



H4 Comments



Adaptation to HS-07

- Size of system/organization
 - Fewer people necessary, person <> role
 - Shorter presentations
 - Merge phase 1 and phase 2
 - Possibly external consultants (because of lack of skills)
 - Probably internal evaluators (because of cost)
- Type of system
 - In-house vs contract vs product
 - Need to represent customer somehow
 - Questionnaires
 - Market survey
 - Constructing “personas”

Observations

- Architect is forced to make architecture precise
- Stakeholders are involved
- Thorough and well-validated process
 - Can find new angles to problems
- Resource demanding
 - Requires substantial work
 - Requires stakeholder presence
 - Not suitable for small projects



Observations

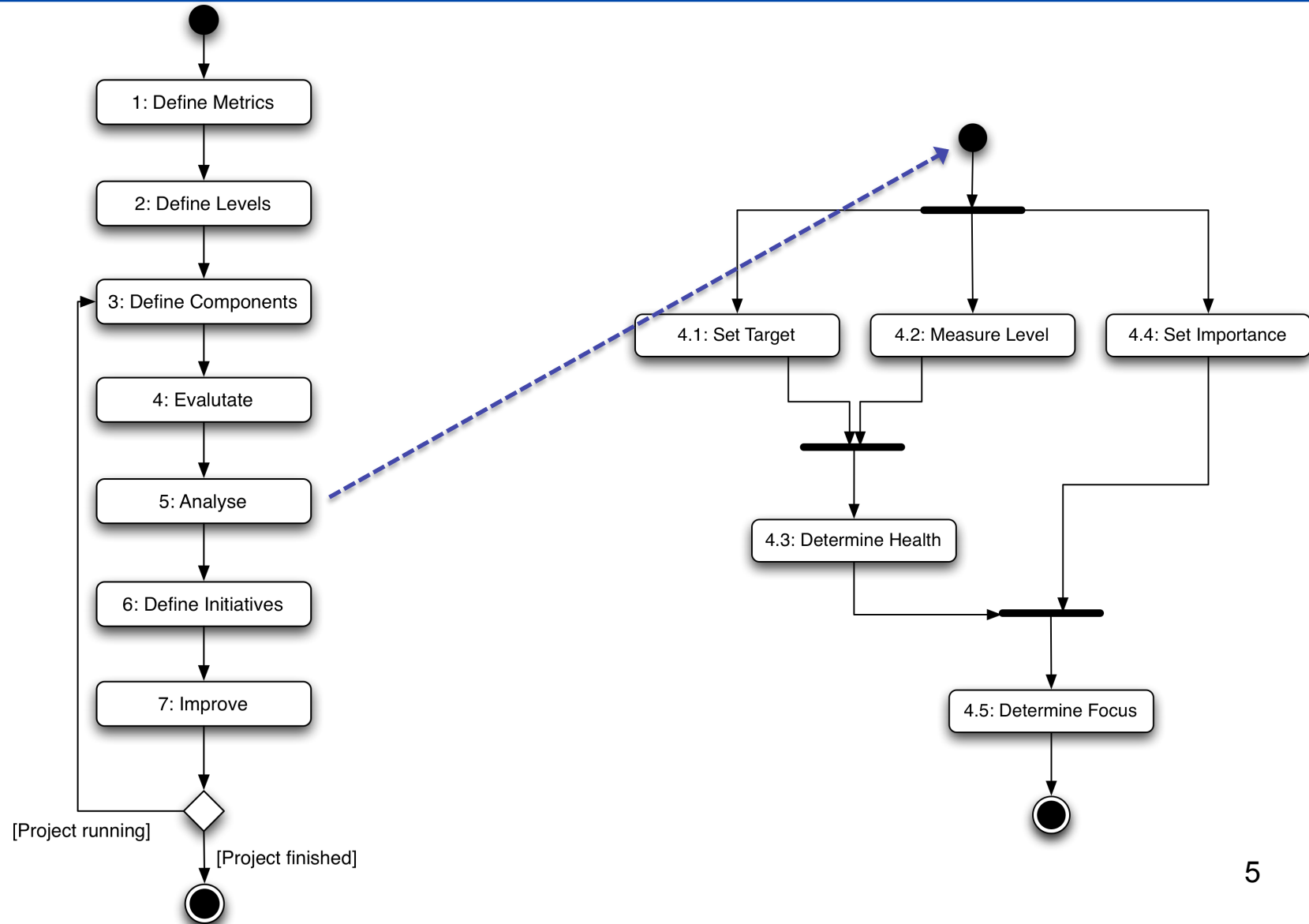
- May be inaccurate
 - Description of utility/response graph => uncertainty
 - Cost estimates may be inaccurate
- Forces focus on benefits and cost of design decisions
- Encourages considering multiple possible decisions
- Heavyweight
- Not necessarily for developers
 - Business-oriented
- Forward-looking



Observations

- Good idea to evaluate repeatedly
 - Fits with iterative development
 - But requires organizational backing
- Validity depends on validity of metrics
- Lightweight
 - Easy to use (a.o. due to orientation towards components)
 - Manageable workload, suitable for repetition
 - Would prefer in daily practice (in comparison to ATAM and CBAM)
 - May suit different project types
 - In particular if focus is on improving quality
- Heavyweight
 - Requires projects of a certain size
- Gives overview for decision makers
 - Can track changes
 - Forward-looking
- Mostly suitable for completed system under modification
 - Requires well-composed system (including stability of interfaces)
 - Cannot apriori handle changes in C&C structure
- Generelle kvalitetsattributter giver kunstig opdeling, bedre med scenarier
- Complements CBAM and ATAM
 - All three methods have their application
 - Overlaps with CBAM in some activities
- Need to become more mature (and validated)

aSQA Revisited





Architectural Prototyping

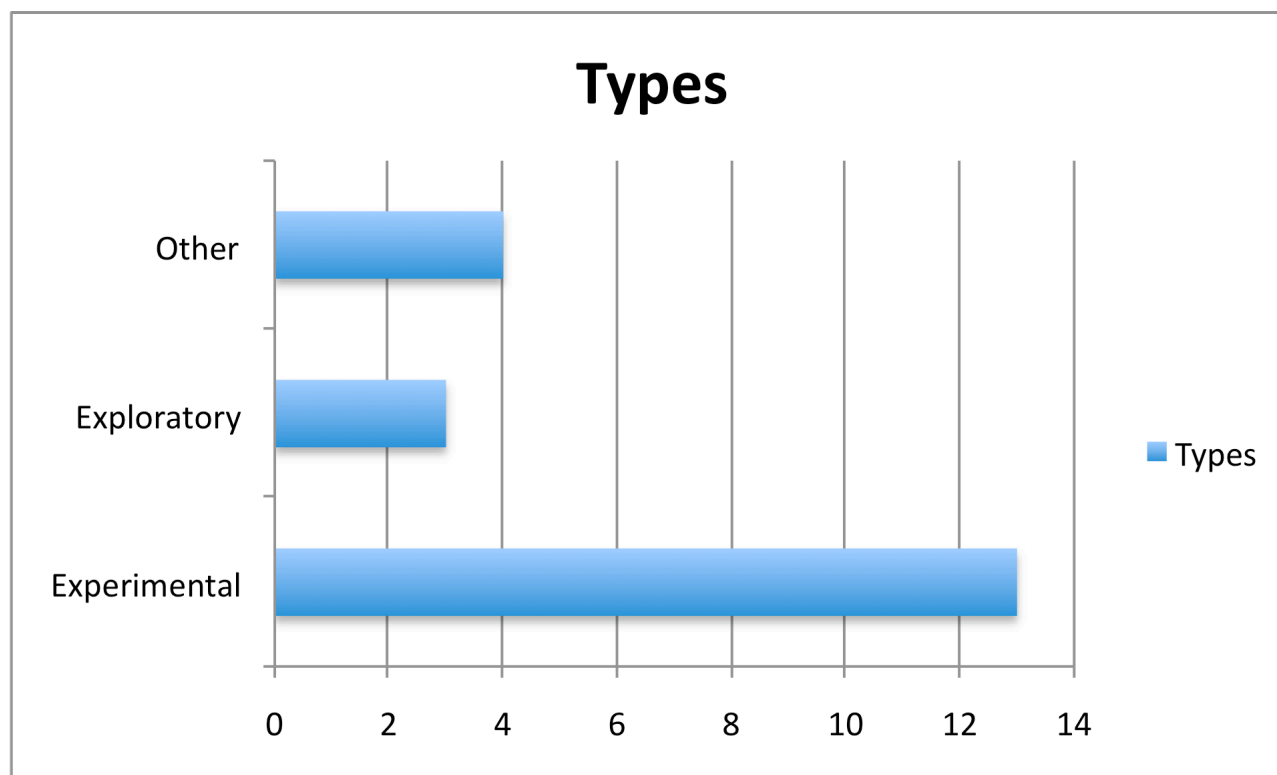
Architectural Prototypes - aSAiP 2008

	Alpha.1	Alpha.2	Bravo.1	Bravo.2	Bravo.3	Bravo.4	Charlie.1	Charlie.2	Delta.1	Delta.2	Delta.3	Foxtrot.1	Foxtrot.2	Foxtrot.3	Golf.1	Golf.2	Hotel.1	Hotel.2	Lima.1	Lima.2
Description	XSLT-overbyg	"Suggest" per	Client-server	Web service in	Message queue	OO persistence	Internet regist	Dynamic, low	Web traffic an	VBA data proc	Multichannel s	Factoring syst	PLC event dat	MVC calculato	Read-only use	Customer-orie	WCF cattle DB	Polyhierarchic	Product config	Online replicat
Classification	Experimental	Experimental	Experimental	Exploratory	Experimental	Experimental	Experimental	Exploratory	Experimental	Experimental	Experimental	Experimental	Experimental	Experimental	Experimental	Exploratory	Experimental	Exploratory +	Exploratory +	Experimental
Characteristics																				
Learning	2	4	4	5	5	4	4	4	5	5	4	3	4	4	5	2	5	4	4	4
Quality issues	5	5	5	5	5	5	5	5	5	5	5	5	5	4	3	4	5	5	4	5
Performance	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x	x
Buildability	x					x									x			x		
Modifiability				x			x		x	x					x		x		x	
Availability								x								x				x
Usability												x				x	x			
Security																	x			
Conceptual integrity																		x		
Testability																				x
No functionality	4	3	5	1	5	4	4	4	2	2	2	5	2	3	2	1	2	2	4	4
Address risks	4	5	5	4	5	4	4	2	4	4	4	5	5	4	2	3	5	5	4	5
Transfer knowledge	3	3	2	5	5	4	2	2	1	3	4	4	4	4	4	4	4	4	5	4

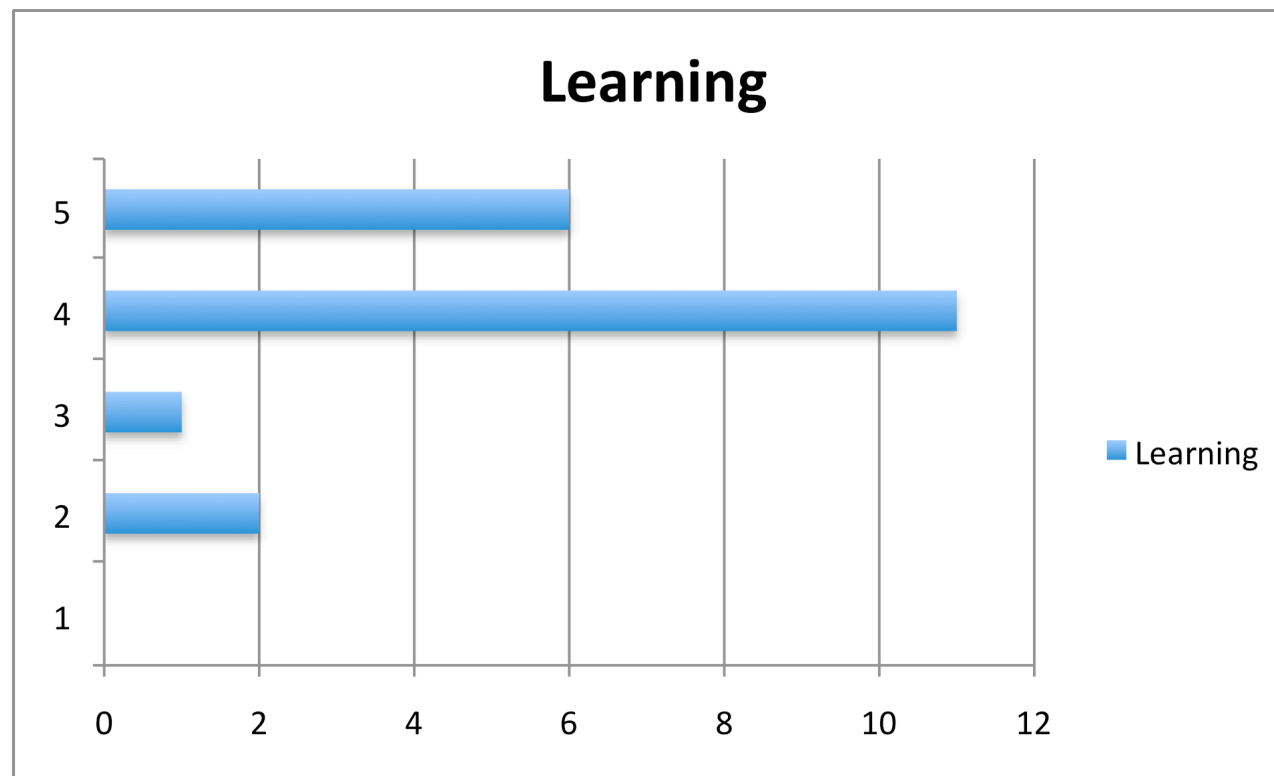
	Alpha.1	Alpha.2
Description	XSLT-overbyg	"Suggest" per
Classification	Experimental	Experimental
Characteristics		
Learning	2	4
Quality issues	5	5
Performance	x	x
Buildability	x	
Modifiability		
Availability		
Usability		
Security		
Conceptual integrity		
Testability		
No functionality	4	3
Address risks	4	5
Transfer knowledge	3	3

Likert scale

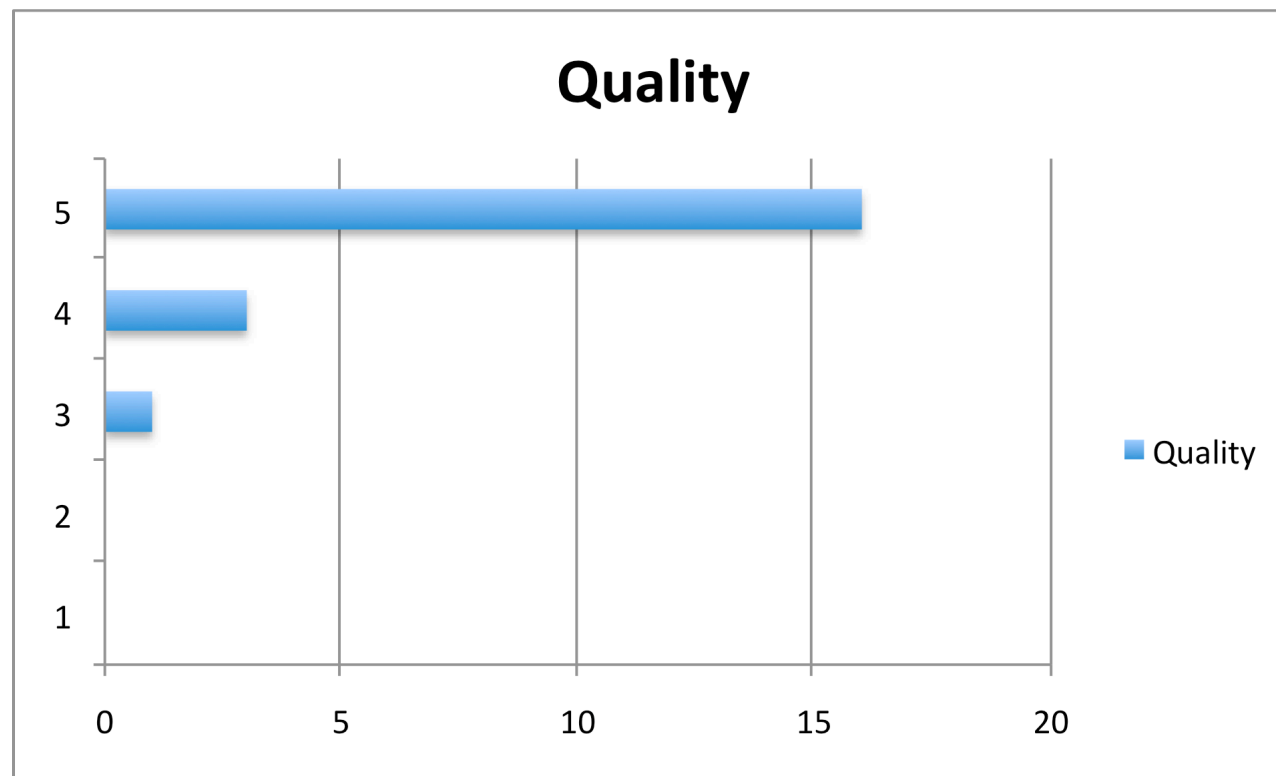
– 1 = highly disagree, 5 = highly agree

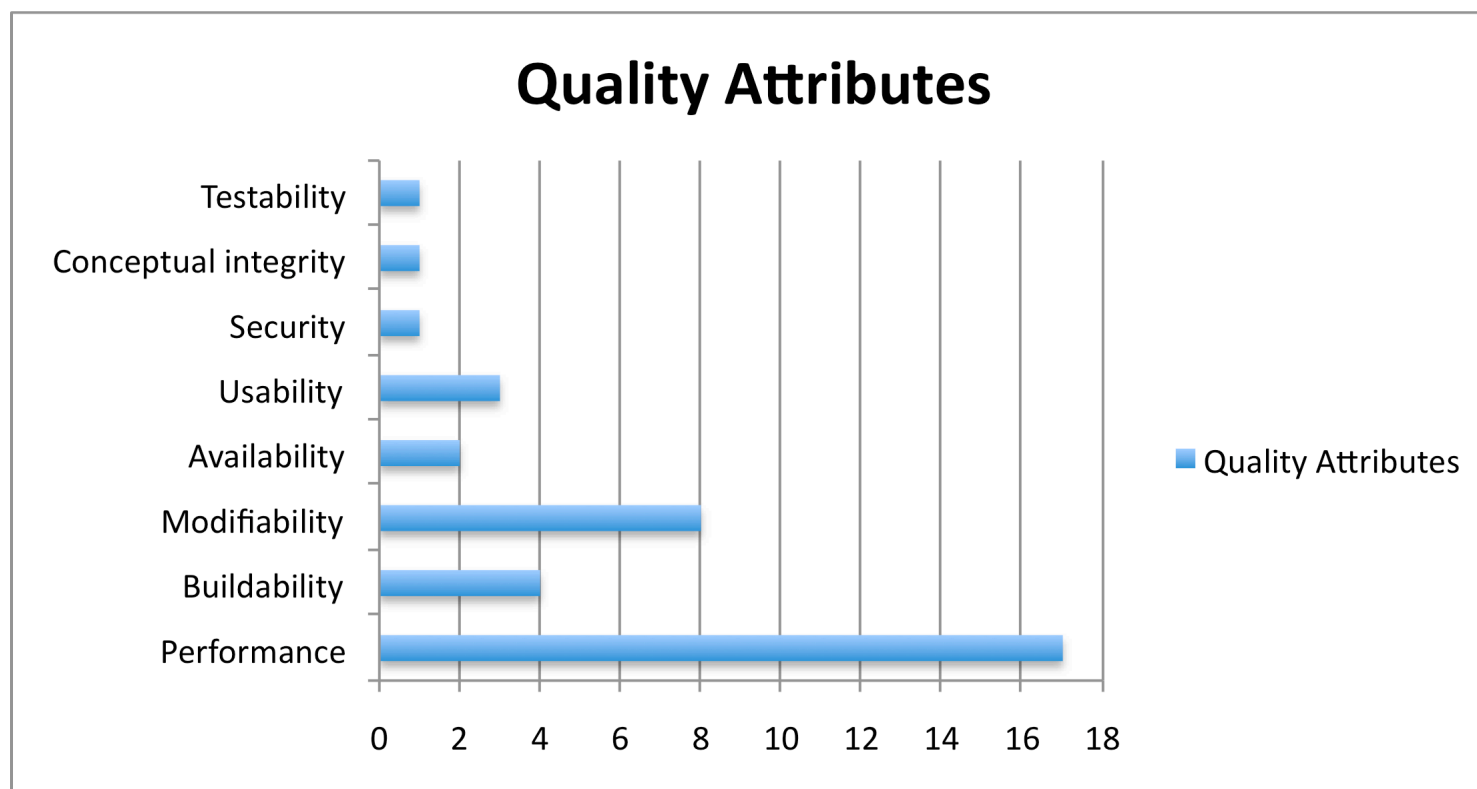


”Architectural prototypes are constructed for exploration and learning of the architectural design space”

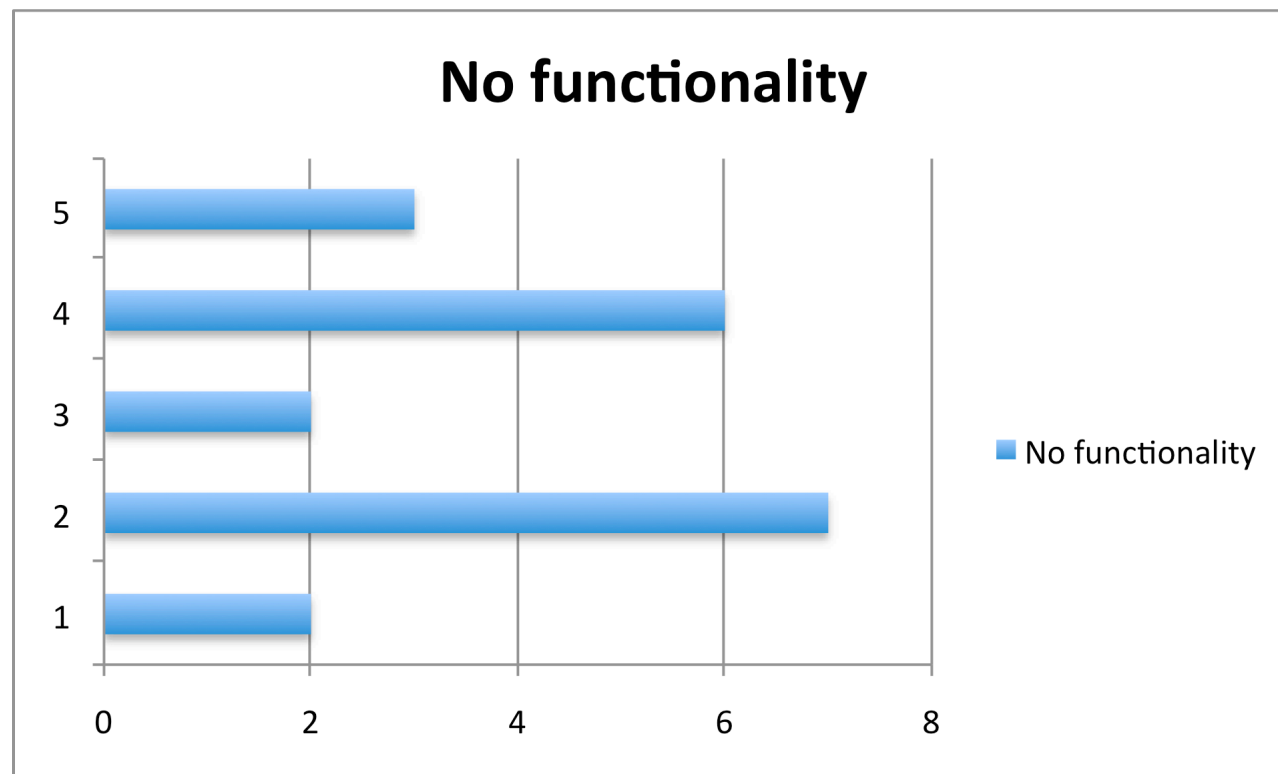


”Architectural prototyping addresses issues regarding architectural quality attributes in the target system”

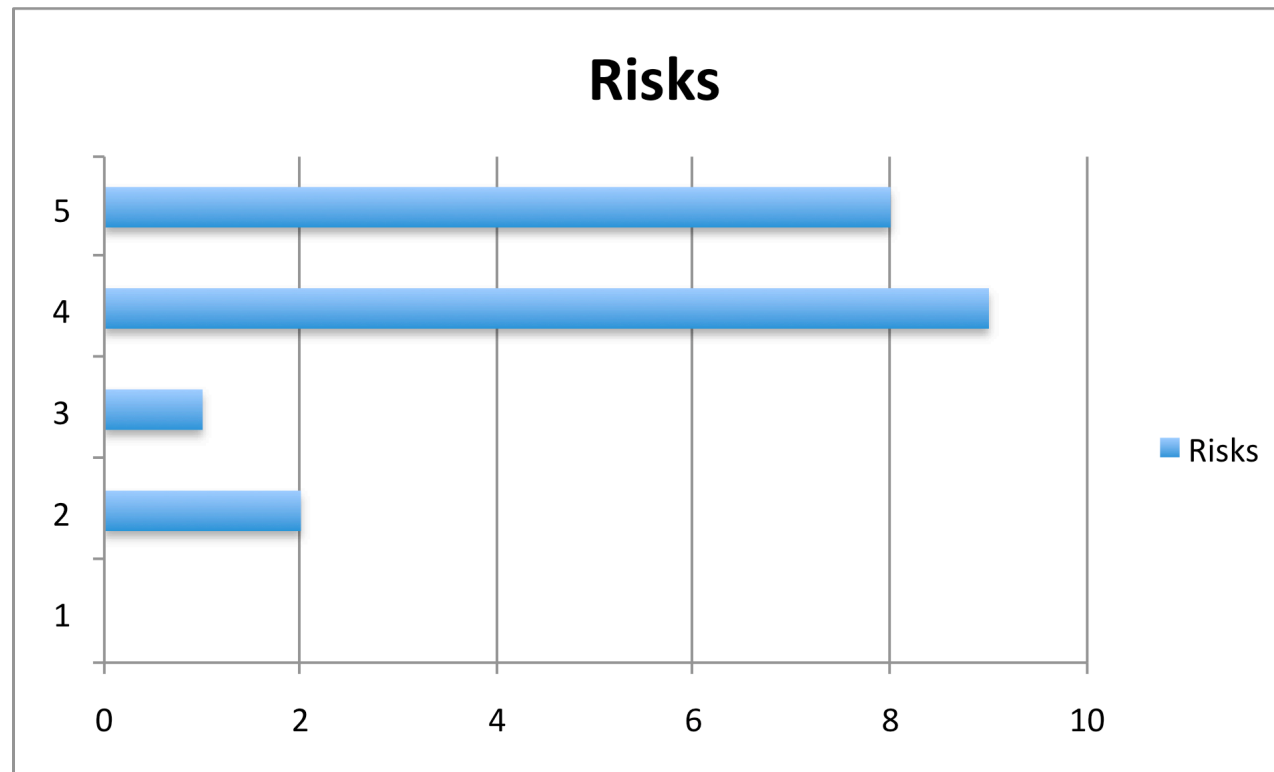




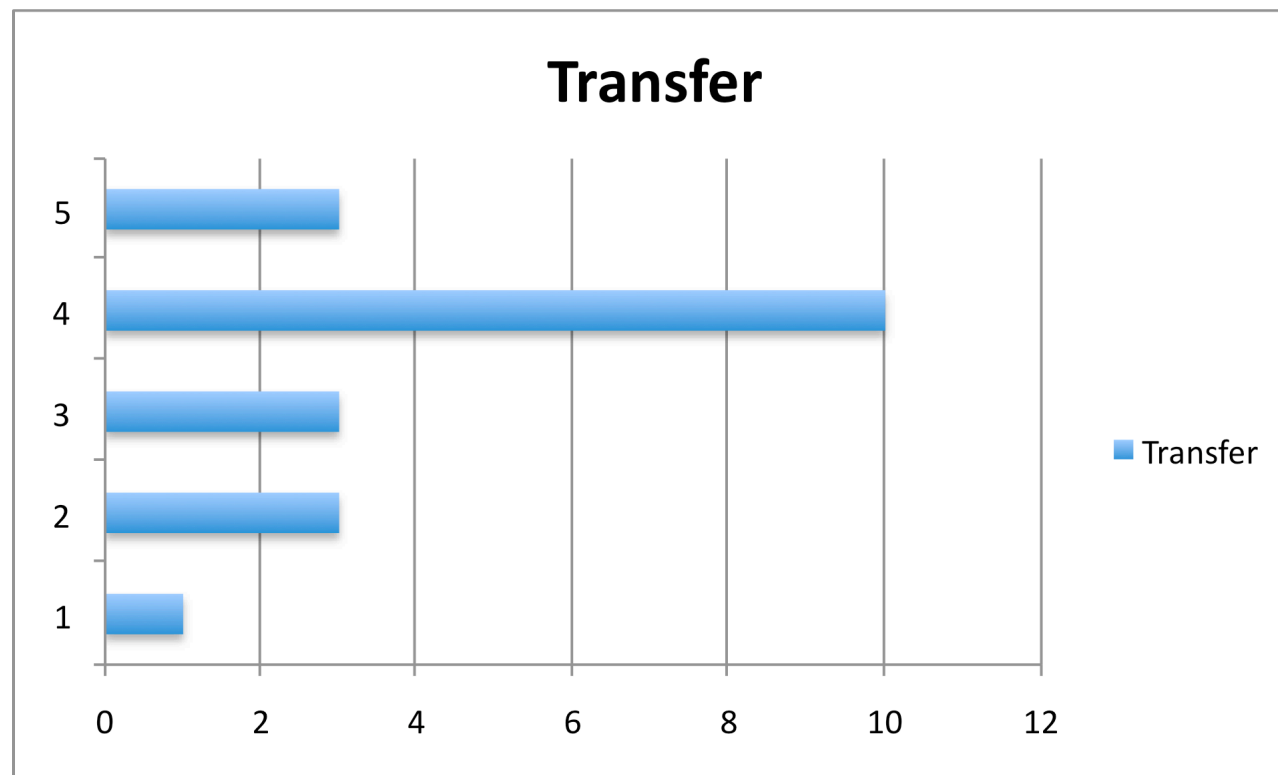
” Architectural prototypes do not provide functionality per se”



” Architectural prototypes typically address architectural risks”



” Architectural prototypes address the problem of knowledge transfer and architectural conformance”



Architectural Prototypes

Advantages

- Good help in architectural decisions (2)
- Little effort, early verification (4)
 - Save money
- Objectivity (3)
- Classification of PTs leads to a common vocabulary
- Does not overlook problems (4)
 - "executables do not forget to think about a problem"
 - Discover new issues
- May express architectural intent (2)
 - in a way that developers understand
 - Work as executable specification
- Tangible for stakeholders (3)
 - Formalized => easier management buyin
- Gives real opportunities for learning (5)
 - Gain knowledge
 - Learn new technologies concepts
 - Playground/learning vehicle
 - Try alternative solutions
- Can experiment with goals as well as solutions

Architectural Prototypes

Disadvantages

- Focus on "easy" quality attributes – can hide important quality attributes (3)
 - Hard to measure, e.g., modifiability
- May lead to skeleton systems even if not intended (4)
 - Hard to throw away
- Customer may feel system is almost complete (3)
- May yield misleading results (3)
 - In particular for performance
 - Hard to create realistic environment
- Cannot replace traditional analysis
- Prototypes may not be maintained as architecture evolves
- May be costly to produce and maintain (4)
 - Collecting and processing measurement data may require substantial workload
 - May be costly when quality tightly coupled to functionality
 - Documentation needed