Requirements Specification & Quality Requirements

Lectures 4b&5, DAT230, Requirements Engineering
Robert Feldt, 2011-09-08 & 2011-09-13

Schedule this week

- L6 is only virtual / video!
- Assignment 3 intro only virtual / video!
- Thursday 13:15-17:00: Workshop! Important!
- Thursday 17:00: Groups uploaded to home page
- Friday/Monday: Convene with group and plan for interview next week

Recap

- Elicitation to find/gather/create/refine/specify reqs & understand stakeholder needs
- Many different elicitation techniques
  - Interviews, Group sessions, Observation are key
  - Always: care, be human, listen, focus on them, glossary
- Other sources: Docs, Strategies, Problem domain, History, Competitors, Environment
- Different abstraction levels
- Structured interview more powerful than open-ended

Elicitation methods

Elicitation methods

Interviews

Questionnaires

Surveys

Doc analysis

"Traditional"

Elicitation methods

Interviews
Questionnaires
Surveys
Doc analysis

"Traditional"

Brainstorming
Focus groups
JAD/RAD
Req Workshops

Group-based

Elicitation methods

Interviews
Questionnaires
Surveys
Doc analysis

"Traditional"

Brainstorming
Focus groups
JAD/RAD
Req Workshops

Group-based

Think-aloud / Protocol Analysis
Laddering
Card sorting
Repertory grids

"Cognitive"

Elicitation methods

Interviews
Questionnaires
Surveys
Doc analysis

"Traditional"

Brainstorming
Focus groups
JAD/RAD
Req Workshops

Group-based

Think-aloud / Protocol Analysis
Laddering
Card sorting
Repertory grids

"Cognitive"

Ethnography
Observation
Conversation analysis

Contextual

Elicitation methods

Interviews
Questionnaires
Surveys
Doc analysis

"Traditional"

Brainstorming
Focus groups
JAD/RAD
Req Workshops

Group-based

Think-aloud / Protocol Analysis
Laddering
Card sorting
Repertory grids

"Cognitive"

Ethnography
Observation
Conversation analysis

Contextual

KAOS
I*
CREWS

Model-driven

Elicitation methods

Interviews
Questionnaires
Surveys
Doc analysis

"Traditional"

Brainstorming
Focus groups
JAD/RAD
Req Workshops

Group-based

Think-aloud / Protocol Analysis
Laddering
Card sorting
Repertory grids

"Cognitive"

Ethnography
Observation
Conversation analysis

Contextual

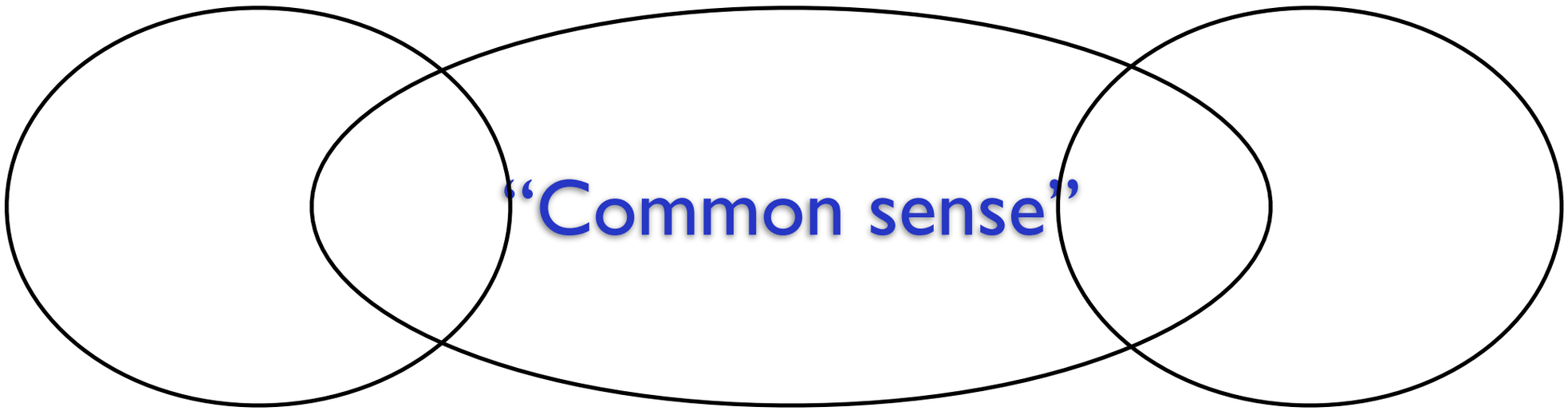
KAOS
I*
CREWS

Model-driven

Working prototypes
Mashups
Drawings

Prototyping

Cost-effectiveness

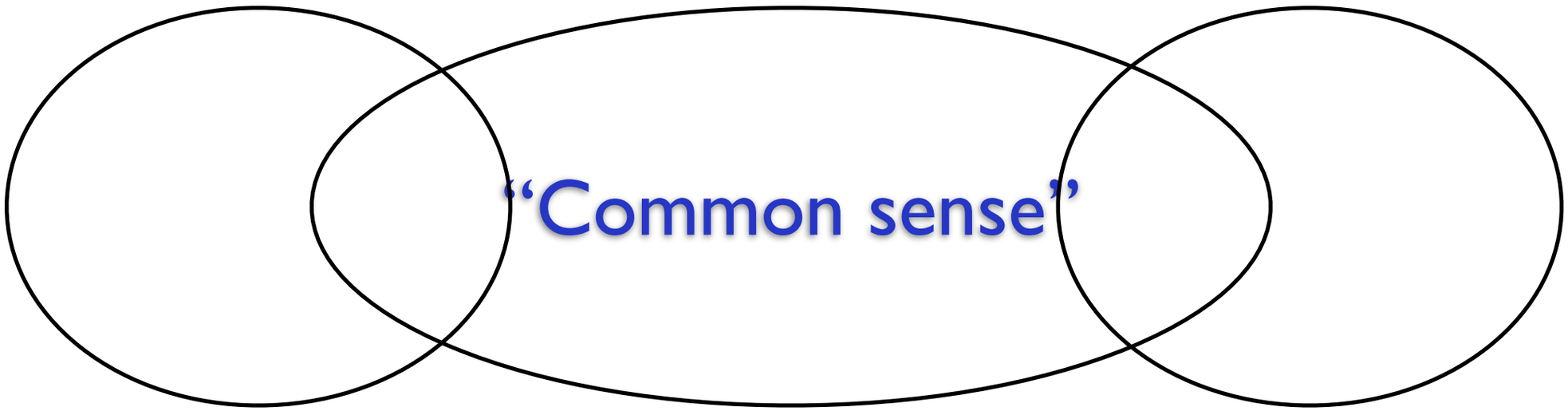


Customers/Users

SRS Doc

Developers

Cost-effectiveness

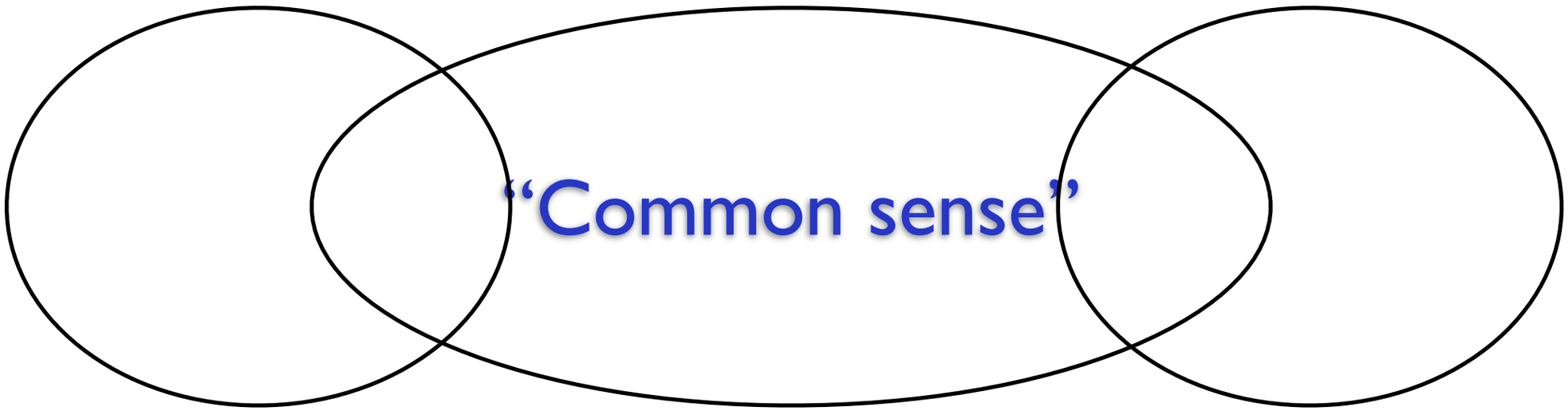


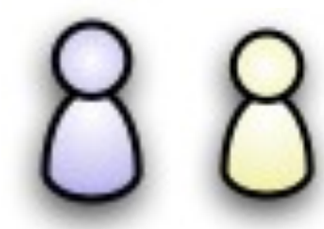
Customers/Users

SRS Doc

Developers

Cost-effectiveness



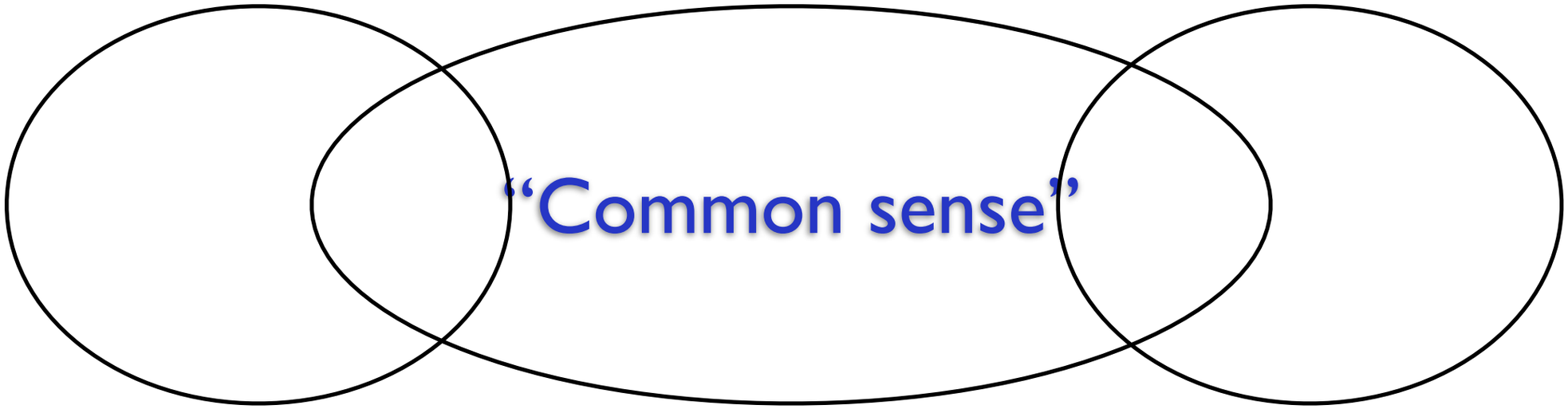


Customers/Users

SRS Doc

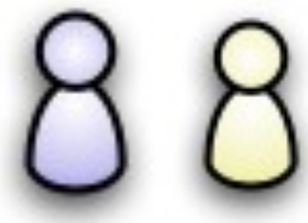
Developers

Cost-effectiveness



Customers/Users

SRS Doc



Developers

Roles of Req Doc

- Communication device between all parties
  - Customers, Marketing, Sales, Finance, Management, Devs, Testers
- Drives design and choices
- Drives testing
- Drives project management
- Basis for evolution / releases

Specification Techniques

Word doc
Excel doc
DB / Req tool

Text

Scenario
Use case
Storyboard
Stimulus-response sequence

Interaction- / Sequence-based

State transition diagram
UML state diagram

State-based

Decision tables
Decision trees

Decision-based

CSP
Z
VDM
Property-based

Formal

PLanguage
Volere
Probabilistic Quality Patterns

Quality Requirements

UI standards
Text
Prototype
Sketches
Look'n'feel samples

User Interfaces

Selecting techniques

- Stakeholders must understand => Natural Language
- Models where NatLang has risks:
  - Complex interactions/sequences/states/decisions
  - Interfaces
  - BUT not "One model to rule them all!"
- Quality requirements:
  - Quantify
  - Capture in structured english or PLanguage

Industrial survey: Methods for ReqEng?

Uses...	**"Yes"**
Reviews of requirements	63.8%
Model-based development	25.0%
Prototype-based development	24.3%
Prioritization of reqs	23.7%
Personas for req elicitation	20.4%
UML	17.8%
Modeling/formalisms for reqs	11.8%
Software Product Lines	5.9%

152 answers from Swedish industry, Spring 2009

Tool for Req Eng work?

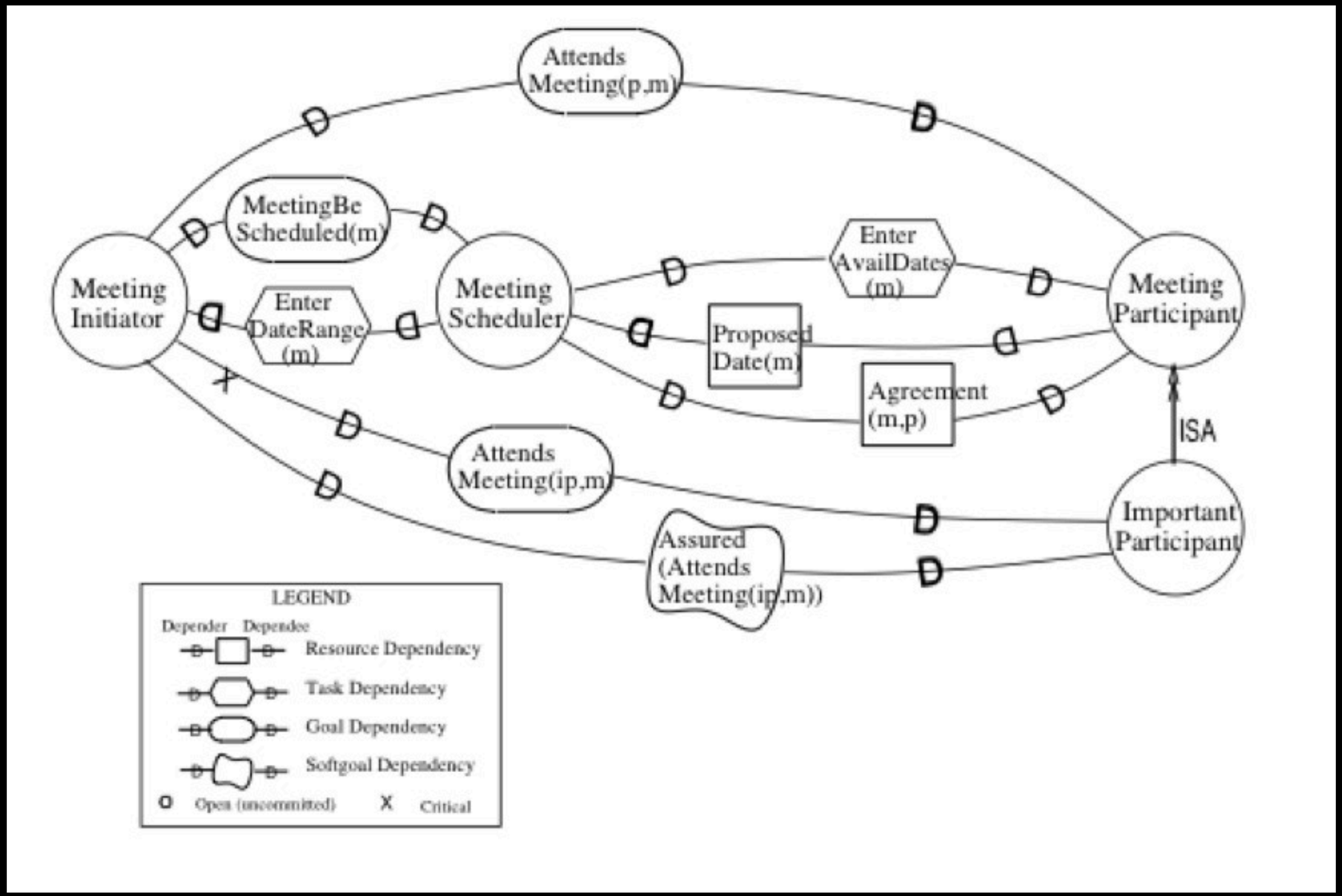
Svarade	Andel
Office (Word, Excel, Visio)	23.8%
None	15.3%
Requisite Pro	10.2%
Quality Center	9.6%
Don't know	5.1%
Focal Point / DOORS	4.0%
Caliber	3.4%
Customer-specific	3.4%
RSA	3.4%
Clear Case	3.4%
Req Test	3.4%
Rest / Other (max 2 mentions per tool)	18.6%

177 tools mentioned in total

I*

- http://www.cs.toronto.edu/km/istar/
- Models Agents and their Intentions
- Early Req Specification together with Customers
- 1. Strategic Dependency Model
  - Actors and Dependencies
  - Certain Actions performed by certain Actors
  - Ex: User depends on system to open door to meet goal to enter building
- 2. Strategic Rationale Model
  - Looks inside actors, what drives them

I* example


Attends Meeting(p,m)
MeetingBe Scheduled(m)
Meeting Initiator
Enter DateRange (m)
Meeting Scheduler
Enter AvailDates (m)
Meeting Participant
Proposed Date(m)
Agreement (m,p)
ISA
Attends Meeting(ip,m)
Important Participant
Assured (Attends Meeting(ip,m))
LEGEND
Depender
Dependee
Resource Dependency
Task Dependency
Goal Dependency
Softgoal Dependency
O Open (uncommitted)
X Critical

Formal languages: Z

- Mathematical language for describing computing system
- Model-based, models abstract data type (ADT)
- ADT = system state and operations on it
  - State = state variables and their values
  - Operation = can change state
- Good match to imperative programming languages
- Also extension for OO languages; form of inheritance
- Very mature, used since 1970's

State Transition Diagram (Z example)

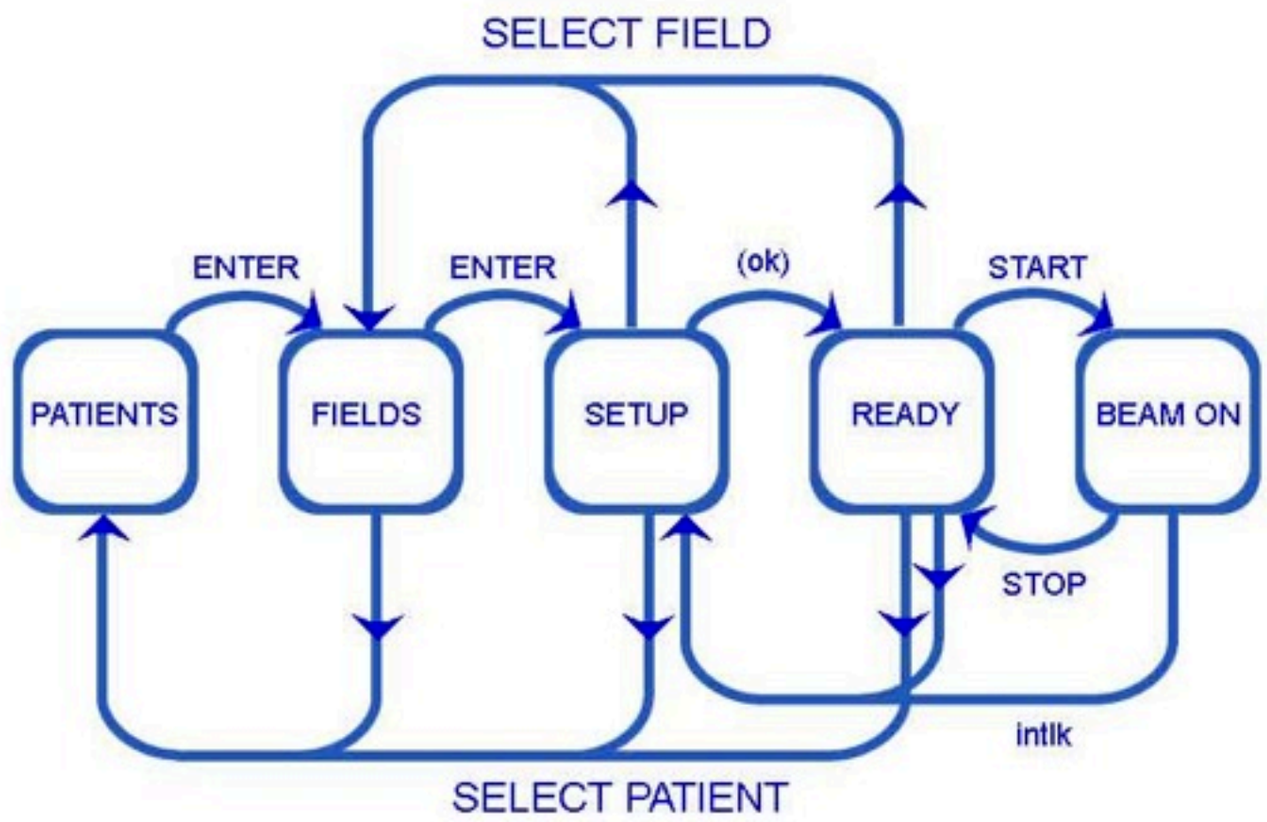


Figure 6.6: Therapy control cascade: state transition diagram

Graphic courtesy of kimberlybatteau.com

From J. Jacky, "The way of Z", chapter 6

State Transition Table (Z example)

	SELECT PATIENT	SELECT FIELD	ENTER	ok	START	STOP	intlk
PATIENTS	---	---	FIELDS	---	---	---	---
FIELDS	PATIENTS	---	SETUP	---	---	---	---
SETUP	PATIENTS	FIELDS	---	READY	---	---	---
READY	PATIENTS	FIELDS	---	---	BEAM ON	---	SETUP
BEAM ON	---	---	---	---	---	READY	SETUP

And now in Z

$STATE ::= patients \mid fields \mid setup \mid ready \mid beam\_on$

$EVENT ::= select\_patient \mid select\_field \mid enter \mid start \mid stop \mid ok \mid intlk$

$FSM == (STATE \times EVENT) \nrightarrow STATE$

$no\_change, transitions, control : FSM$

$control = no\_change \oplus transitions$

$no\_change = \{ s : STATE; e : EVENT \bullet (s, e) \mapsto s \}$

$transitions = \{ (patients, enter) \mapsto fields,$

$(fields, select\_patient) \mapsto patients, (fields, enter) \mapsto setup,$

$(setup, select\_patient) \mapsto patients, (setup, select\_field) \mapsto fields, (setup, ok) \mapsto ready,$

$(ready, select\_patient) \mapsto patients, (ready, select\_field) \mapsto fields, (ready, start) \mapsto beam\_on, (ready, intlk) \mapsto setup,$

$(beam\_on, stop) \mapsto ready, (beam\_on, intlk) \mapsto setup \}$

Non-functional reqs - customer importance?

NFR type	Avg. weight (of 100)	Std.dev.
Usability	23.21	+/- 13.7
Reliability / security	22.79	+/- 10.6
Performance	22.44	+/- 9.4
Stability / Robustness	19.87	+/- 11.5
Maintainability	11.69	+/- 7.1

149 answers from Swedish industry, Spring 2009

SMART NFRs

- NFRs / QRs should be:
  - Specific = without ambiguity, using consistent terminology, simple and at the appropriate level of detail.
  - Measurable = possible to verify req is met. What tests must be performed?
  - Attainable = technically feasible. What is your professional judgement of the technical "do-ability" of the requirement?
  - Realizable = realistic given available resources (skill, staff, schedule etc).
  - Traceable = connected to sources as well as to later dev artefacts.

PLanguage

- Keyword-based language for requirements
- Developed by Tom Gilb, famous SE consultant
- Used in many large corporations
- Often for Quality Requirements: focus on quantification

PLanguage Keywords

TAG	A unique, persistent identifier
GIST	A short, simple description of the concept contained in the Planguage statement
STAKEHOLDER	A party materially affected by the requirement
SCALE	The scale of measure used to quantify the statement
METER	The process or device used to establish location on a SCALE
MUST	The minimum level required to avoid failure
PLAN	The level at which good success can be claimed
STRETCH	A stretch goal if everything goes perfectly
WISH	A desirable level of achievement that may not be attainable through available means
PAST	An expression of previous results for comparison
TREND	An historical range or extrapolation of data
RECORD	The best-known achievement
DEFINED	The official definition of a term
AUTHORITY	The person, group, or level of authorization

Table 2: Sub-keywords for the METER Keyword

METHOD	The method for measuring to determine a point on the Scale
FREQUENCY	The frequency at which measurements will be taken
SOURCE	The people or department responsible for making the measurement
REPORT	Where and when the measurement is to be reported

PLanguage - Additionals

- Fuzzy: <fuzzy concepts>
- Modifiers: Keyword [Qualifier1, Qualifier2, ...]
- Collections: {item1, item2, ...}
- Source for statement: Statement <- source

PLanguage example

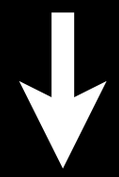
NatLang: "The system must be easy to learn"

↓

StructEnglish: "The system must be used successfully to place an order in under 10 minutes without assistance by at least 80% of test subjects with no previous system experience."

PLanguage example

NatLang: "The system must be easy to learn"



Tag: Learnable
Gist: Ease of learning to use system
Scale: Time for Novice to complete a 1-item order using only onlie help system
Meter: Measurements on 100 novices during UI testing
Must: <7 minutes 80% of the time
Plan: <5 minutes 80% of the time
Wish: <3 minutes 100% of the time
Past [old system]: 11 minutes <- recent site statistics
Novice: **Defined**: A person with <6 months experience with Web applications and no prior exposure to our web application

NFRs in Volere

Volere
Requirements Resources

http://www.volere.co.uk/

HOME | TEMPLATES | RESOURCES | TOOLS | COURSES | SERVICES | CONTACT | SITEMAP

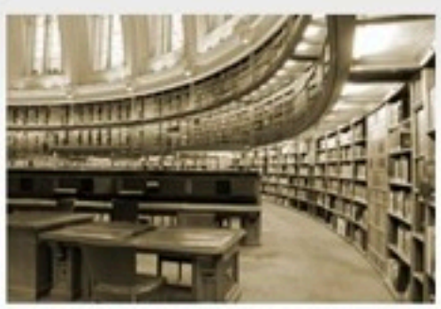
Volere (Voh-lair-ray) the Italian verb to want, or to wish.

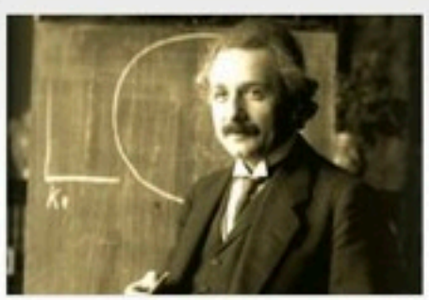
This is the home of the Volere Requirements Specification Template and other requirements and business analysis resources. Volere is the umbrella that covers the collection of requirements templates, processes, books, consulting and training. Since its inception, Volere has been used by thousands of organizations around the world. Read what some of them have to say.

We offer courses including the flagship Mastering the Requirements Process, specification and requirements reviews, requirements process design, and consulting to make you better at requirements and business analysis.

Available downloads are the Volere Requirements Specification Template, Stakeholder Analysis, Prioritisation Analysis and the Atomic Requirement Template, or as it is more popularly known, the Snow Card. Most content is free with registration.

The Volere process provides a well-defined structure and guides as to which requirements content is appropriate for you. The process and template work with existing tools (Caliber, DOORS, Requisite, IRqA, etc.) and methods including agile methods. Volere is applicable to modeling techniques such as UML, business process, data and state models. The process is based on experience from worldwide business analysis projects, and is continually improved with input from our users.





News and Updates

August 2011

A new article Simplicity and Requirements by Suzanne Robertson has been posted.

Volere Events

Brussels, September 13,14. Suzanne Robertson teaches the advanced Mastering the Requirements Process part 2. Please contact IT Works for details. **Sorry, this course is now full.**

Brussels, October 3-5. James Robertson teaches Mastering the Requirements Process. Please contact IT Works for details.

Bonn, October 10-12. The Atlantic Systems Guild presents the Guild Way towards Great Projects. Registrations at SIGS-DATACOM.

London, October 17-19. James Robertson teaches Mastering the Requirements Process. For more information on this popular course, contact IRM UK.

The Netherlands, October 24-26. James Robertson teaches Mastering the Requirements Process. For details and registration, please contact Array Seminars.

Volere

Requirements Specification Template

Edition 15—March 2010

by James & Suzanne Robertson
principals of the Atlantic Systems Guild

The Volere Requirements Specification Template is intended for use as a basis for your requirements specifications. The template provides sections for each of the requirements types appropriate to today's software systems. You may download the template from the Volere site and adapt it to your requirements gathering process and requirements tool. The template can be used with Requisite, DOORS, Caliber RM, IRqA and other popular tools see http://www.volere.co.uk/tools.htm

Contents

Functional requirements are the fundamental or essential subject matter of the product. They describe what the product has to do or what processing actions it must take.

Non-functional requirements are the properties that the functions must have, such as performance and usability. Do not be deterred by the unfortunate name for this kind of requirements, they are as important as the functional requirements for the product's success.

Project constraints are restrictions on the product due to the budget or the time available to build the product.

Design constraints impose restrictions on how the product must be designed. For example, it might have to be implemented in the hand-held device being given to major customers, or it might have to use the existing servers and desktop computers, or any other hardware, software, or business practice.

Project drivers are the business-related forces. For example, the purpose of the project is a project driver, as are all of the stakeholders—each for different reasons.

Project issues define the conditions under which the project will be done. Our reason for including them as part of the requirements is to present a coherent picture of all factors that contribute to the success or failure of the project and to illustrate how managers can use requirements as input when managing a project.

Volere NFRs

- **Look & Feel** - Appearance, Style
- **Usability** - Ease of Use, Personalization/ Internationalization, Learning, Understandability, Accessibility
- **Performance** - Speed & Latency, Safety, Precision/ Accuracy, Reliability, Robustness, Capacity, Scalability, Longevity
- **Operational** - Environment, Adjacent systems, Productization, Release
- **Maintainability** - Maintenance, Supportability, Adaptability
- **Security** - Accessability, Integrity, Privacy, Audit, Immunity
- **Cultural & Legal**

Volere Overall