How to write good requirements
Session 5 of 10
Documenting and storing stakeholders' needs

Version 1.0.1

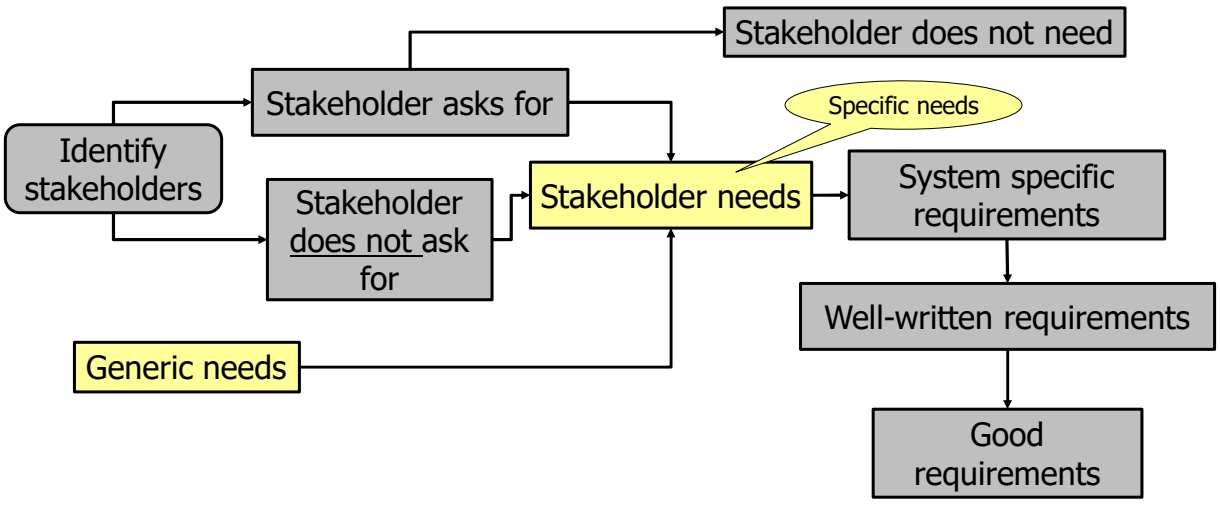
Notice

- This session is being recorded
- By participating in the session, the participants agree :
  1. To be recorded
  2. Prof Kasser has the right, but not the obligation, to make the recording available to the participants
  3. Prof Kasser has the right to use the recordings in other classes (present and future)
- Prof Kasser agrees
  1. To delay publication of the recording for at least 48 hours to allow participants to request, in writing, removal of remarks they made in the session (nature of remarks and approximate timing to be provided with the request).
  2. To remove those requested remarks before publishing the recordings

Course session topics

1. Introduction to requirements
2. Stakeholders and their importance
3. Comminating with the stakeholders
4. Converting stakeholder wants to needs
5. **Documenting and storing stakeholders' needs**
6. Converting stakeholder needs to requirements
7. Converting requirements to well-written requirements
8. Converting well-written requirements to good requirements
9. The use of requirements in the rest of the system development process
10. Summary and closeout

Workflow analysis for good requirements



Objectives of Session 5

1. To present and discuss the BWMQ and exercise presentations from previous session
2. To explain how to document and store the stakeholder needs
3. To provide reasons for documenting needs
4. To explain models and modelling
5. To discuss modeling tools in general
6. To explain some of the different types of models
7. To explain uses, advantages and limitations of models
8. To explain how to create functional models
9. To explain how to communicate functional and operational models to stakeholders
10. To practice creating parts of a model
11. To practice writing requirements traceable to scenarios
12. To provide the opportunity to exercise the 5 levels of knowledge in the updated Blooms taxonomy

Knowledge components

- Lecture
  - Sets the context and provides overview
- Readings
  - None
- Exercises
  - 5-1 Creating scenarios
  - 5-2 Deriving requirements from scenarios
    - The point of this exercise may not be what you think it is
  - 5-3 The real requirement

Session topics

- **BWMQ**
- Exercise presentations
- Documenting needs
- Models
- Modeling tools
- Types of models
- Uses, advantages and limitations of models
- Creating functional models
- Communicating models to stakeholders
- Exercises



BWMQ

- Best
  - Quite interested in learning more about the risk profiles
- Question
  - English doubt: is it "Misuse function" or "Function misuse" scenario"?

Session topics

- BWMQ
- **Exercise presentations**
- Documenting needs
- Introduction to models
- Modeling tools
- Types of models
- Uses, advantages and limitations of models
- Creating functional models
- Communicating models to stakeholders
- Exercises



- NY
  - LM
- EU
  - SZS
  - RJNR
  - IB

Exercise 4-1 scenarios to requirements

1. Update the extracted the scenarios in your Exercise 2-1 presentation
2. Assign an identification to each scenario
3. Add at least one misuse function to at least one scenario
4. Write at least 10 well-written requirements with traceability to any of the scenarios including the scenario(s) updated with the misuse function(s)
5. Prepare a <5 minute presentation containing
   1. The misuse function in its scenario highlighting the modification(s) if any
   2. At least 10 well-written requirements including some traceable to the misuse function
   3. Traceability of each requirement to the source scenario and function
   4. A compliance matrix for the exercise
   5. Formulated problem per COPS problem formulation template
   6. Lessons learned from exercise
6. Save as a PowerPoint file in format Exercise4.1-abcd.pptx
7. Post/email presentation as and where instructed

- NY
  - LM
  - ST
- EU

Exercise 4-2 knowledge reading

1. Prepare a brief on two main points on reading 0402 (< 5min)
2. Presentation to contain
   1. Formulated problem per COPS problem formulation template
   2. A summary of the content of the reading (<1 minute)
   3. The compliance matrix
   4. This slide
   5. The main points (<1 minute)
   6. Brief on one main point (<1 minute per point)
   7. Reflections and comments on reading (<2 minute)
   8. Comparisons of content with other readings and external knowledge
   9. Why you think the reading was assigned to the module
   10. Lessons learned from module and source of learning e.g. readings, exercise, experience, etc. (<2 minutes)
3. Save as a PowerPoint file as Exercise4.2-abcd.pptx
4. Post/email presentation as and where instructed

- NY
- EU
  - RJNR

Exercise 4-3 knowledge reading (optional)

1. Prepare a brief on two main points on reading 0403 (< 5min)
2. Presentation to contain
   1. Formulated problem per COPS problem formulation template
   2. A summary of the content of the reading (<1 minute)
   3. The compliance matrix
   4. This slide
   5. The main points (<1 minute)
   6. Brief on one main point (<1 minute per point)
   7. Reflections and comments on reading (<2 minute)
   8. Comparisons of content with other readings and external knowledge
   9. Why you think the reading was assigned to the module
   10. Lessons learned from module and source of learning e.g. readings, exercise, experience, etc. (<2 minutes)
3. Save as a PowerPoint file as Exercise4.3-abcd.pptx
4. Post/email presentation as and where instructed



Session topics

- BWMQ
- Exercise presentations
- **Documenting needs**
- Introduction to models
- Modeling tools
- Types of models
- Uses, advantages and limitations of models
- Creating functional models
- Communicating models to stakeholders
- Exercises



Why document and store needs

- Documents consensus at a specific time
- Provides traceability
- Provides baseline
  - for the rest of the system development process
- Provides history
  - when dealing with changes
- Provides authoritative information
  - In case of faulty stakeholder memory
  - in case of law suits (risk management)

Where to document and store stakeholder needs

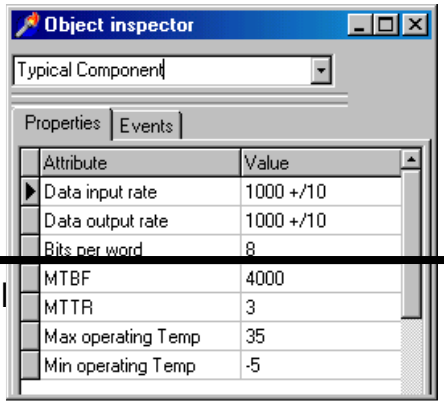
- Text documents
  - Pre 21st century but appropriate under some circumstances
  - With or without graphics
- Engineering drawings
- Concept of operations (CONOPS)
- Use cases or scenarios
- Databases
- Models
- Simulations (executable models)
- PowerPoint presentations
- Others

Documenting non-functional needs

- Non-functional models
- Text statements
- Tables and databases
- Properties and attributes

Functional

Non-functional



Session topics

- BWMQ
- Exercise presentations
- Documenting needs
- **Introduction to models**
- Modeling tools
- Types of models
- Uses, advantages and limitations of models
- Creating functional models
- Communicating models to stakeholders
- Exercises



Models

- "Remember, always, that everything you know, and everything everyone knows, is only a model.
- Get your model out there where it can be viewed.
- Invite others to challenge your assumptions and add their own.
- Instead of becoming a champion for one possible explanation or hypothesis or model, collect as many as possible.
- Consider all of them to be plausible until you find some evidence that causes you to rule one out.
- That way you will be emotionally able to see the evidence that rules out an assumption that may become entangled with your own identity."*

* Donella H. Meadows; Diana Wright. Thinking in Systems: A Primer (Kindle Locations 3073-3076). Chelsea Green Publishing. Kindle Edition

Models

- "Essentially, all models are wrong, but some models are useful."
  - George E. P. Box, Science and Statistics, the Journal of the American Statistical Association Vol. 71, No. 356 (Dec., 1976), pp. 791-799.
  - Referred to statistical and analytical models
- Statistical models contain three basic types of assumptions
  1. Assumptions about the distribution of values in a variable
  2. Assumptions about the functional relationship between variables
  3. Assumptions about the probabilities
- Assumption
  - What applies to statistical models applies to all other models
  - Other types of models contain other types of assumptions

Models and representations

- A model is an abstract representation of an existing part of reality or a part of an imagined reality
  - To be created
  - To use as background to a story
- May need more than one model to develop an understanding (HTP)
  - Allow us to engineer things
    - E.g. wave and particle theory of electromagnetic propagation
- Various types of models/representations, including
  - Breadboards for form and fit
  - Relationships for function
  - Performance
  - Physical mock ups

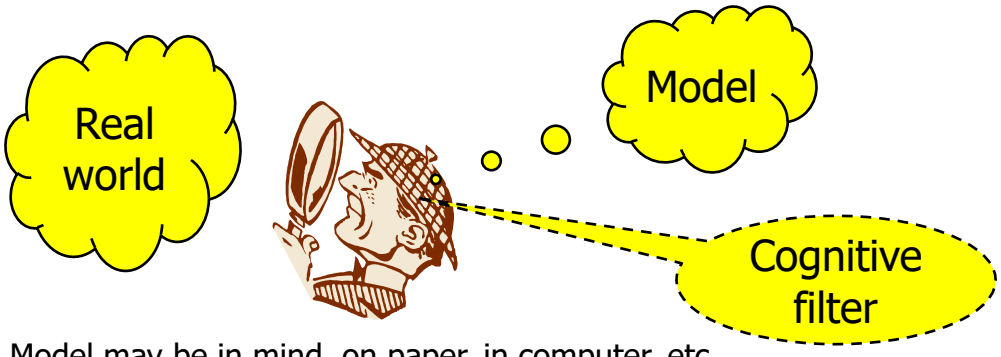


- **Be aware of the assumptions underlying a model before reuse**

Choice of models

- Models reflect understanding of situation
  - Current situation and CONOPS
- There may be more than one way to express or model the understanding
  - E.g. travelling models
    1. Half journey steps (never arrives, but gets exasperatingly asymptotically close)
    2. Whole journey in one step
- If model becomes objectively or subjectively complex, rethink!
- Reduce objective complexity
  - Principle of Hierarchies
  - Atomic functions in scenarios
- Subjective complexity
  - Understanding may be incomplete or incorrect

Building models



- Model may be in mind, on paper, in computer, etc.
- Cognitive filters (Holistic Thinking page 22) are one cause of the model diverging from the real world
- Models based on observation cannot be validated without a reference
- Generic flaw in Standards

Session topics

- BWMQ
- Exercise presentations
- Documenting needs
- Introduction to models
- **Modeling tools**
- Types of models
- Uses, advantages and limitations of models
- Creating functional models
- Communicating models to stakeholders
- Exercises



Modelling tools – Be aware

- Tools need to be simple
- Tools need to help the user
- Tools need to be cost effective
- Tools are dumb
- Tools need to be used intelligently
- Picking the right tool for the job is critical
- There isn't a single tool that will solve all problems (*Continuum* HTP)
- Tools don't solve problems, people do, using tools

Deploying tools

- Introducing new tools is covered in literature on organizational change
- May require a change in the organization
- Implement a paradigm change
- Find early adopters of technology
- Support them
- Management commitment is critical
- Show benefits periodically

Compatibility issues

- Types of tools
- Types of knowledge
- Representation of knowledge
- Communications between tools
- Different tools in
  - Customer-contractor
  - Prime contractor-subcontractor
  - Within distributed organizations
  - Other places

Session topics

- BWMQ
- Exercise presentations
- Documenting needs
- Introduction to models
- Modeling tools
- **Types of models**
- Uses, advantages and limitations of models
- Creating functional models
- Communicating models to stakeholders
- Exercises



Some types of models

- Context models
- Structural models
- Function and flow models
- Pace layer model
- KANO model of customer satisfaction
- Financial models
- Others

Context models

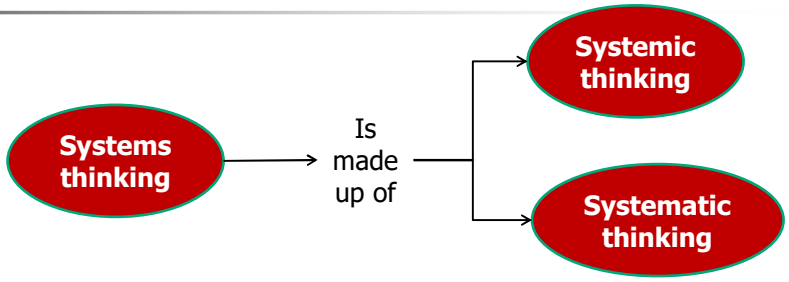
- *Big Picture* and *Operational*
- Describe a system and the adjacent systems that interact with the system
- Sketch the interfaces between a system and its context (e.g., in terms of what information is exchanged)
- Contain context diagrams
  - Used for expressing use cases or scenarios
  - Have no standardized notation
  - UML and SysML used by INCOSE

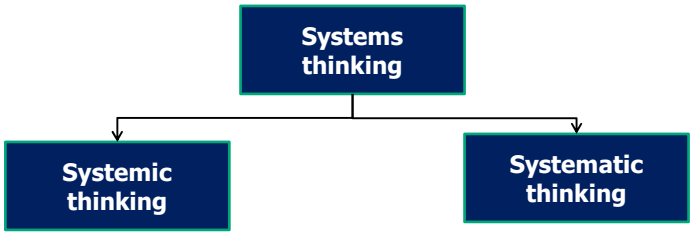
Structural models

- Physical (including software objects) representations
- Architectural drawings
- Wiring diagrams
- Plumbing diagrams
- Electrical schematics
- Physical scale models
- Etc.

Two *Structural* representations of systems thinking

- Do not combine *Structural* and *Functional* relationships in the same graphic
- Use two graphics
- Using a single graphic creates artificial complexity





Functional models and simulations

- A simulation is an executable model
- Functional models and simulations focus on the 'how' it is being done
  - Useful when the underlying mechanisms are well-understood and the functionality can be expressed mathematically.
  - When the underlying mechanisms in an unprecedented system are unknown, then using simulations as training tools can be downright dangerous.
- The model and simulation is only as good as its underlying assumptions, and when they are wrong, people can die.

Operational models and simulations

- Operational models and simulations focus on actual and conceptual views of a system:
  - what the system is doing, what the system can do, what the system should do and what the system needs to do
  - models can be used to gain consensus on the 'what' aspect of a conceptual system
- The model and simulation is only as good as its underlying assumptions, and when they are wrong, people can die

Simulations (executable models)

- *Functional*, *Operational* and *Quantitative* HTPs
- Relationships
- Effects
- Dynamic
- Games
- Training
- Others



Function and flow models

- Models that focus on function and flow aspects specify
  - Requirements for the sequence of actions required to produce the required results from given inputs
  - The actions required to execute a (business) process, including the flow of control and data between the actions and who is responsible for which action.
- Activity models are used to specify system functions
- Activity diagrams can also express who is responsible for which action
- Process models are used to describe business processes or technical processes
- Domain story models
  - Specify visual stories about how 'actors' interact with devices, artifacts and other items in a domain
  - Are a means for understanding the application domain in which a system will operate.

Advantages of use of functions

- Forces stakeholders to think in abstract terms in the early stages of identifying a problem and providing a solution
  - State problem in terms of functions
- Can produce solutions that are innovative and more complete than thinking in 'implementation' terms
- Using a standard functional template for a system can help maximize completeness of the resulting system
  - Functions can be inherited from standard patterns for class of system
  - It is easier to identify missing functions in system functional descriptions than in implementation descriptions
- A system in its functional form can be modelled at design time to determine how well the solution functionality remedies the problem

Functional approach

1. **Current functions** = functionality provided in the existing situation which may range from zero (nothing exists) to some functionality in an existing system deemed as not providing a complete solution
2. **Functions performed by solution system** = desired functions (to be developed) **+** good current functions **–** bad current functions
3. **Functions performed to realize the solution system** = the system development process

- and,
  - functions performed by the solution system and the functions performed to realize the solution system both consist of mission and support functions

Pace Layer model*

- In any complex system, there are different layers or components that change at different rates or paces. These layers range through
  1. Fast
     - Includes fashion, versions of technological products (cell phones, software product versions), teenager's moods
  2. Medium
     - Includes management and engineering fads; e.g. Management by Objectivities, Business Process Reengineering, Model-based Systems Engineering, Network-centric-Warfare, (Systems Engineering ?), computer operating systems
  3. Slow
     - Long term stability, e.g. building and logical infrastructure
- Relevance to requirements elicitation and elucidation
  1. Probability of the stability of the requirement (how quickly might it change)
  2. May be more applicable in the Subsystem Design States of the System Development Process

* Brand, S., The Clock Of The Long Now: Time and Responsibility, Phoenix Paperbacks, 2000

KANO model*

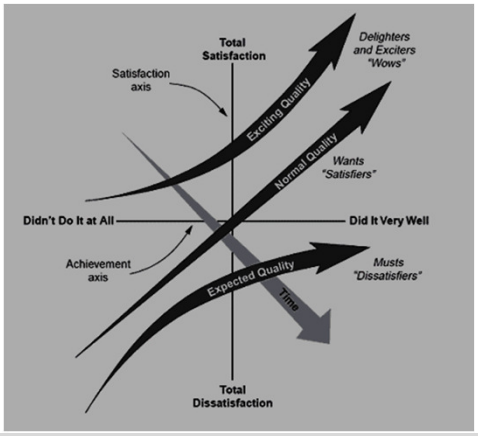
- A model of customer satisfaction with a simple ranking scheme to distinguish between essential and differentiating attributes related to concepts of customer satisfaction**
- Applicable to new product development for a mass market
- Two dimensions
  1. Achievement
     - Ranges from (low) the supplier didn't do it at all, to the supplier did it very well (high)
  2. Satisfaction
     - Ranges from total dissatisfaction with the product or service (low), to total satisfaction with the product or service (high)

** Kano, Noriaki; Nobuhiku Seraku; Fumio Takahashi; Shinichi Tsuji (April 1984). "Attractive quality and must-be quality". Journal of the Japanese Society for Quality Control 14(2), 147-156, 1984-04-15, The Japanese Society for Quality Control*
** ChatGPT, accessed 8 October 2023

Kano Model: Types of customer requirements

- Different versions, different terminology

1. The five–level version (original by Kano)
   1. Musts (expected)
   2. One-Dimensional (wants)
   3. Delighter (exciting)
   4. Don't care
   5. Reverse (dissatisfiers)
2. A three-level version of needs*
   1. Expected (musts)
   2. Normal needs (wants)
   3. Exciting needs (delighters or wows)



* https://asq.org/quality-resources/kano-model

Highlights show comparison between versions

The five levels-1, 2, 3 of 5

1. Musts (Expected)
   - Basic requirements or features that customers expect as a minimum
   - Leads to dissatisfaction if they are missing, but their presence doesn't significantly increase customer satisfaction because they are considered basic and essential.
2. One-Dimensional (wants)
   - Features that linearly correlate with customer satisfaction.
   - Features that directly impact customer satisfaction in a predictable way
   - The more you improve these features, the more satisfied customers become
3. Delighter (Exciting)
   - Features that were not asked for but when present, significantly enhance customer satisfaction
   - Their absence does not necessarily lead to dissatisfaction
   - Customers may not even be aware they want these features until they experience them

The five levels- 4, 5 of 5

4. Don’t care
   - Features that don't strongly influence customer satisfaction one way or the other
   - Their presence or absence has no influence on customer satisfaction
5. Reverse (dissatisfiers)
   - Features that, when present in excess, can actually lead to dissatisfaction
   - For example, if a product has too many complicated options and settings, it may overwhelm customers and decrease satisfaction

The three-level version-1*

1. Expected needs (must)
   - Fully satisfying the customer at this level simply gets a supplier into the market.
   - The entry level expectations are the *must* level qualities, properties, or attributes
   - These expectations are also known as the *dissatisfiers* because by themselves they cannot fully satisfy a customer
   - However, failure to provide these basic expectations will cause dissatisfaction
   - The *musts* include customer assumptions, expected qualities, expected functions, and other *unspoken* expectations

* https://asq.org/quality-resources/kano-model

The three-level version-2

2. Normal needs (wants)
   - the qualities, attributes, and characteristics that keep a supplier in the market
   - These next higher level expectations are known as the wants or the satisfiers because they are the ones that customers will specify
   - They can either satisfy or dissatisfy the customer depending on their presence or absence
   - The wants include voice of the customer requirements and other spoken expectations

The three-level version-3

3. Exciting needs (delighters or wows)
   - These are features and properties that make a supplier a leader in the market
   - The highest level of customer expectations, as described by Kano, is termed the *wow* level qualities, properties, or attributes
   - These expectations are also known as the *delighters* or *exciters* because they go well beyond anything the customer might imagine and ask for
   - Their absence does nothing to hurt a possible sale, but their presence improves the likelihood of purchase
   - *Wows* not only excite customers to make on-the-spot purchases but make them return for future purchases
   - These are *unspoken* ways of delighting the customer
   - Over time, *wows* become *wants* become *musts*

Application to requirements elicitation and elucidation

- "Musts" become high priority "needs"
- Delighters and Reverses might be identified by asking stakeholders for "wishes"
- Delighters, when identified, become low priority "needs"
- Delighters, which show up when system is in use become "musts "or "wishes" for upgrades or copies

Session topics

- BWMQ
- Exercise presentations
- Documenting needs
- Introduction to models
- Modeling tools
- Types of models
- **Uses, advantages and limitations of models**
- Creating functional models
- Communicating models to stakeholders
- Exercises



IREB use of models*

- Specifying (primarily functional) requirements in part or even completely, as a means of replacing textually represented requirements.
- Decomposing a complex reality into well-defined and complementing aspects; each aspect being represented by a specific model.
- Paraphrasing textually represented requirements in order to improve their comprehensibility, in particular with respect to relationships between them.
- Validating textually represented requirements with the goal of uncovering omissions, ambiguities, and inconsistencies.

The systems approach uses functional models in the form of the CONOPS prior to writing the system and process requirements

* IREB, 3.10, EU 3.4.1

Models (advantages and limitations)*

- Advantages
  - Relationships and interconnections between requirements are easier to understand with graphic models
  - Focusing on a single aspect reduces the cognitive load for understanding the modeled requirements
  - Requirements modeling languages have a restricted syntax which reduces possible ambiguities and omissions
- Limitations
  - Keeping [different] models that focus on different aspects consistent with each other is challenging
  - Information from different models needs to be integrated for causal understanding
  - Models focus primarily on functional requirements; most quality [non-functional] requirements and constraints cannot be expressed in models with reasonable effort
  - The restricted syntax of a graphic modeling language implies that not every relevant item of information can be expressed in a model

* IREB, 3.10, EU 3.4.1

Session topics

- BWMQ
- Exercise presentations
- Documenting needs
- Introduction to models
- Modeling tools
- Types of models
- Uses, advantages and limitations of models
- **Creating functional models**
- Communicating models to stakeholders
- Exercises



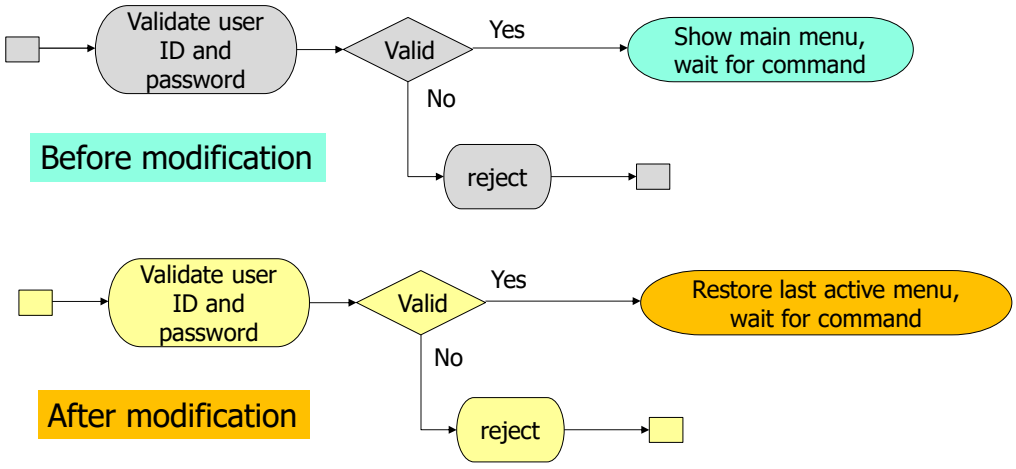
Building a functional conceptual needs model

- Use current situation as pre-baseline
- Identify objects and relationships in current situation
- Modify to remove undesirability
- Add new desired functionality
- Create scenarios
- Decompose scenarios into functions (Session 4)
- Quantify each function
- Prioritize each function
- Create relationship diagrams
- Combine relationship diagrams
- Use the three ways to maximize completeness (Session 4)
  - Corresponding inputs and outputs
  - Corresponding functions

Example: "Wants" to scenarios-1

- Generic stakeholder role
  - Faculty
- Specific stakeholder
  - Professor Joseph
- Wanted improvement
  - "when I log in to the SECTS, I want to be taken to the place I was at when I logged out of my previous session"
- Question (*Generic* HTP)
  - Should this apply to all users or just faculty?
- Answer
  - Make it an option, default to "on" for faculty (implementation domain knowledge) unless other stakeholders have asked for it, or want it when they hear about it
  - Discuss with other stakeholders before SRR

Logging in scenario (focused)



Example: Wants to scenarios-2

- Scenarios affected
  - (applying *Three ways to maximize completeness* from Session 4)
  1. User logs into the system (stakeholder want)
     1. Initial menu changed to last active menu
  2. User logs out of the system
     1. Must remember active menu
  3. New user set up
     1. Must set up option
  4. Disconnect without logging out in all other scenarios
     1. Must remember last active menu
- Other system attributes affected if any
  - Must remember user's active menu each time a menu is shown
  - Oops, just designed the change in other scenarios

Session topics

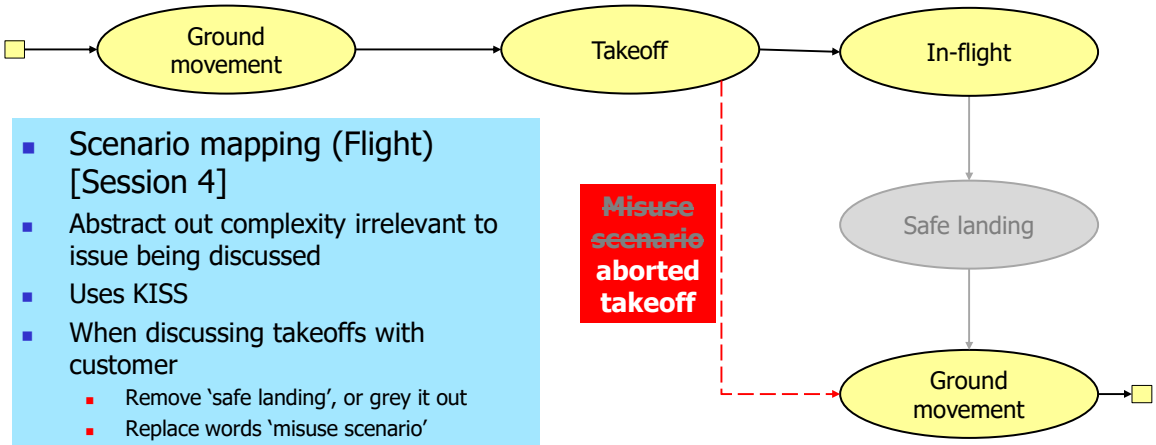
- BWMQ
- Exercise presentations
- Documenting needs
- Introduction to models
- Modeling tools
- Types of models
- Uses, advantages and limitations of models
- Creating functional models
- **Communicating models to stakeholders**
- Exercises



Communicating models to stakeholders

- Don't !!!!!!!!!!
- Don't expect the stakeholders to learn a modelling language or understand the graphics produced by your modelling tools
- Communicate the relevant data with the rest abstracted out (next slide)
  - Using the Principle of Hierarchies in Reading 0302
- Use customer's language to represent the data in the model
  - Simple connected blocks and circles
  - Sketches, Rich Pictures (Checkland)
  - Videos and animations
  - Pictures
  - Others
- Use the eight tools to overcome the barriers from Session 3 as appropriate

Show something like this



- Scenario mapping (Flight) [Session 4]
- Abstract out complexity irrelevant to issue being discussed
- Uses KISS
- When discussing takeoffs with customer
  - Remove 'safe landing', or grey it out
  - Replace words 'misuse scenario'

Session topics

- BWMQ
- Exercise presentations
- Documenting needs
- Introduction to models
- Modeling tools
- Types of models
- Uses, advantages and limitations of models
- Creating functional models
- Communicating models to stakeholders
- **Exercises**



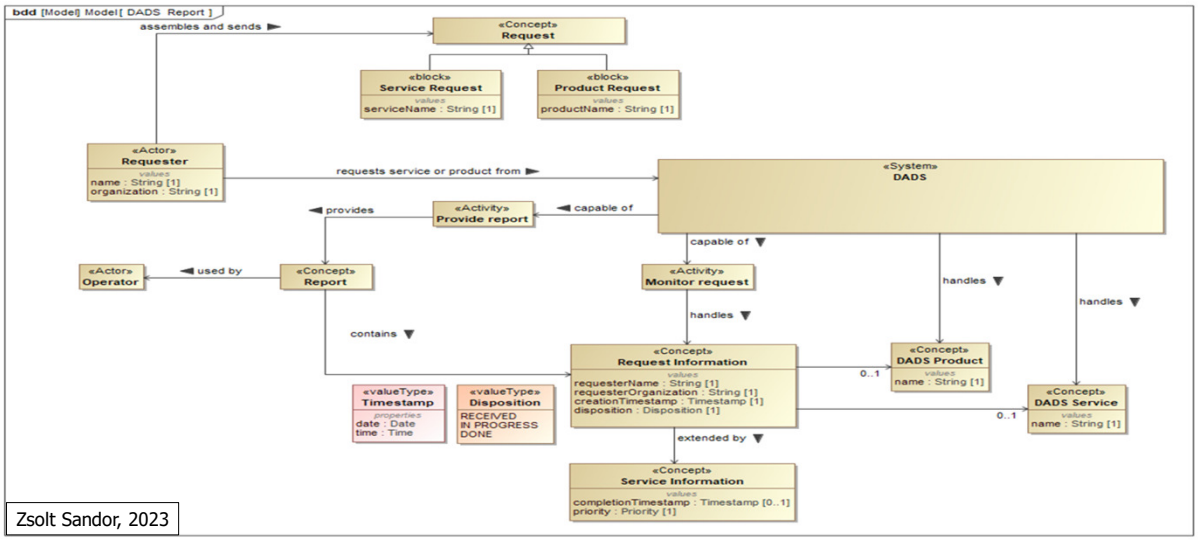
Exercise 5-1 Creating scenarios

1. Prepare a <5 minute presentation containing
    1. Convert any 10 of the responses from the stakeholder in Exercise 3-1 to a set of linked scenarios
    2. Identify the mission, support and risk prevention/mitigation functions (as appropriate)
    3. A higher level drawing (or N^2 chart) showing the scenarios and the interfaces between the scenarios
    4. The exercise problem formulated per COPS problem formulation template
    5. A compliance matrix for the exercise
    6. Lessons learned from exercise
2. Save as a PowerPoint file in format Exercise5.1-abcd.pptx
3. Post presentation as and where instructed

Exercise 5-2 Deriving requirements from scenarios

1. Write a complete set of well-written requirements traceable to the scenario in the next slide (as complete as you can, within a 60 minutes time limit)
2. Prepare a <5 minute presentation containing
    1. The well-written requirements
    2. Identify the mission, support and risk prevention/mitigation requirements (if any)
    3. If there is anything in the scenario that does not generate a requirement, discuss the reason(s) (maximum of two)
    4. The exercise problem formulated per COPS problem formulation template
    5. A compliance matrix for the exercise
    6. Lessons learned from exercise (<3 min)
    7. Why you think this exercise was included in the lesson
3. Save as a PowerPoint file in format Exercise5.2-abcd.pptx
4. Post presentation as and where instructed

Exercise 5-2 Scenario



Exercise 5-3 Background

- Imagine you have just watched 5 student presentations
- The attributes of the presentations are shown in compliance matrix format
- All 5 presentations comply with the 4 requirements for the exercise
- 4 presentations contain some additional attributes

Attributes of exercises	Presentations				
	A	B	C	D	E
Exercise requirement 1	√	√	√	√	√
Exercise requirement 2	√	√	√	√	√
Exercise requirement 3	√	√	√	√	√
Exercise requirement 4	√	√	√	√	√
Slides are in a logical order	√			√	
Title and presenter's name slide	√			√	
Colorful theme	√	√	√		
Closing slide (any questions)	√		√	√	

Exercise 5-3 the real requierment

1. Rank the presentations (A-E) in order of your preference from liking most to liking least
2. Prepare a <5 minute presentation containing
    1. A sorted list of the presentations in order of preferences (high to low)
    2. The reasons for the ranking (sorted order)
    3. A compliance matrix for the exercise
    4. Lessons learned from exercise
    5. The exercise problem formulated per COPS problem formulation template
3. Save as a PowerPoint file in format Exercise5.3-abcd.pptx
4. Post presentation as and where instructed

Meeting the objectives

#	Objectives	Met
1	To present and discuss the BWMQ and exercise presentations from previous session	8-13
2	To explain how to document and store the stakeholder needs	16-17
3	To provide reasons for documenting needs	15
4	To explain models and modelling	19-22
5	To discuss modeling tools in general	24-26
6	To explain some of the different types of models	28-46
7	To explain uses, advantages and limitations of models	48-49
8	To explain how to create functional models	51-54
9	To explain how to communicate functional and operational models to stakeholders	56
10	To practice creating parts of a model	58
11	To practice writing requirements traceable to scenarios	59
12	To provide the opportunity to exercise the 5 levels of knowledge in the updated Blooms taxonomy	58-62

Objectives of session 6

1. To present and discuss the BWMQ and exercise presentations from previous session
2. To explain what to do before writing the requirements
3. To explain what to do while writing the requirements
4. To explain what to do immediately after writing the requirements
5. To explain how to deal with some types of poorly written requirements
6. To discuss attributes of requirements for flexible systems
7. To provide the opportunity to use a software tool to convert some poorly-written requirements into well-written requirements
8. To provide the opportunity to exercise the 5 levels of knowledge in the updated Blooms taxonomy

Any questions ?

1. Best
2. Worst
3. Missing
4. Questions



Email:
beyondsystemsthinking@yahoo.com
Subject: <class title> BWMQ Session #