

Software Requirements

L10: Requirements Negotiation and Prioritization

Prof. W. Maalej – @maalejw

Overview

1

Requirements Negotiation

2

Negotiation Techniques

3

Requirements Prioritization

4

Prioritization Techniques

What is requirements negotiation?

“A **collaborative** approach for **resolving conflicts** by the exploration of range of possibilities. It is characterized by the participants attempting to **find a settlement** that **satisfies all parties** as much as possible”

[Easterbrook, 1991]



Goals of requirements negotiation

In theory

- To make the **conflicts** explicit
- To develop **alternatives**, and **arguments** for and against each alternative
- To facilitate in that “right” **decision** is made

In practice

- Either one of the conflicting parties is educated to the level when it changes its mind
- Or a compromise is found

Requirements negotiation benefits

- Understanding project constraints
- Finding “better” solutions
- Managing complexity
- Externalizing **tacit knowledge**
- Fostering team learning
- Adapting to **changes**
- Dealing with **uncertainty**

Adapting to changes

- Requirements are highly volatile
- Stakeholders are forced to frequently adopt to new situations
- Negotiation raise awareness on existing issues and alternatives



Dealing with Uncertainty

- Stakeholders do not know their needs and what is technologically **feasible**
- Negotiation helps to **reduce uncertainty**, by highlighting things needing attention
- Negotiation **fosters a shared vision** among stakeholders



What stakeholders need to know about requirements?

- Online survey
- 475 responses
 - 307 at least half answered
 - 262 full answers
- Quantify information needs and the tool support

Information Needs in Requirements Engineering

0%  100%

I. Information Needs

Which information needs do you have in the following situations?
How well are these needs supported by the current tools you are using?

* 1. When I'm defining new requirements, I need to know... / my tool support is...

	I need to know...					My tool support is...				
	Never	Rarely	Sometimes	Often	Usually	Absent	Poor	OK	Good	Excellent
Are there other redundant requirements?	<input type="radio"/>									
Are there related requirements, of which I should be aware (e.g. "refines", "is a part of", "depends on", "incompatible" relations) or which I can reuse?	<input type="radio"/>									
With which topic , category, keywords, should I annotate this requirement?	<input type="radio"/>									
Am I using the right vocabulary , or should I rename particular keywords?	<input type="radio"/>									
Am I using the right templates ?	<input type="radio"/>									
With whom I should discuss the new requirement?	<input type="radio"/>									
How would other stakeholders assess the quality of the new requirement (concreteness and understandability)?	<input type="radio"/>									
How would other stakeholders estimate the feasibility of the new requirement?	<input type="radio"/>									

Others:

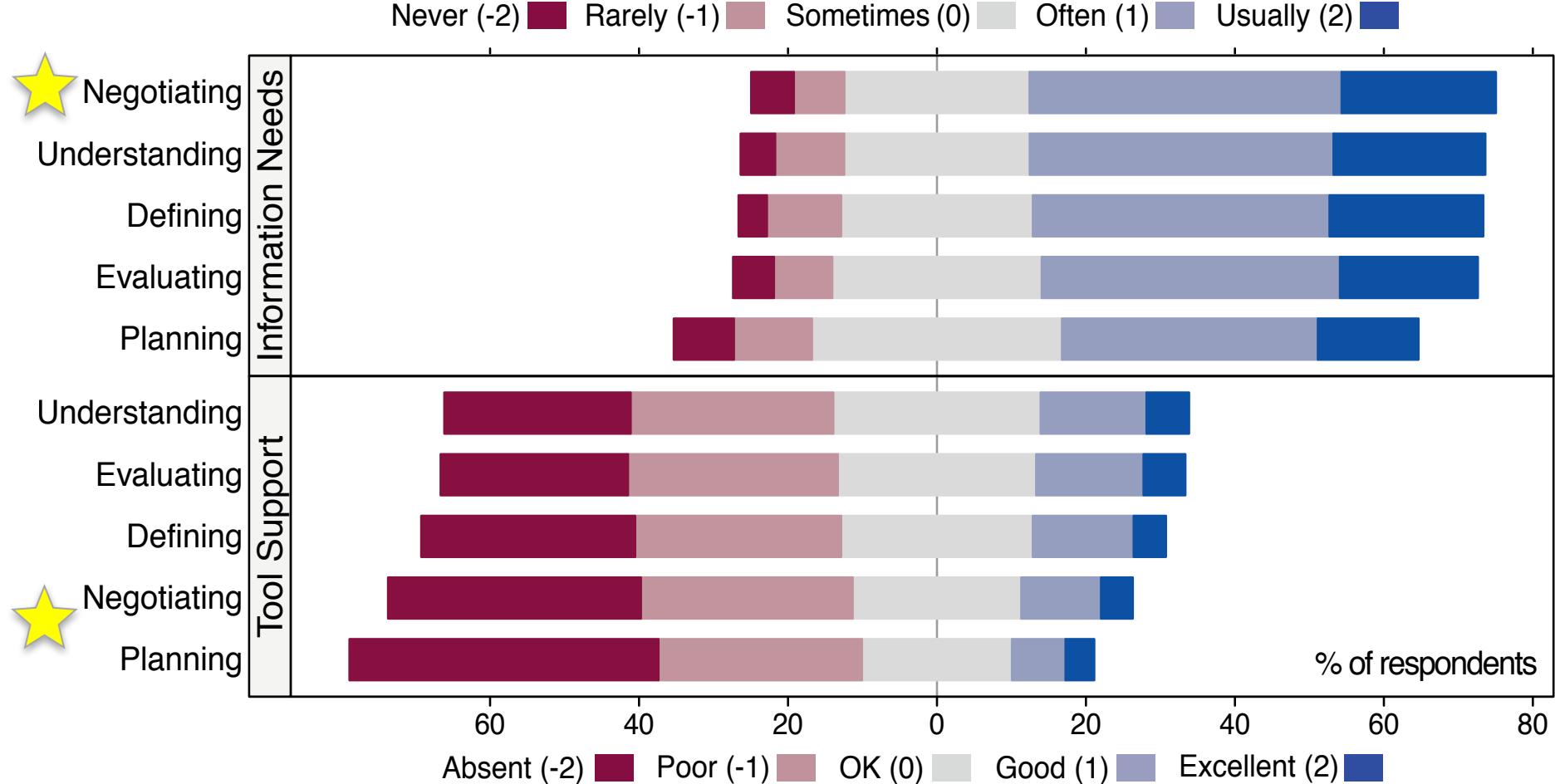
Others:

* 2. When I'm trying to understand other's requirements, I need to know... / my tool support is...

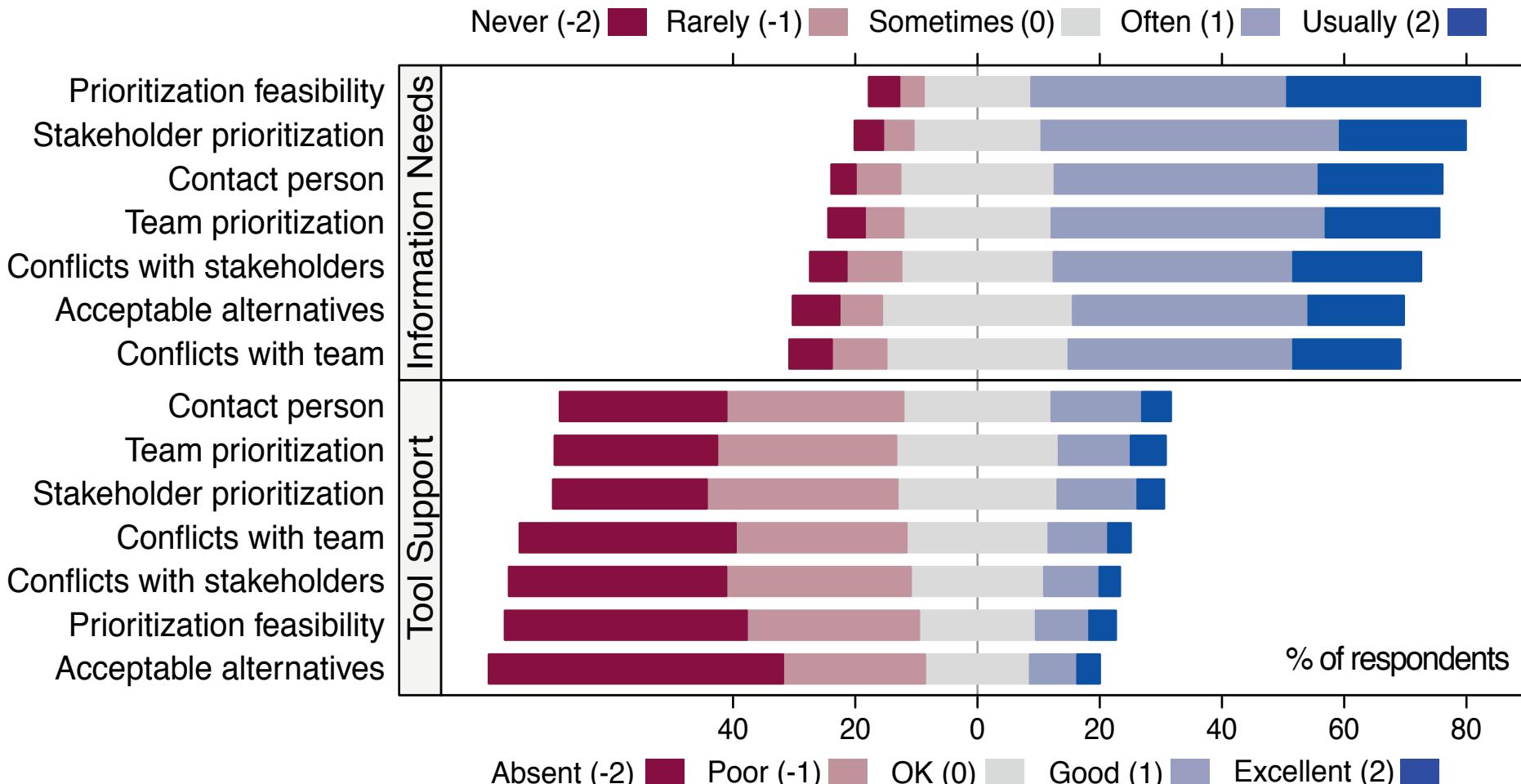
	I need to know...					My tool support is...				
	Never	Rarely	Sometimes	Often	Usually	Absent	Poor	OK	Good	Excellent
What was the stakeholder's intention when defining this requirement?	<input type="radio"/>									
What are related requirements?	<input type="radio"/>									
Which other artifacts do I need to look at (models, source	<input type="radio"/>									

[Maalej et al., 2014]

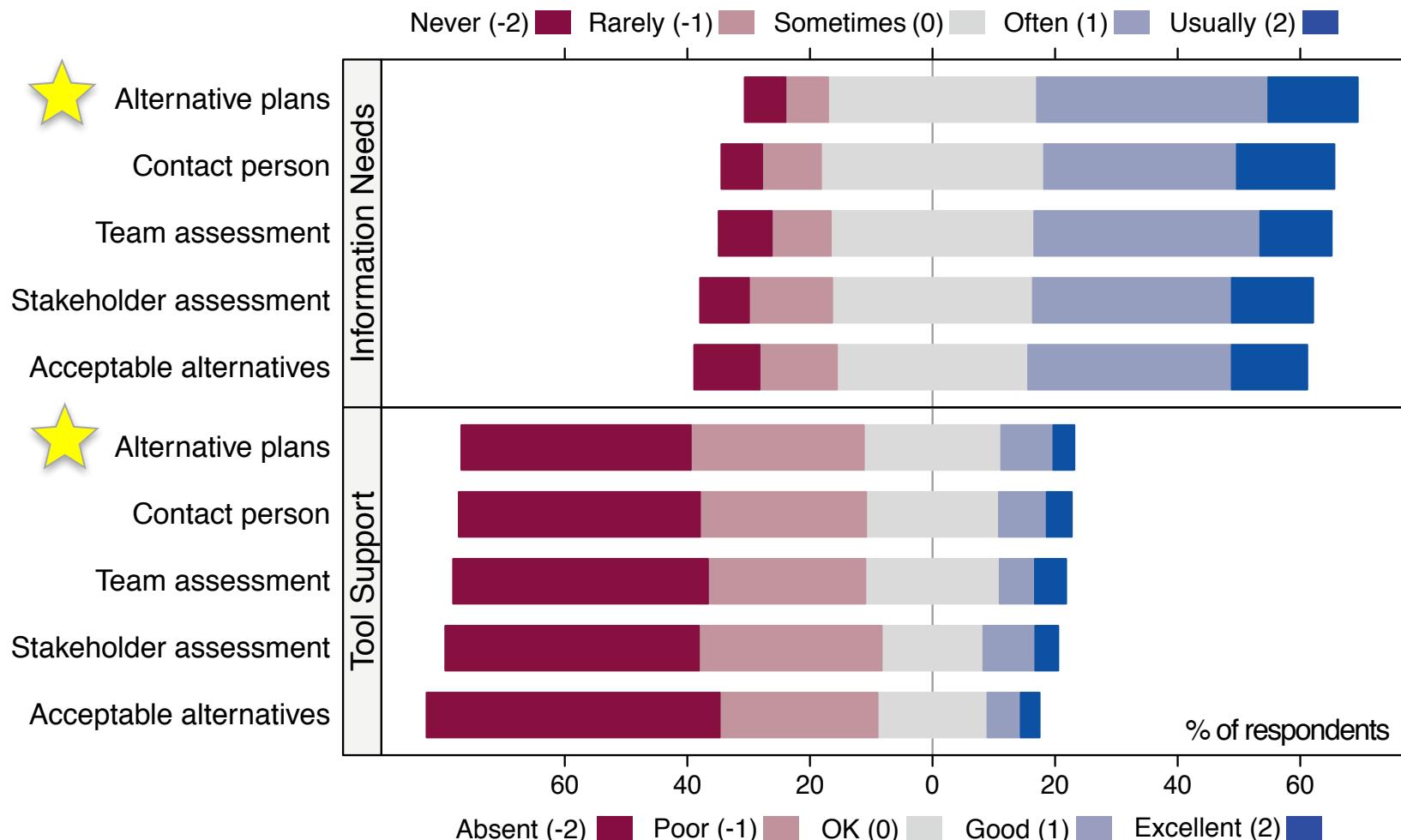
Information needs and tool support during requirements related tasks



Information needs when negotiating requirements



Information needs when planning requirements



Overview

1

Requirements Negotiation

2

Negotiation Techniques

3

Requirements Prioritization

4

Prioritization Techniques

Negotiation techniques



- Group decisions (social dynamics)
 - Majority voting
 - Fairness/ Maximum benefits
 - Committee decides



Rationale thinking (rationale management)



- Consensus building (WinWin)
- Consistency with goals (NFR Framework)



Priority-based



What is rationale?

Rationale is the reasoning that lead to decisions (about the system).

Rationale includes:

- The **issues** that were addressed
- The **alternatives** that were considered
- The **decisions** that were made to resolve the issues
- The **criteria** that were used to guide decisions
- The **debate** stakeholders went through to reach a decision

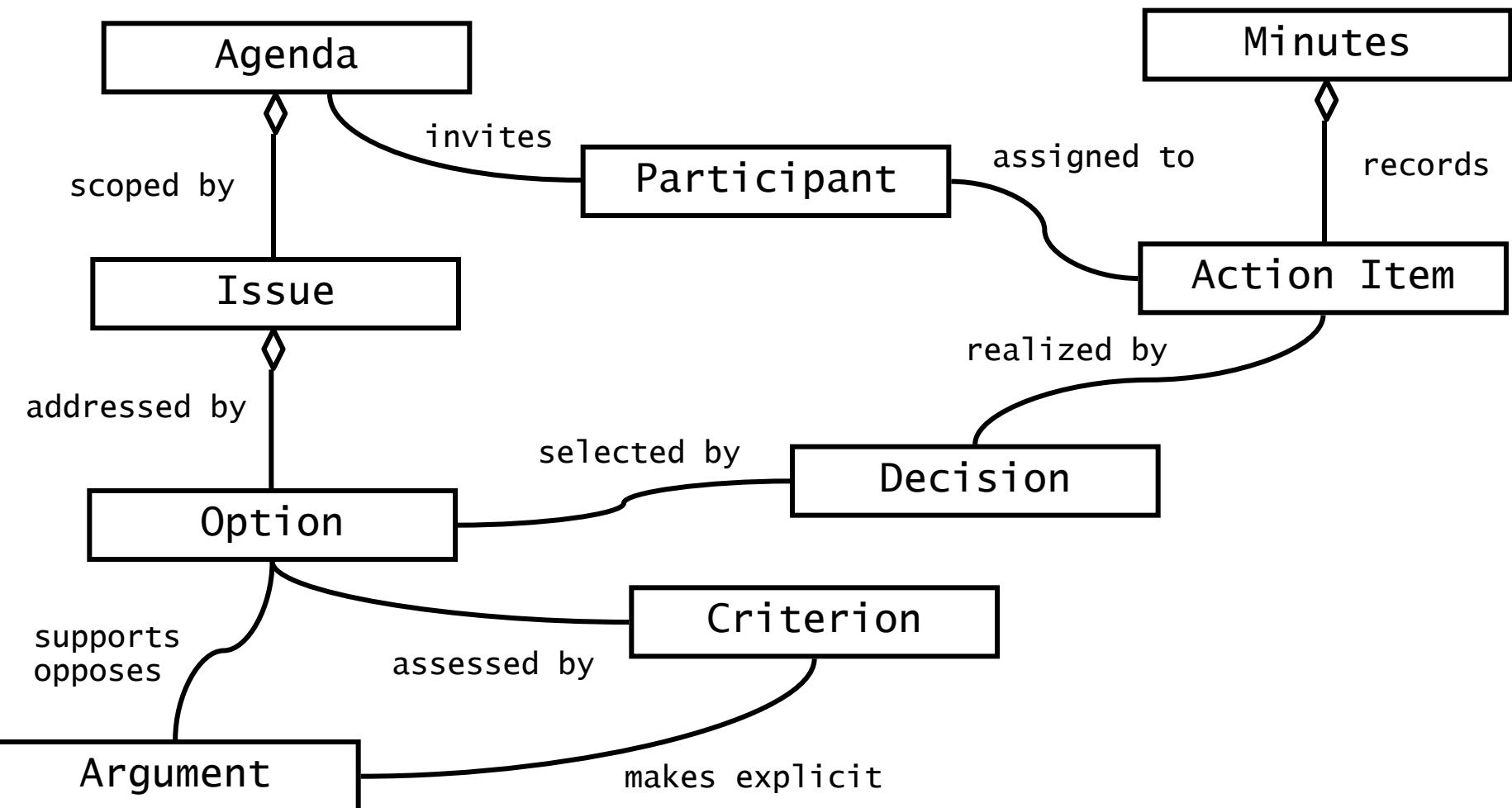
Debates often take place in meetings



Meeting ABC

- Keep meetings short
 - Define scope of meeting before hand
 - Focus only on issues within scope
- Invite the right participants
 - Do not invite participants who cannot contribute
 - Do invite participants whose agreement is needed
- Record to agreements for closed items
 - Ensure that all remember the same outcome
- Record context for open items
 - Shorten follow up discussions on the same theme

Meeting concepts



ATM example

Question: Alternative Authentication Mechanisms?

References: Service: Authenticate

Decision: Smart Card + PIN

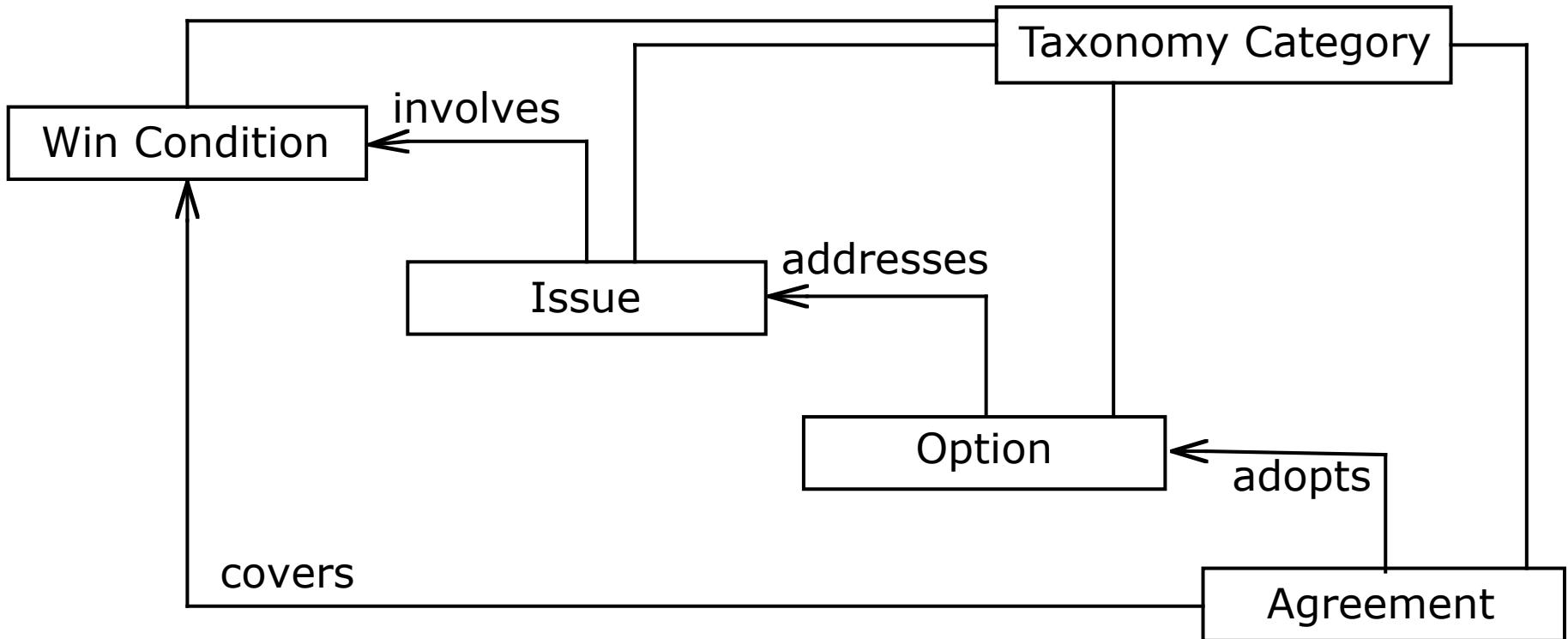
	Criteria 1: ATM Unit Cost	Criteria 2: Security
Option 1: Account number	+	-
Option 2: Finger print reader	-	+
Option 3: Smart Card + PIN	+	+

Consensus building: WinWin

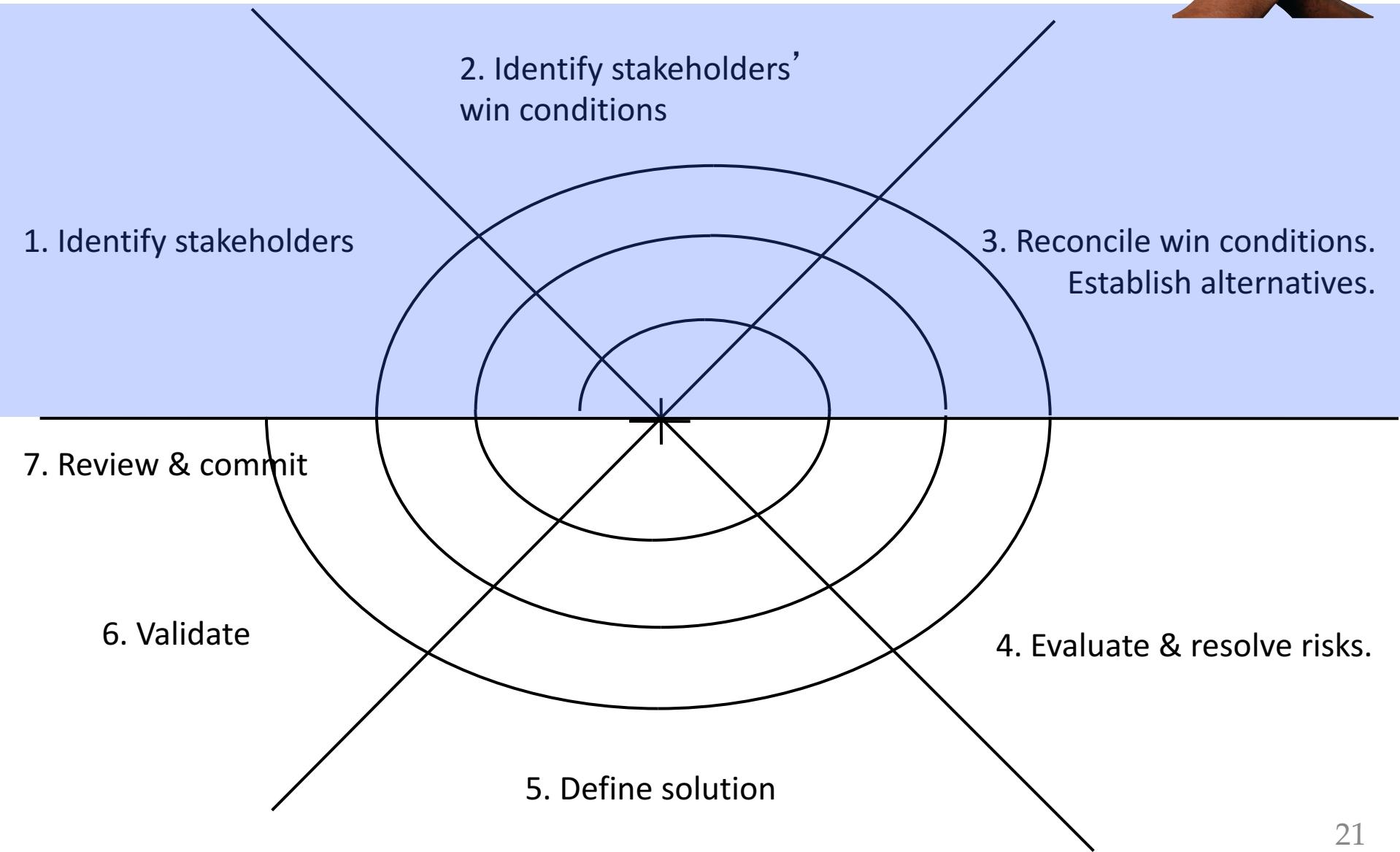
- Incremental, risk-driven spiral process:
 - Identification of stakeholders
 - Identification of win conditions
 - Conflict resolution
- Asynchronous groupware tool
 - Stakeholders post win conditions
 - Facilitator detects conflict
 - Stakeholders discuss alternatives
 - Stakeholders make agreements



Consensus building: model



Consensus building: process



Consensus building: easy WinWin tool

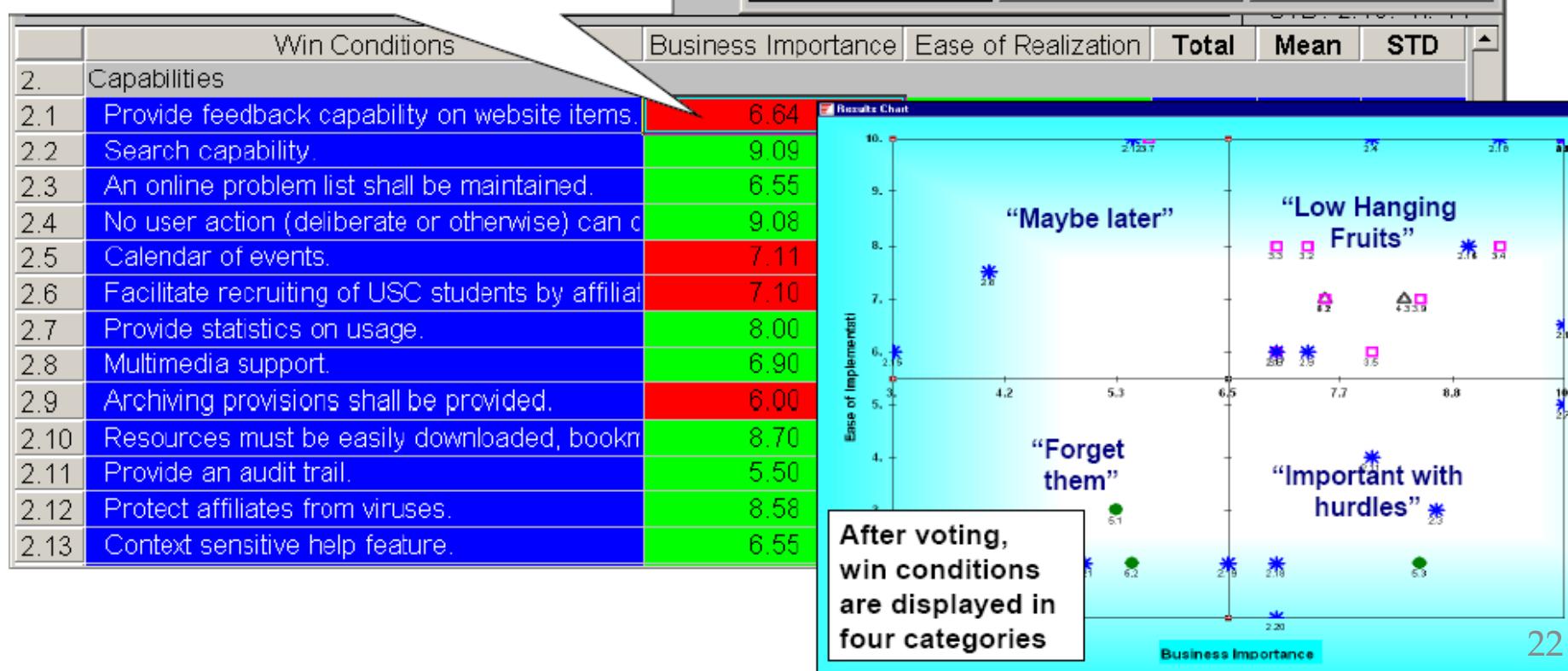


Prioritize Win Conditions (Alternative Analysis)

Rate from 1 to 10, with 10 the highest value.

Red cells indicate lack of consensus.

Oral discussion of cell graph reveals unshared information, unnoticed assumptions, hidden issues, constraints, etc.



Consistency with goals

Problem

- Once multiple criteria have been acknowledged
 - Find solutions that satisfy all of them
 - Document the trade-offs that were made

Example

- Authentication should be *secure*, *flexible* for the user, and *low cost*

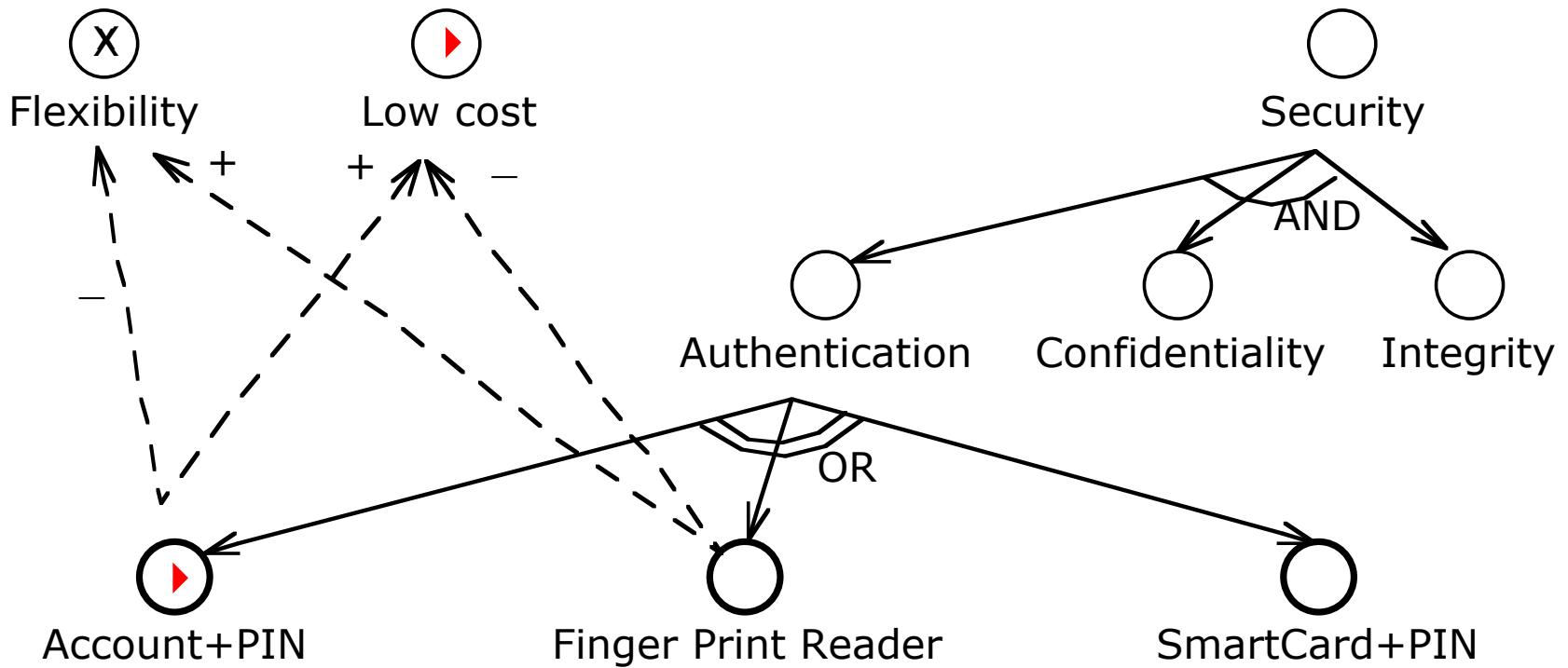


Consistency w. Goals: NFR framework



- NFR goal refinement
 - NFRs are represented as goals in a graph
 - Leaf nodes of the graph are operational requirements
 - Relationships represent “help” “hurt” relationships
 - One graph can represent many alternatives
- NFR evaluation
 - Make and break values are propagated through the graph automatically
 - Developer can evaluate different alternatives and compare them

Consistency with goals: model



Priority-based conflict resolution strategies

Disagreement between	Resolution strategy
Individual users	Product champion or product owner decides
User classes	Favored user class gets preference
Market segments	Segment with greatest impact on business success gets preference
Corporate customers	Business objectives dictate direction
Users and User managers	Product owner or product champion for the user class decides
Development and customers	Customers get preference, but in alignment with business objectives
Development and marketing	Marketing gets preference

Overview

1

Requirements Negotiation

2

Negotiation Techniques

3

Requirements Prioritization

4

Prioritization Techniques

Prioritization is about optimization



“Prioritization helps the project deliver the **maximum** business value as **quickly** as possible within the project **constraints**.”

[Wiegers, 2013]

When is prioritization needed?

- Decide on core requirements
- Select an ordered, optimal set of requirements for implementation
 - Plan releases by indicating which features are critical and which can be scheduled for future releases
- Balance the business benefit of each requirement against its cost
- Balance implications of requirements on the current software architecture and against future evolution of the product
- Keep on schedule

What to prioritize

Use Case Prioritization

- Some alternative flows could have a higher priority than others

Feature Prioritization

- Related to scope and release planning
- Helps distinguish core requirements from refinements
 - Initial prioritization of features
 - Prioritize the functional requirements within features

Priority levels

- **High priority**
 - A high-priority requirement must be demonstrated at the delivery of the system
 - Addressed during analysis, design, and implementation
- **Medium priority**
 - Medium priority requirements are usually demonstrated in future iterations
- **Low priority**
 - Illustrates how the system is going to be used in the future with not yet available technology
 - Low priority requirements have impact only on the analysis model

Stakeholder participation

Various stakeholders need to participate in prioritization sessions

- Customers
- Project sponsors
- Project management
- Developers
- ...

Example

- Developers may be unaware about the business value of a requirements, while the customer can be unaware about the implementation complexity

Six issues for successful prioritization

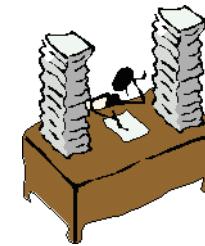
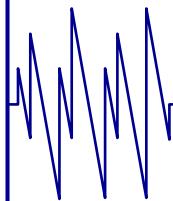
- The user and customer **needs**
- The **relative importance** of requirements to the customers
- The **timing** at which capabilities need to be delivered
- Requirements that serve as **predecessors** of other requirements and other relationships among requirements
- Identifying requirements that need to be implemented as a **group**
- The **cost** to satisfy each requirement

Different views



Customers...

- ...don't like to prioritize
- ...think, they will never get the low prioritized requirements implemented



Requirements Engineers...

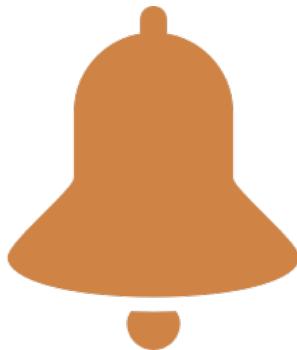
- ...have to prioritize
- ...prioritizes requirements, to achieve the most important capabilities for the business objectives of the customer

Helping stakeholders with low-prioritized requirements

- Is there some **other way** to satisfy the need that this requirement addresses?
- What would be the **consequences** of omitting or deferring this requirement?
- What **effect** would it have on the project's **business objectives** if this requirement weren't implemented for several months?
- **Why** might a customer be unhappy if this requirement were deferred to a later release?
- Is having this feature **worth** delaying release of **all of the other** features with this same priority?

Aspects of prioritization

Importance



Cost



Risk



Time



Financial benefit

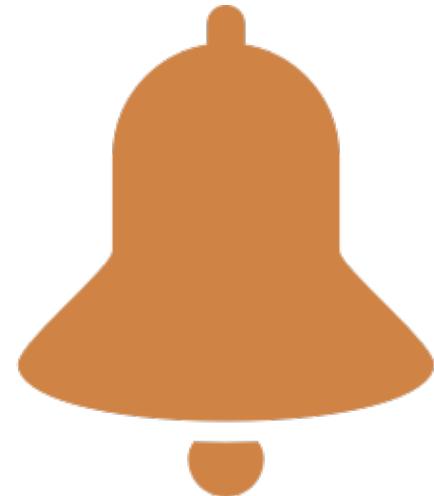


Penalty



Importance

- Indicates whether a requirement is important for the system
- How important is the requirement to the stakeholders?
 - Depends on stakeholder's perspective
- Examples
 - Urgency of implementation
 - Strategic importance



Implementation cost

- Usually estimated by the development team
- Some factors influencing cost
 - Complexity
 - Code-reuse ability
 - Testing and documentation needed
 - Validation and verification effort



Penalty

- An implication if a requirements is not fulfilled
- Examples
 - Failing to conform to law obligations
 - Removing a feature the users like



Time

- Factors influencing time
 - Conformity to industry standards
 - Degree of parallelism possible during implementation



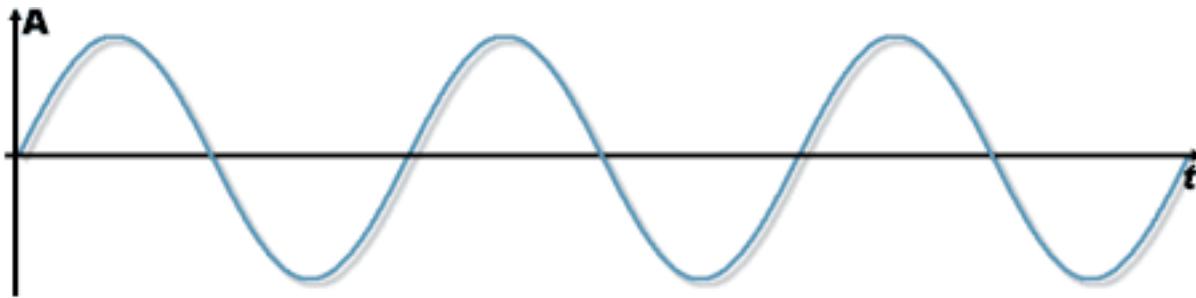
Risk

- Risk management is used to cope with internal and external risks
 - Internal risk:
technical risk, market risks, etc.
 - External risk
regulations, suppliers, etc.
- Both likelihood and impact must be considered
- Important part of release planning



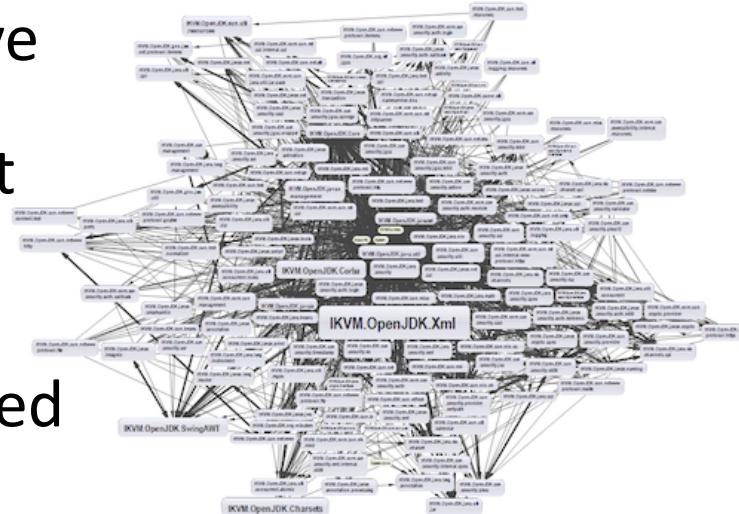
Priorities change over time

- When...
 - Stakeholder's expectations or preferences change
 - Stakeholders change
 - Project context changes
 - The system evolves
 - Domain understanding improves
 - ...



Aspects depend on each other

- Change in one aspect could result in another aspect
 - A high priority requirement can be changed to a low priority one, if it is expensive
- It is important to consider different aspects when deciding if a requirement should be implemented
 - Tradeoff Analysis



Overview

1

Requirements Negotiation

2

Negotiation Techniques

3

Requirements Prioritization

4

Prioritization Approaches

Overview of prioritization approaches

- In or Out
- Three level Scale approach
- 100\$ prioritization approach
- 100 points prioritization
- Cost-value approach
- Cost-value risk
- Numerical assignment
- Binary search tree
- Analytical Hierarchy Process (AHP)
- ...

In or Out

The simplest prioritization method



Approach

- Work down a list of requirements and make a binary decision: Is the requirement in, or is it out?
- Result: two sets
 - The In-Set: selected requirements for the current release
 - The Out-Set: requirements that will be considered for the next releases

Three level scale

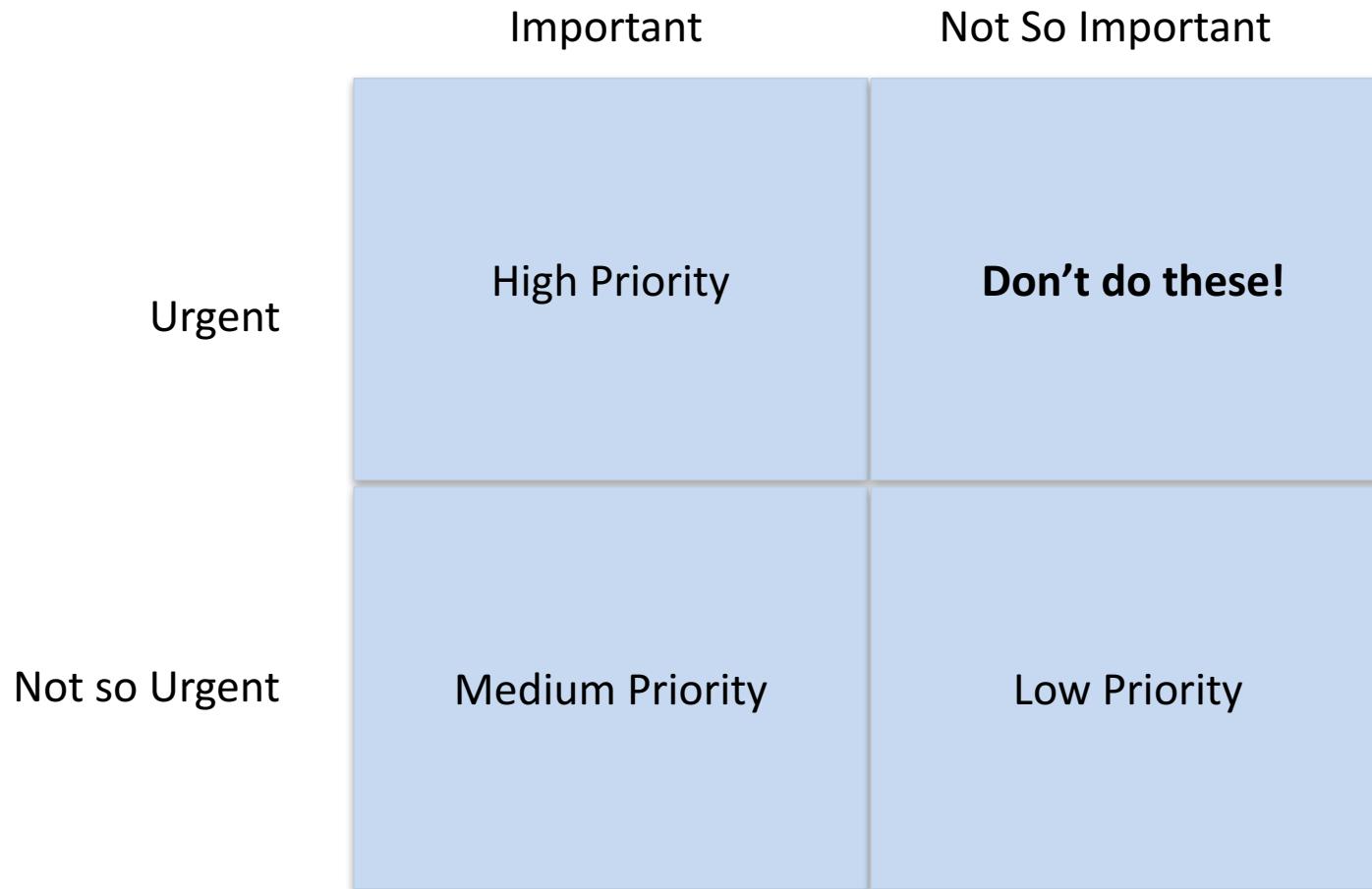


Groups the requirements in three categories:
high, medium, and low priority

Approach

- Stakeholders need to agree on the meaning of each scale
- Using two priority dimensions (e.g. importance and urgency) yields to four possible combinations
- Each requirement can be:
 - Either important to achieving business objectives or not
 - Either urgent or not

Importance vs urgency



Three level scale approach: “don’t do these” quadrant

Captures urgent requirements that are not so important.

- Don’t waste your time working on these!

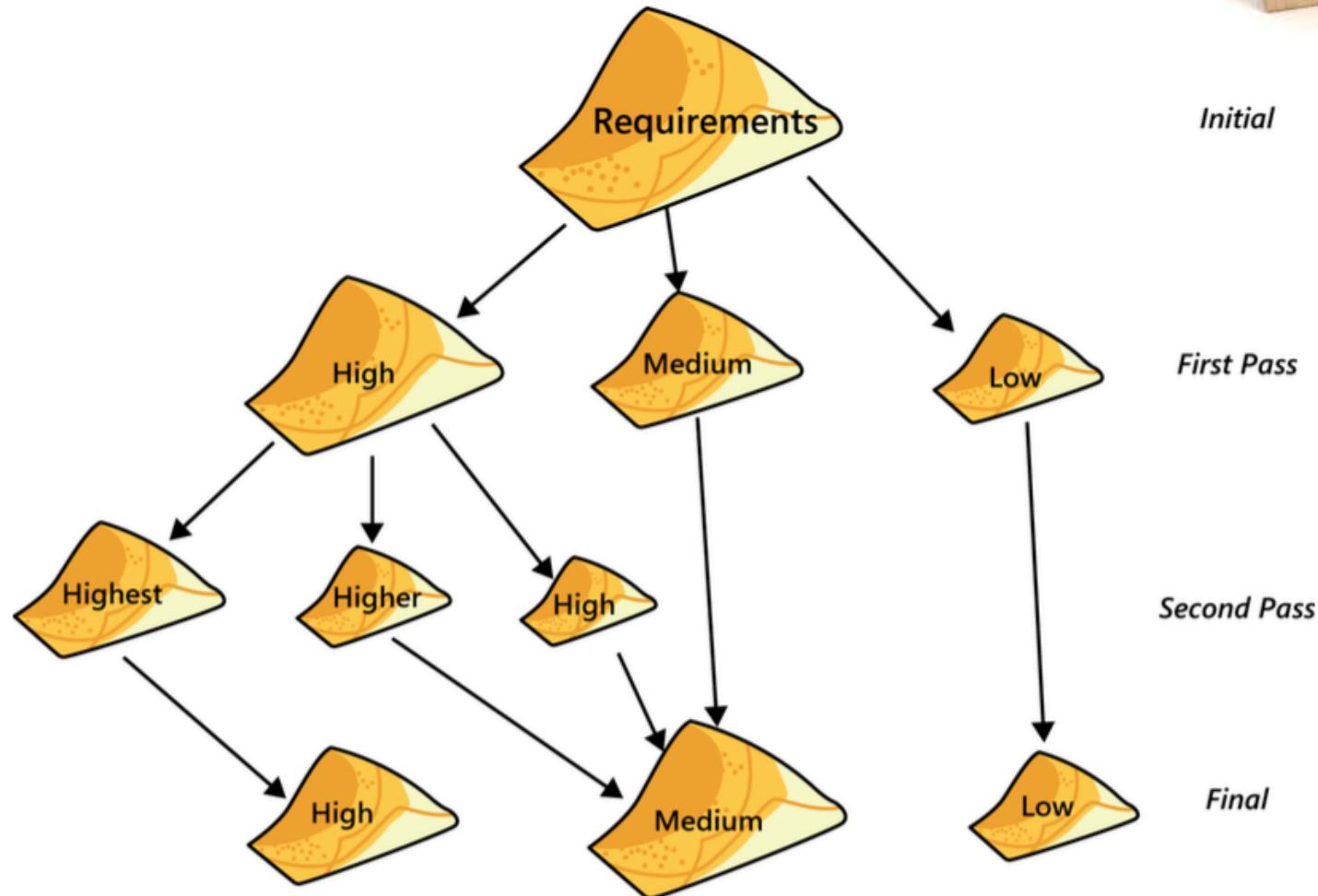
Rationale

- They are probably urgent for political reasons, but aren’t important for achieving the business objectives.
- They don’t add sufficient value to the product.

What to do with them?

- Either set them to low priority or scrub them entirely.

Iterative prioritization



Iterative prioritization (1)



In large projects iterative prioritization is more appropriate

- Keep the focus on a **manageable** set of top-priority requirements

First Pass

- Rate requirements as high, medium, or low priority
- If the number of high-prioritized is still too high conduct a second pass

Second Pass

- Rate the high-prioritized requirements as high, higher, and highest

Final

- Higher -> high-prioritized requirements
- High and higher -> medium-prioritized requirements

100\$ prioritization approach



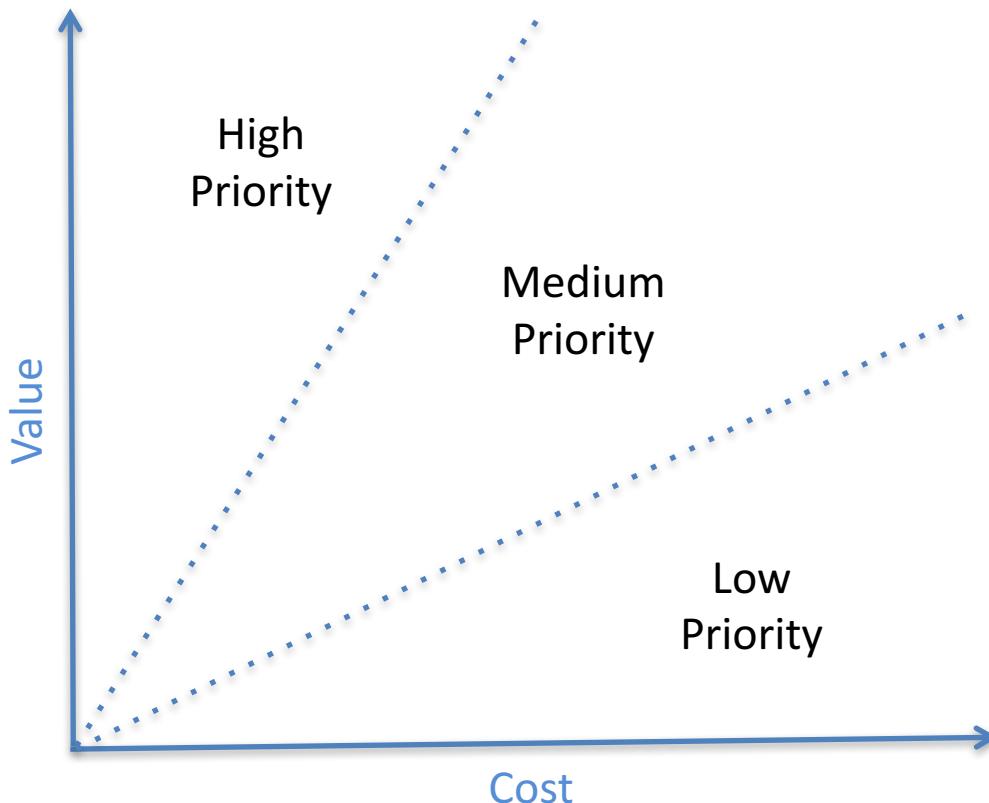
An approach to make prioritization more tangible by using the resource **money**.

- Use limited resources to achieve the maximum benefit from the project investment (100\$)

Approach

- The prioritization team gets 100 imaginary dollars to work with
- The team allocates the dollars to a set of requirements they would like to have implemented
- Higher prioritized requirements get more dollars assigned
- When the team runs out of money, no further requirements can be selected for implementation

A cost-value approach



Estimating cost and value



- Two approaches
 - Use absolute scale (e.g. currency)
 - Requires domain knowledge
 - Use relative values (e.g. less, more, a little, etc.)
 - Easier to elicit

A cost-value approach: challenges

- Express differences between requirements
 - The easy way
 - Is x more important than y?
 - The hard way
 - How much is requirements x more important than y?
- Not all requirements are comparable
 - Requirements depend on each other
- Disagreements between stakeholders



A cost-value-risk approach



- Relate customer values to proposed product features
- Apply this prioritization scheme to requirements, that aren't obviously top priority
- Don't choose requirements that implement core business functions, key product differentiators, or items required for regulatory compliance

The stakeholders



- The project manager and the business analyst:
 - Lead the process, arbitrates conflicts, and adjust prioritization data
- Customer representatives (e.g. product champions, a product manager or a product owner):
 - Supply the benefit and penalty rating
- Development representatives:
 - Provide the cost and risk ratings

An example



Relative weights		2	1		1		0.5			
Feature		Relative benefit	Relative penalty	Total value	Value %	Relative cost	Cost %	Relative risk	Risk %	Priority
1.	Print a material safety data sheet.	2	4	8	5.2	1	2.7	1	3.0	1.22
2.	Query status of a vendor order.	5	3	13	8.4	2	5.4	1	3.0	1.21
3.	Generate a chemical stockroom inventory report.	9	7	25	16.1	5	13.5	3	9.1	0.89
4.	See history of a specific chemical container.	5	5	15	9.7	3	8.1	2	6.1	0.87
5.	Search vendor catalogs for a specific chemical.	9	8	26	16.8	3	8.1	8	24.2	0.83
6.	Maintain a list of hazardous chemicals.	3	9	15	9.7	3	8.1	4	12.1	0.68
7.	Change a pending chemical request.	4	3	11	7.1	3	8.1	2	6.1	0.64
8.	Generate a laboratory inventory report.	6	2	14	9.0	4	10.8	3	9.1	0.59
9.	Check training database for hazardous chemical training record.	3	4	10	6.5	4	10.8	2	6.1	0.47
10.	Import chemical structures from structure drawing tools.	7	4	18	11.6	9	24.3	7	21.2	0.33
Totals		53	49	155	100.0	37	100.0	33	100.0	

Project manager & Business Analyst:

- List of features to prioritize
- Relative weights

Customer representative:

- Relative benefit and penalty for the Customer

Developers:

- Relative cost and risk

Priority:

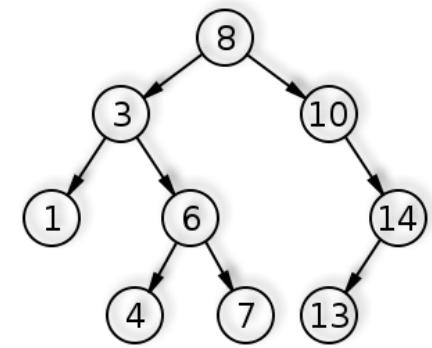
Value% / (cost% + risk%)

Challenges & recommendations



- Approach is limited by the team's ability to estimate the benefit, penalty, cost, and risk
- Use calculated priorities only as guidelines & review them
- How to tune the weighting factors?
 - Use a set of implemented requirements from a previous project
 - Adjust the weighting factors until the calculated priority sequence correlates with the real priorities of the requirements

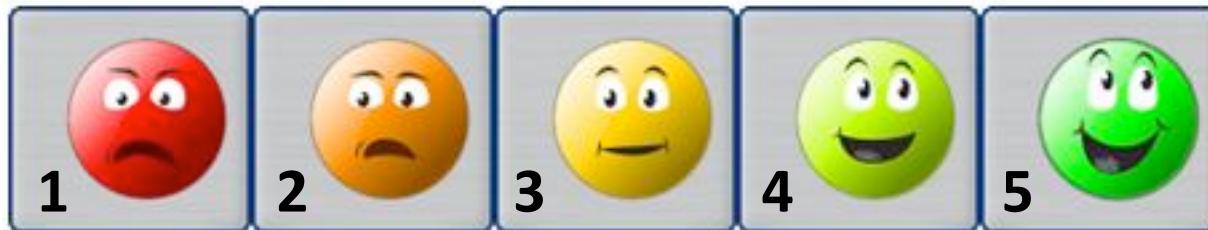
Binary search tree



- Search algorithm adapted for requirements prioritization
- Approach
 - Put all requirements in one bucket
 - Take one requirement as root
 - Take other requirements and add them to the tree as leaf, starting from root
 - Traverse left, if it's less important than the current node
 - Or right, if it's more important than the current node

Numerical assignment

- Requirements are divided into three groups
 - Mandatory, desirable, and unessential
- Stakeholders assign to each requirements a number between 1 and 5
 - 1 = does not matter, ..., 5 = mandatory
- The final ranking is the average of all stakeholders' ranking for each requirement



Summary

1

Requirements negotiation aims at resolving conflicts among stakeholders and **externalizing issues and solution alternatives**

2

Negotiation approaches can be based on **rationale thinking**, group decisions or importance/priority of stakeholders

3

Prioritization is a specific form of negotiation **needed for release planning** and for dealing with resource limitation

4

Important prioritization approaches include **3 level scales**, \$100, and cost-value-risk prioritization approach