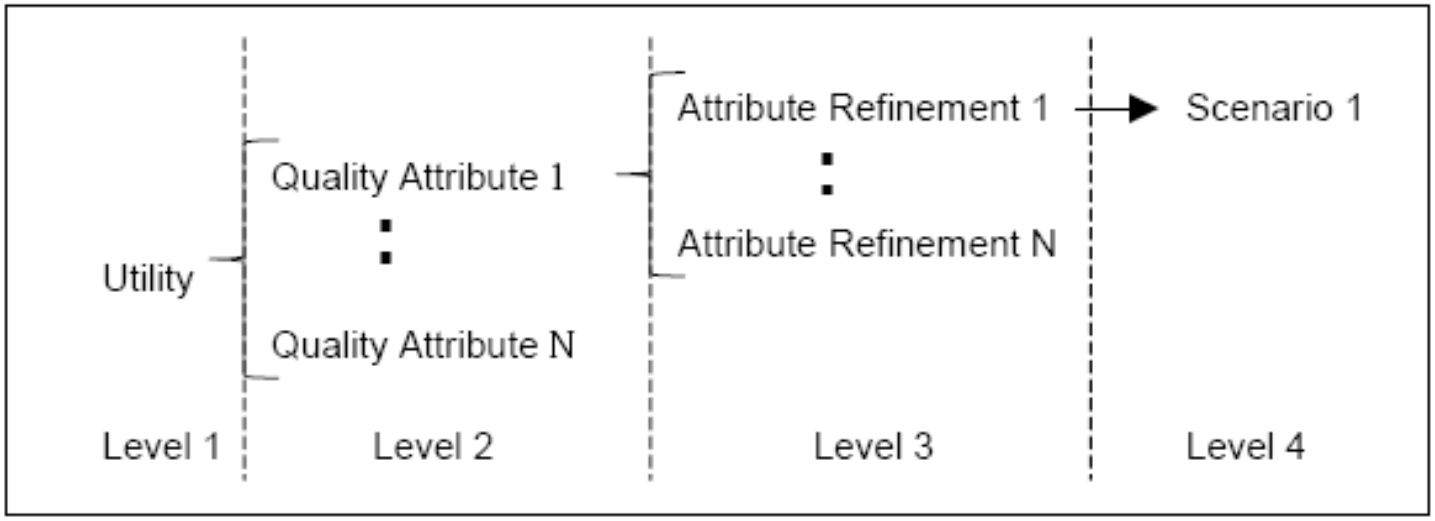
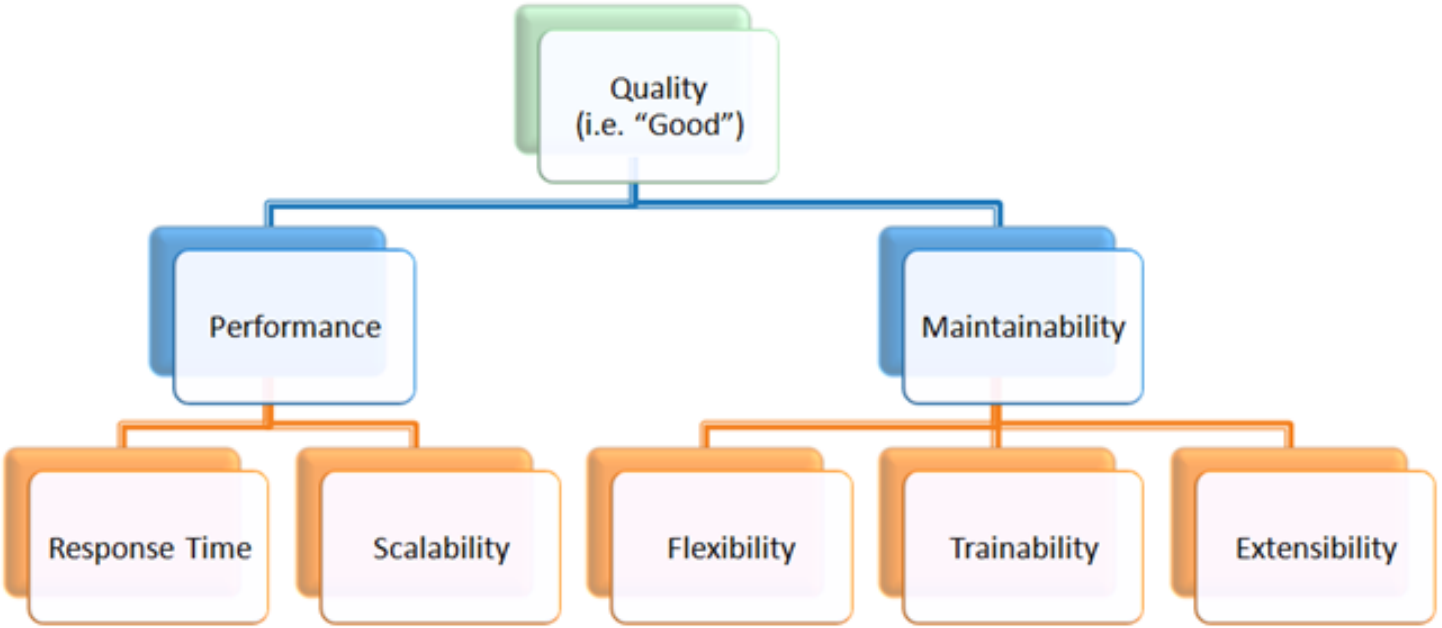
Quality Attribute Utility Tree

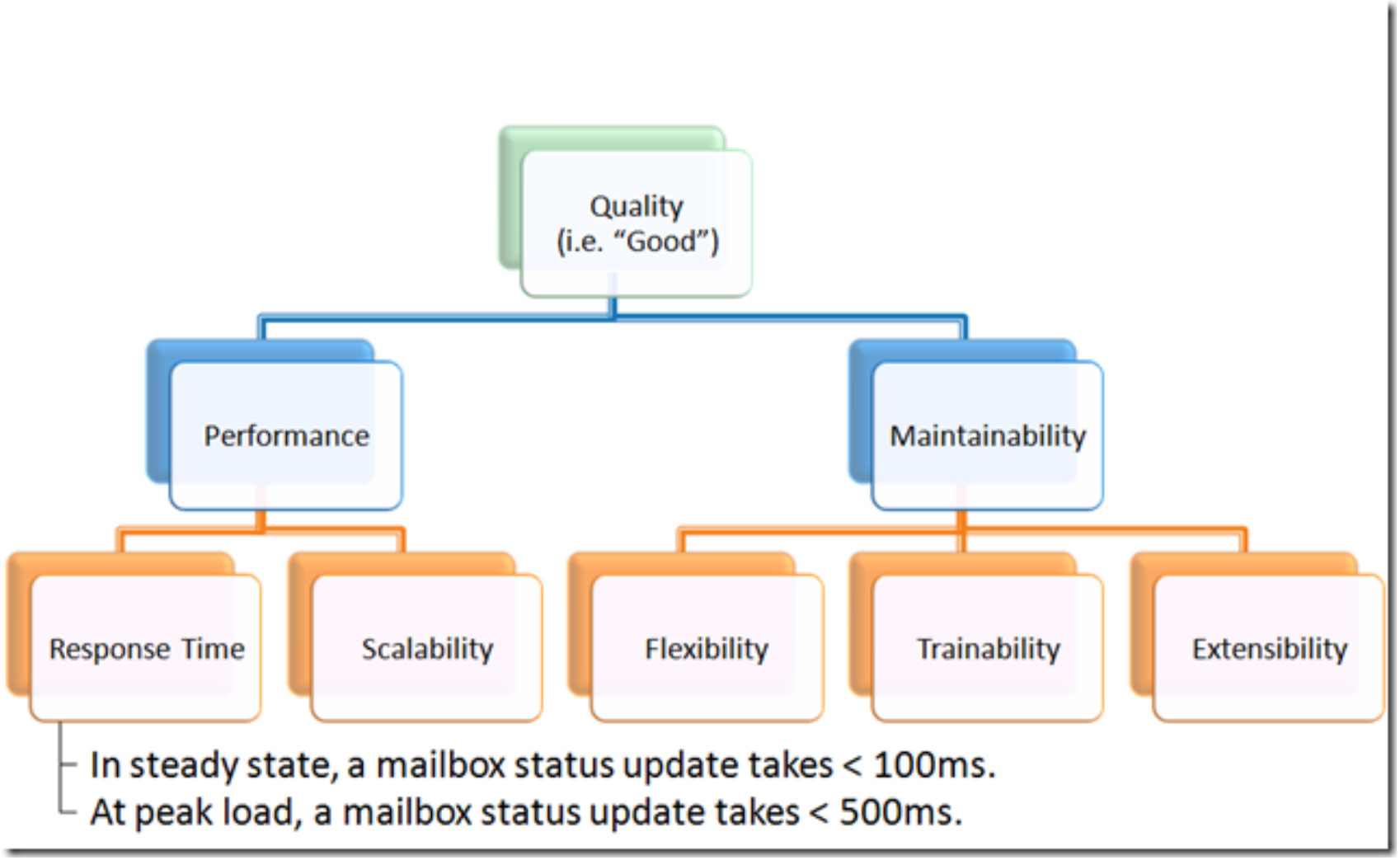


Utility tree has four levels with the root node labeled “Utility.”

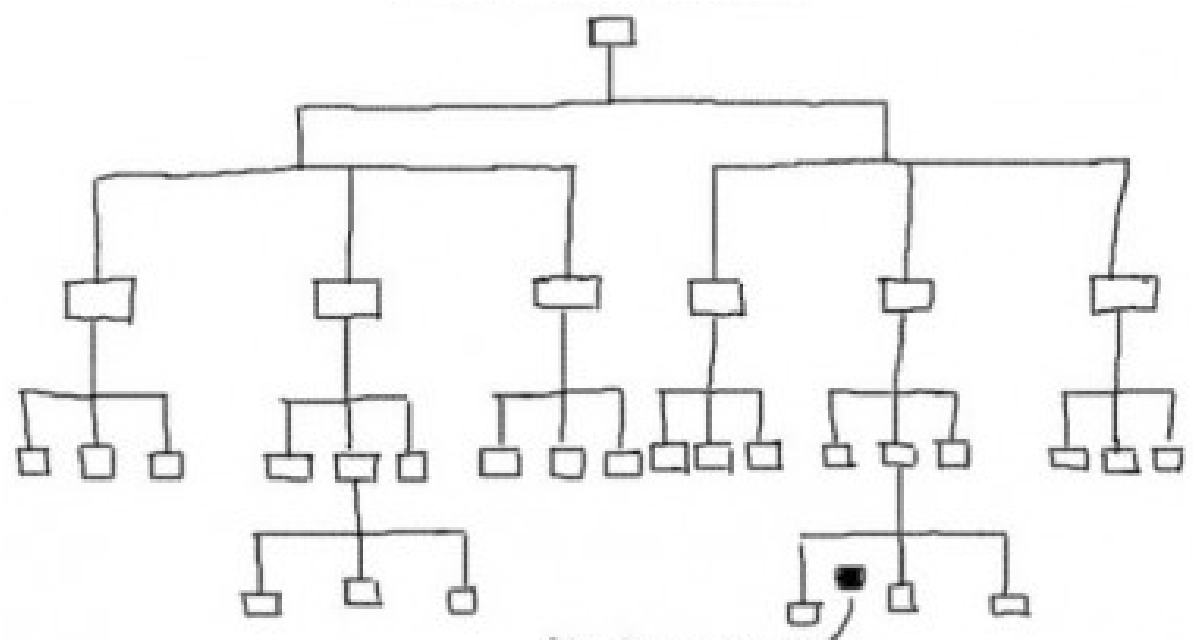


Utility tree organizes scenarios according to the quality attributes.

Quality goals form the nodes, Scenarios form the leaves



- Utility
  - Performance
    - Data latency
      - — Minimize storage latency on customer DB to 200 ms
      - — Deliver video in real time
    - Transaction throughput
      - — Maximize average throughput to authentication server
  - Modifiability
    - New Product Categories
    - Change COTS
      - — change web user interface in < 4 person weeks
  - Availability
    - Hardware Failure
      - — power output at site 1 requires traffic redirect to site 3 in < 3 s
      - — network failure is detected and recovered in < 1,5 min
  - Security
    - Data confidentiality
      - — customer database authorisation works 99,999% of time



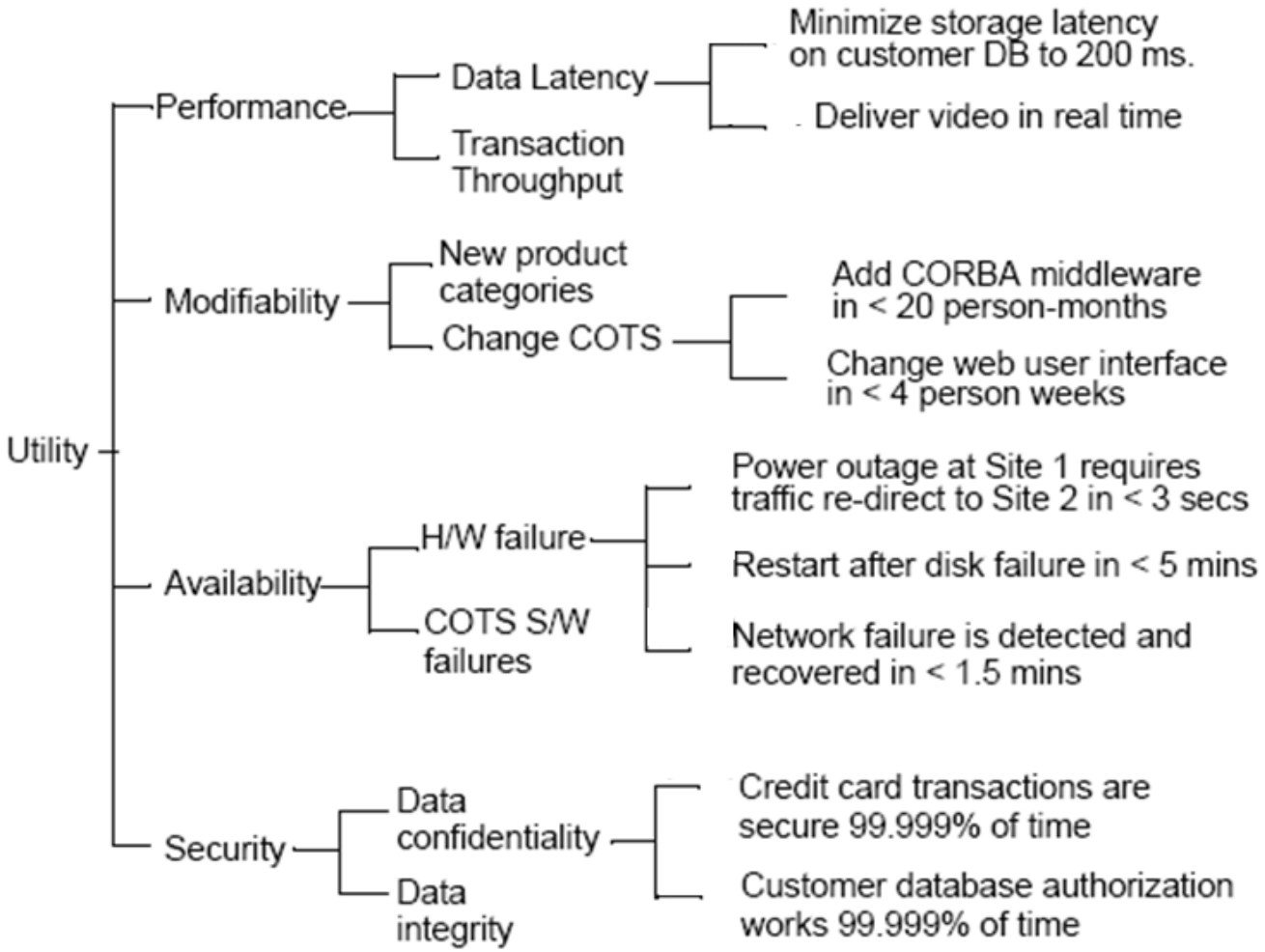
Utility tree is not an attempt at defining a rigorous taxonomy of quality attributes

There is no defined standards for creating the utility tree

Quality Model vs Utility Tree

Unlike Quality Model, utility tree has only one level refinement and the definitions of the quality attributes are not standards

The utility tree is used to give priorities to scenarios to identify sensitive points, from which a set of “test” cases for the architecture can be derived.

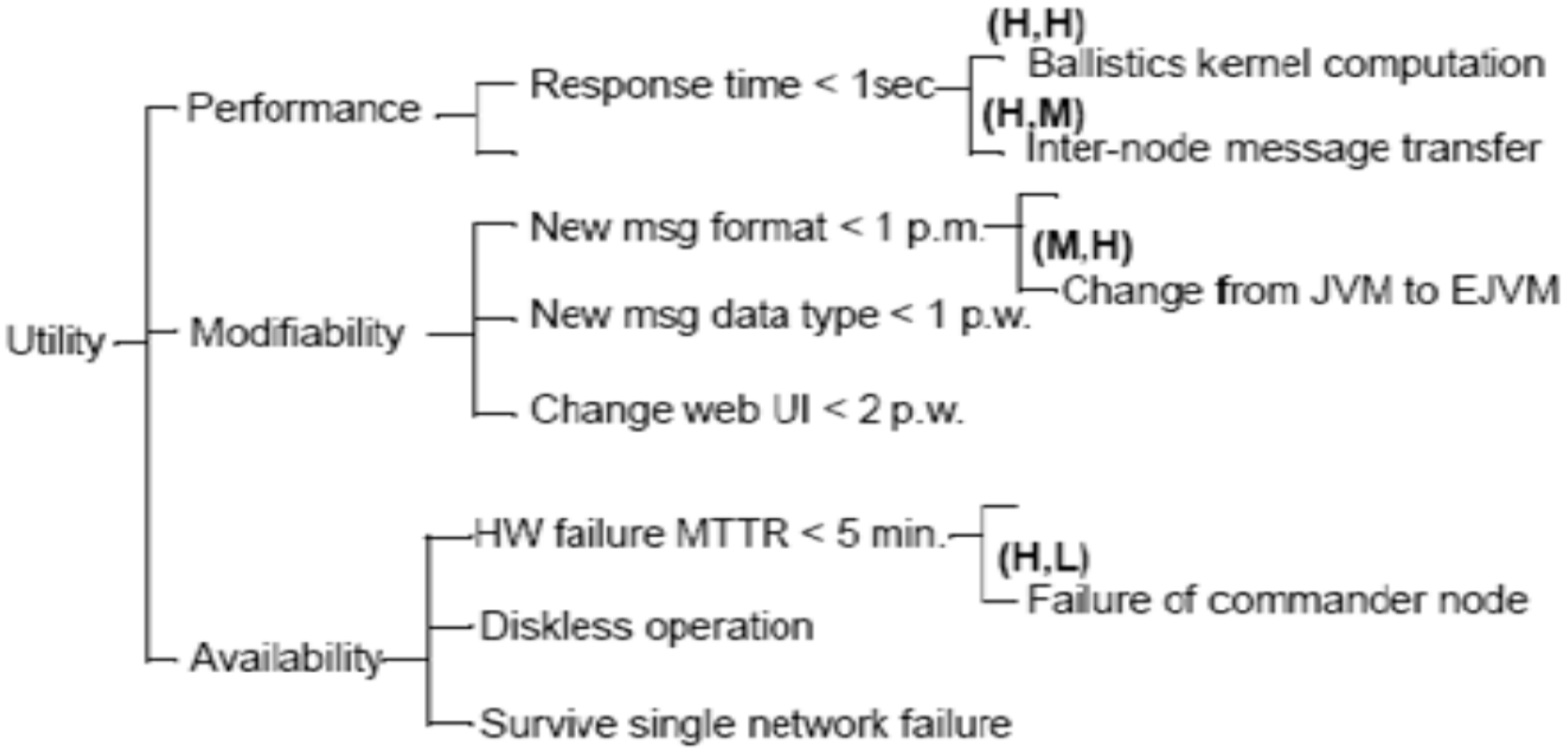


Its purpose is to elicit quality requirements in a practical, operational sense that stakeholders can understand.

Quality Attribute	Attribute Refinement	Scenarios
Performance	Response time	Ex) When a user request to launch IE in normal operation time, IE should be launched in 1 sec including a finding mobile device. When [Who:a user/N users/engineer][Doing what:request launch M application/request display M/request play music][When:in a normal operation time/for the first time/executing M application], [Who: the system/M application] should [What: respond/launch/start] in [How long: N secs/msecs]
	Power on time	Ex) When a user request to power on the system, the system should be ready in 2 secs and display the user's last things to do last time. When [Who:a user/N users/engineer][Doing what:request turning on the system/to power on the system] ([When:]), [Who: the system] should [What: be ready] in [How long: N secs/msecs]
	Seamless display	Removed for confidential
	Time for switching screens	
	Memory size	
	Network bandwidth	
	CPU occupation	
	Power usuage	
	Event dispatching	
	Application switching speed	
Modifiability/ Reusability	Easy to modify/delete functionalities	
	Easy to add new functionalities	
	Easy to change GUI	
	Portable to other O/S	

• • •

Prioritize Scenarios



Utility tree is prioritized along two dimensions
Business Importance , Technical Difficulty

Purchase2Pay: Quality Attribute Tree

QA-L1	QA- L2	BP	TP	Scenario
Performance	Latency	H	H	Opening a e-invoice for reading takes less that 3 seconds from any site that is in scope of p2p
Performance	Throughput	H	H	Opening documents at a continuous rate of 2 documents per second has average response time better than 3 sec per doc for any of the sites in scope of p2p
Availability	Overall	H	H	A site that is disconnected due to network failure is re-connected with full bandwidth in less than 2 hours
Availability	Overall	H	M	Hardware failure of one CPU in the infrastructure components (SAP, Documentum) has no effect on realization of QA
Availability	Overall	H	H	There will be no more than 4 unavailability situations per year

	Utility Tree	Brain Storming
Participants	Architects, Project Leaders	All Stake Holders
Typical Group Size	Evaluators, 2-3 Project Personnel	Evaluators, 5-10 Project related Personnel
Primary Goals	Elicit, make concrete and prioritize the driving quality requirements	To validate quality attribute goals elicited via the utility tree
Approach	General to specific, begin with quality attributes, refine until scenario emerges	Specific to general, begin with scenarios, then identify quality attributes they express