

Object-Oriented Software Analysis and Design

School of Computer Science
University of Windsor

Other Requirements

- ▶ Besides use cases, what other UP requirement artifacts?
- ▶ Use cases are not the whole story!
- ▶ There are at least the following other requirement artifacts:

Other Requirement Artifacts: Supplementary Specification

- ▶ captures and identifies other kinds of requirements, such as reports, documentation, packaging, supportability, licensing, and so forth.

Other Requirement Artifacts: Glossary (Data Dictionary)

- ▶ captures terms and definitions; it can also play the role of a data dictionary.

Other Requirement Artifacts: Vision

- ▶ summarizes the “vision” of the project - an executive summary. It serves to tersely communicate the big ideas.

Other Requirement Artifacts:

Business Rules (or Domain Rules)

- ▶ capture long-living and spanning rules or policies, such as tax laws, that transcend one particular application.

Guideline: Should we analyze these thoroughly during inception?

- ▶ No.
- ▶ The UP is an iterative and evolutionary method, which means that production-quality programming and testing should happen very early, long before most requirements have been fully analyzed or recorded. Feedback from early programming and tests evolve the requirements.
- ▶ However, research shows that is useful to have a high-level “top ten” list of coarse-grained requirements near the start. It is also useful to spend non-trivial early time understanding the non-functional requirements (such as performance or reliability), as these have a significant impact on architectural choices.

Guideline: Should These Artifacts be at the Project Website?

- ▶ Definitely.
- ▶ These should usually be digital artifacts recorded only online at the project website. And instead of being plain static documents, they may be **hyperlinked, or recorded** in tools other than a word processor or spreadsheet.

For example,

many of these could be stored in a Wiki Web.

NextGen Example: (Partial) Supplementary Specification

- ▶ uploaded separately in the BB
- ▶ The Supplementary Specification captures other requirements, information, and constraints not easily captured in the use cases or Glossary, including system-wide “**URPS+**” (usability, reliability, performance, supportability, and more) quality attributes or requirements.

NextGen Example: (Partial) Vision

- ▶ uploaded separately in the BB
- ▶ A Vision with less than 10 features is desirable—more can't be quickly grasped. If more, consider grouping and abstracting the features.

NextGen Example: A (Partial) Glossary

- ▶ uploaded separately in the BB
- ▶ Start the Glossary early. It will quickly become a useful repository of detailed information related to fine-grained elements.

NextGen Example: Business Rules(Domain Rules)

- ▶ uploaded separately in the BB
- ▶ Domain rules dictate how a domain or business may operate. They are not requirements of any one application, although an application's requirements are often influenced by domain rules. Company policies, physical laws (such as how oil flows underground), and government laws are common domain rules.

Evolutionary Requirements in Iterative Methods

Table 7.1. Sample UP artifacts and timing. s - start; r - refine

Discipline	Artifact	Incep.	Elab.	Const.	Trans.
	Iteration→	I1	E1..En	C1..Cn	T1..T2
Business Modeling	Domain Model		s		
Requirements	Use-Case Model	s	r		
	<i>Vision</i>	s	r		
	<i>Supplementary Specification</i>	s	r		
	<i>Glossary</i>	s	r		
	<i>Business Rules</i>	s	r		
Design	Design Model		s	r	
	SW Architecture Document		s		
	Data Model		s	r	

It's Quiz Time

1. Elements of the Supplementary Specification include:
hardware and software constraints (operating and networking systems,...) (True or False)
2. The is a list of noteworthy terms and their definitions.
 - 2.1 Supplementary specification
 - 2.2 Vision
 - 2.3 Data dictionary
 - 2.4 Business rules
3. Are physical laws and government laws are common domain rules? (True or False)