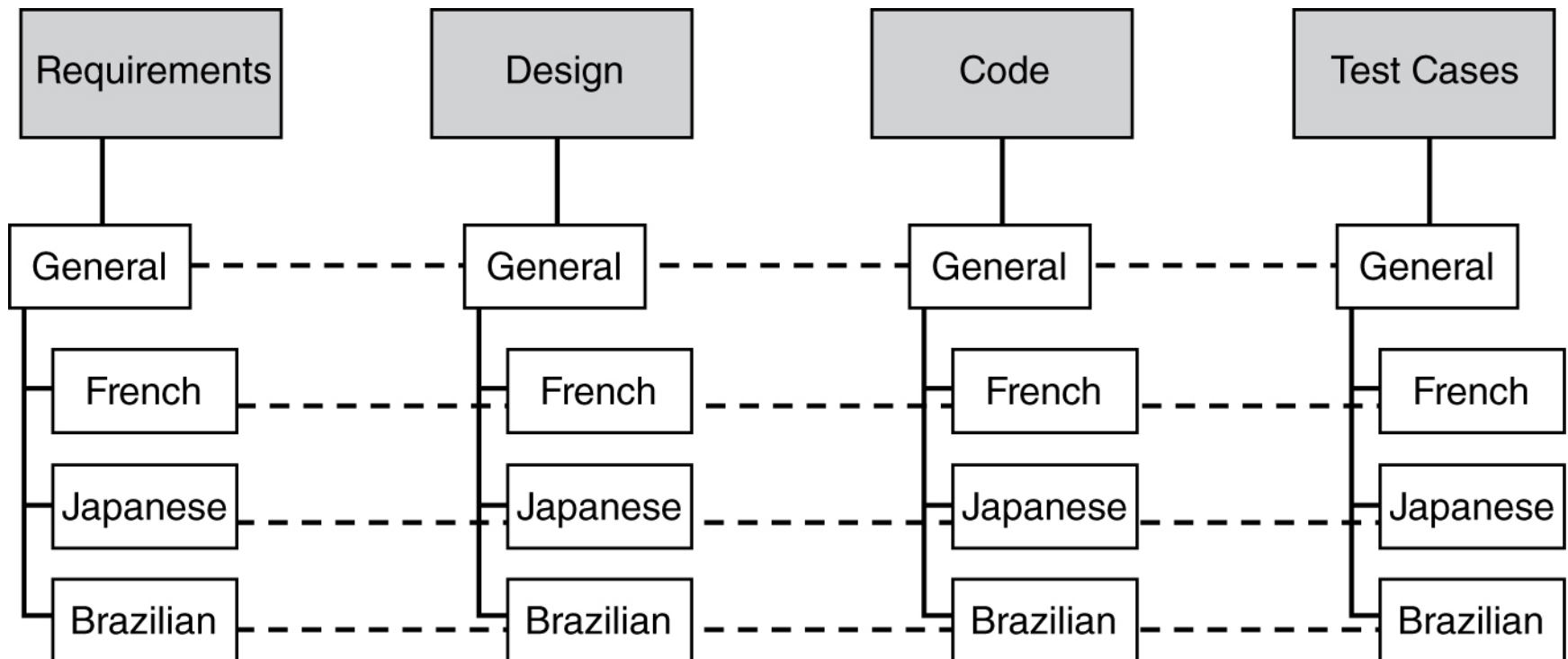
Relationship Matrix

Table 11.1 A Matrix Showing Interartifact Relationships

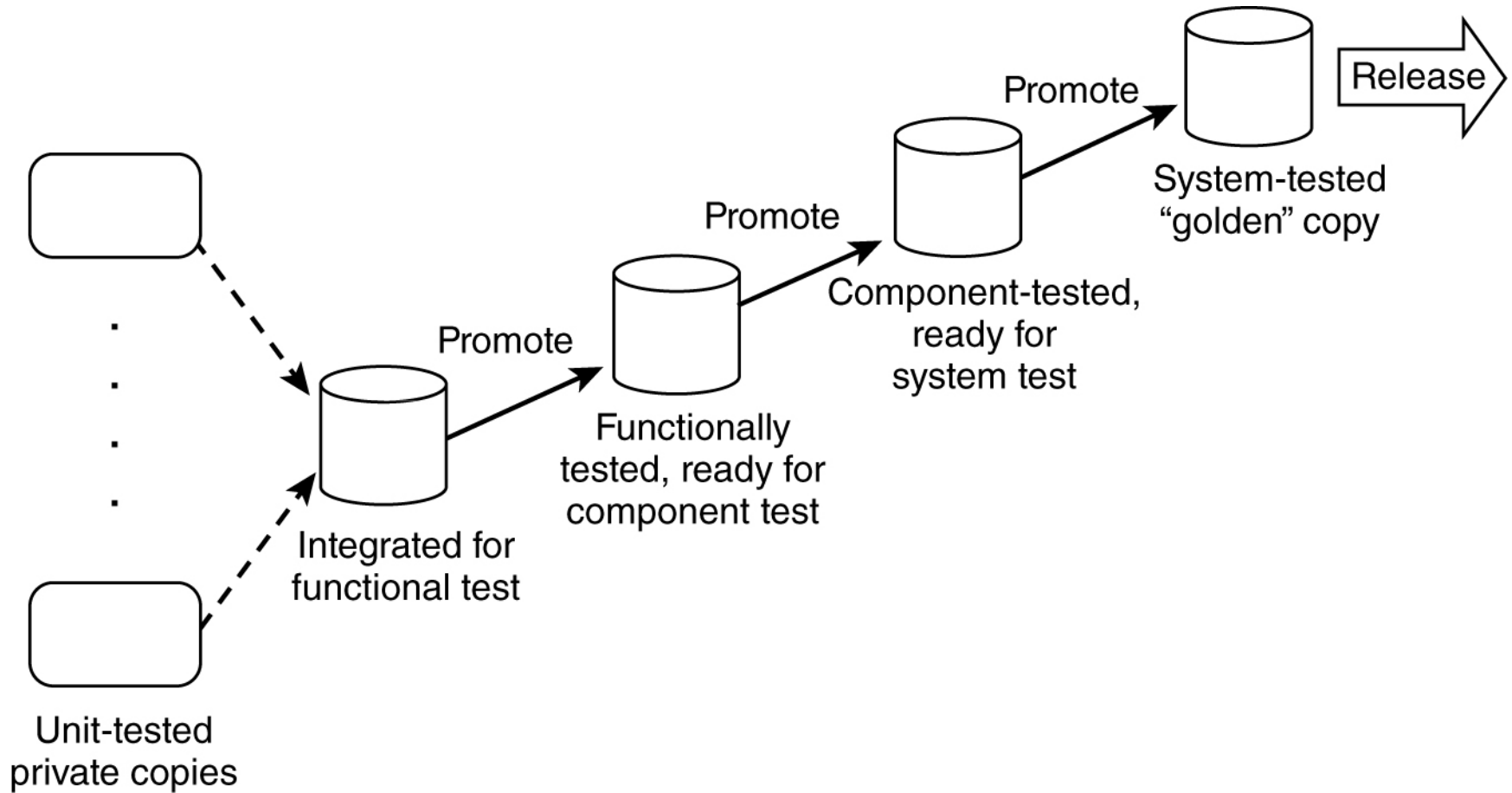
	Requirements Elements	Design Elements	Code Logic	UI Screens	DB Tables	Initialization Data	Test Cases
Requirements Elements		X	X	X			X
Design Elements	X		X	X	X	X	
Code Logic	X	X		X	X	X	X
UI Screens	X	X	X				X
DB Tables		X	X			X	
Initialization Data		X	X		X		X
Test Cases	X		X	X		X	

Multiple Relationships

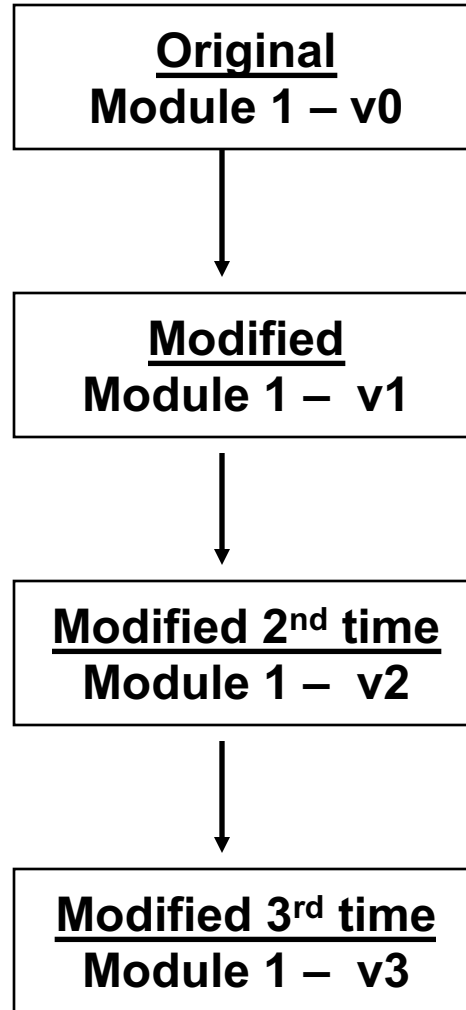
- Intra-entity relationships are vertical
- Inter-entity relationships are horizontal



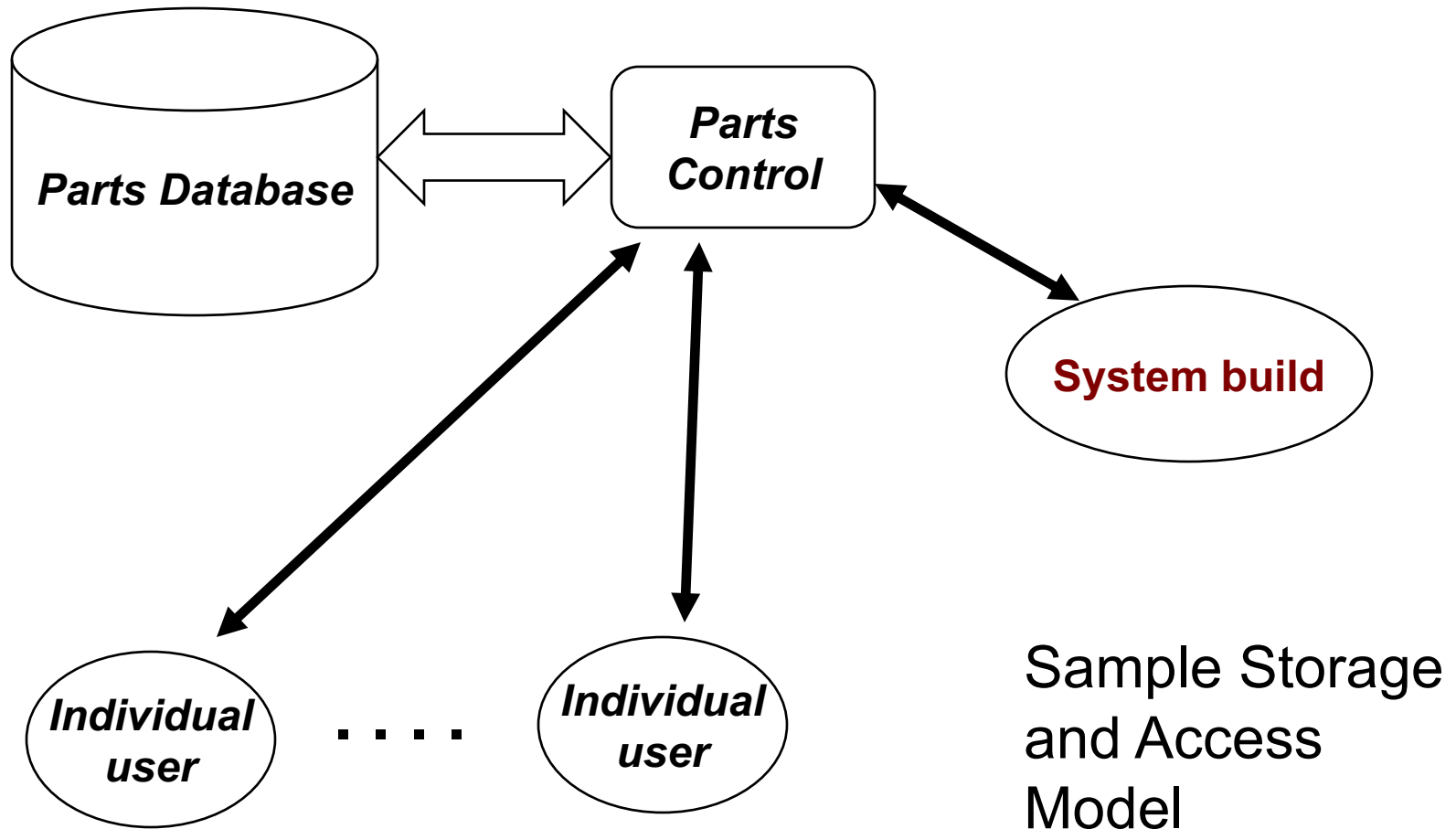
Intra-Entity Example



Intra-Entity Example



Configuration Management Frameworks



Configuration Management Frameworks (cont.)

- Basic
 - Create
 - Delete
- Access
 - View
 - Modify
 - Return
- Control and service
 - Import
 - Export
 - List
 - Set release or version number
 - Change name
 - Gather
 - Merge
 - Promote
 - Compare
 - Lock/unlock
 - Where-use
 - Cross-references

CM System Activities

- Configuration identification
- Baseline control
- Change control
- Version control
- Configuration auditing
- Configuration status reporting
- Release management and delivery

Configuration Identification

- Sample naming model
 - Six-tuple suffix code
 - PP.CC.RRR.VVV.TT.FF
 - Product
 - Country
 - Release
 - Version
 - Type
 - Format

Configuration Identification (cont.)

- Also need
 - Name
 - Date
 - Author
 - Revision history
 - Status

Baselines

- Artifact that has been formally reviewed and approved
 - Basis for future development
- Also collections of artifacts at key milestones
 - Version release
- Baselines provide
 - Reproducibility
 - Traceability
 - Reporting
 - E.g. differences between versions

Change Control

- Manages changes to baselines
 - Request
 - Evaluate
 - Approve/disapprove
- For each change, note
 - Name of requester
 - Description and extent of change
 - Reason for change
 - Urgency
 - Elapsed time to make change
 - Impact on other artifacts

Change Control (cont.)

- Analysis and evaluation
 - Of changes to baseline
 - Review and approval becomes more formal as project gets closer to release time
 - Initially peers or project manager
 - Later Change Control Board
 - PM, marketing, QA, development
 - Consider
 - Reason for change
 - Extent of change (e.g., LOC involved)
 - Complexity
 - Other artifacts affected
 - Testing needed to verify change

Change Control (cont.)

- Approval or disapproval
 - Or defer
- Verify, implement and release changes