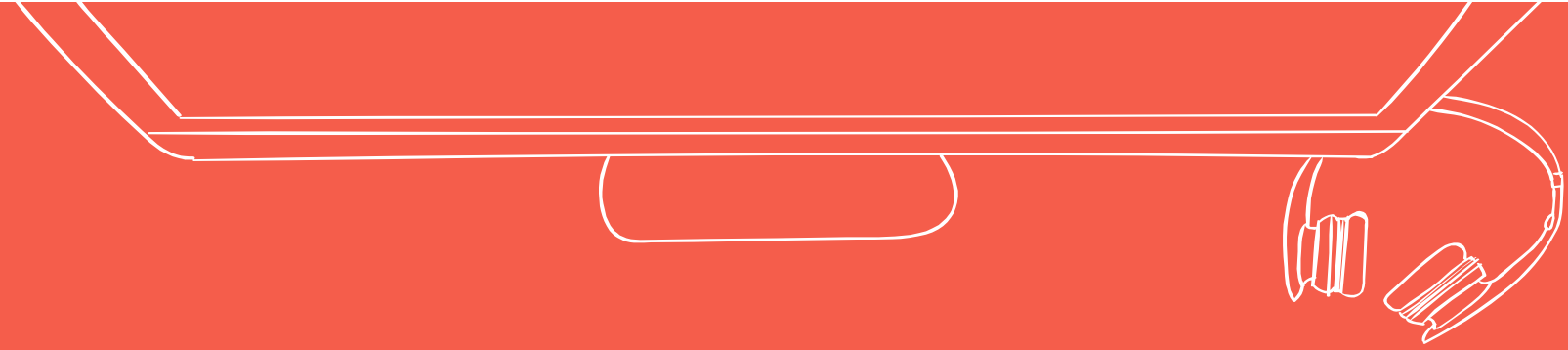


Software Architecture

Session 6: Quality Attributes

Instructor: Perla Velasco-Elizondo

Outline

1. Quality Attributes Defined
2. Walk the System Properties Web
3. Writing Quality Attributes as Scenarios
4. Writing Quality Attributes as Acceptance Criteria
5. Summary

Outline

1. Quality Attributes Defined
2. Walk the System Properties Web
3. Writing Quality Attributes as Scenarios
4. Writing Quality Attributes as Acceptance Criteria
5. Summary

User Requirements

What is the main user requirement of this system?

Quality Attributes

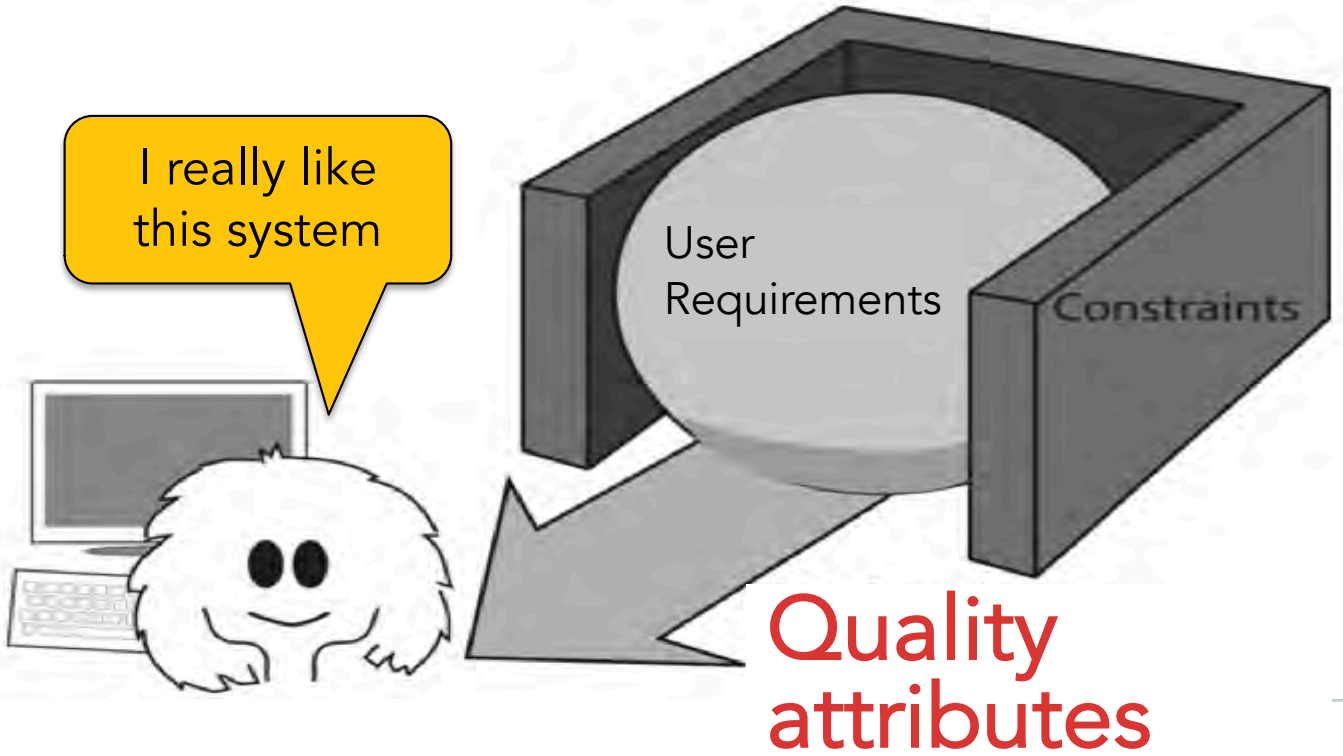
Why these systems are different?

Nice, safe, fast, expensive, etc…

Small, cheap, good for environment, funny, etc.

Quality Attributes

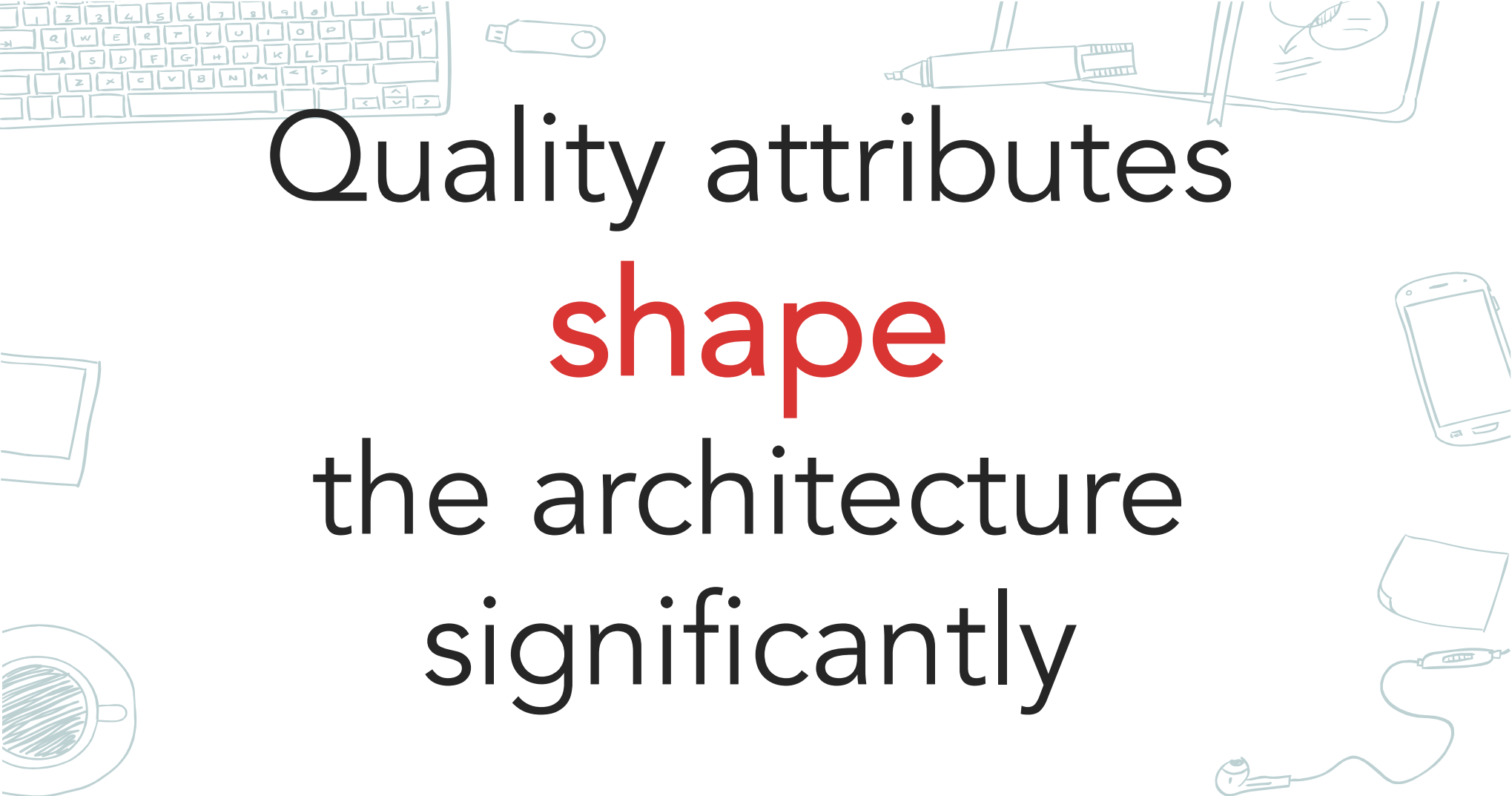
Quality attributes are measurable or testable properties of a system …



* From S. Robertson and J. Robertson. 2012. *Mastering the Requirements Process: Getting Requirements Right* (3rd ed.). Addison-Wesley Professional.

Systems are frequently redesigned not because they are functionally deficient but because

- they are difficult to **maintain** or **scale**;
- they are **too slow**;
- they have been **compromised by hackers**;
- and so on …



Quality attributes shape the architecture significantly

Quality Attributes

"… The system must provide an **efficient way** to both provide and to request data. The pages that display the data should be **light** and the steps required for to both provide and to request data should involve a **very few number** of screens."

Is this quality specification useful?

Quality Attributes

Do not start design without a prioritized list of measurable quality attributes.

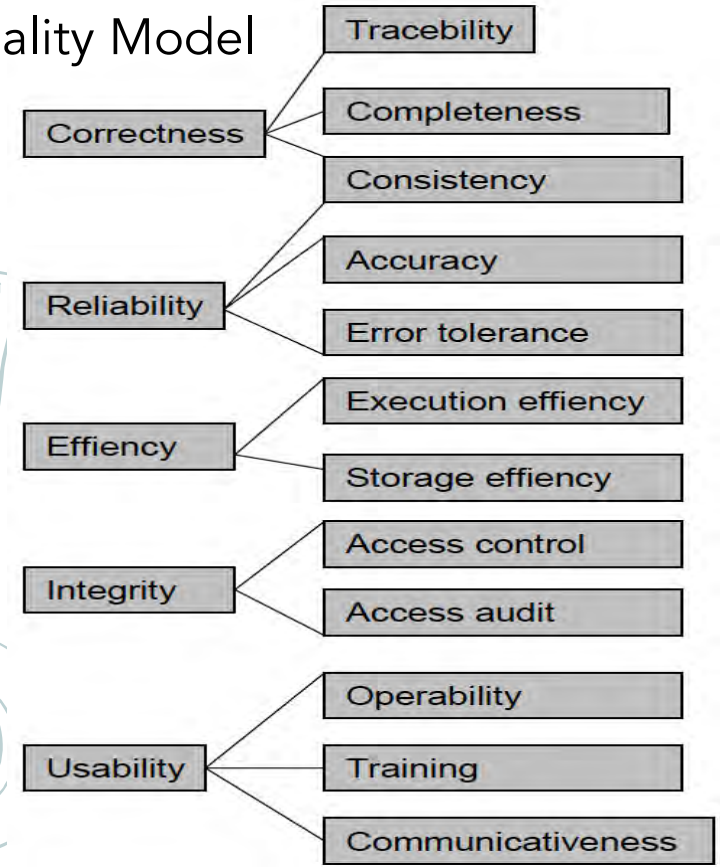
"The system should be "fast"

- Is 10 second response time acceptable?
- Is 5 second response time acceptable?

By asking these questions you can at least "box response time in" to a range.

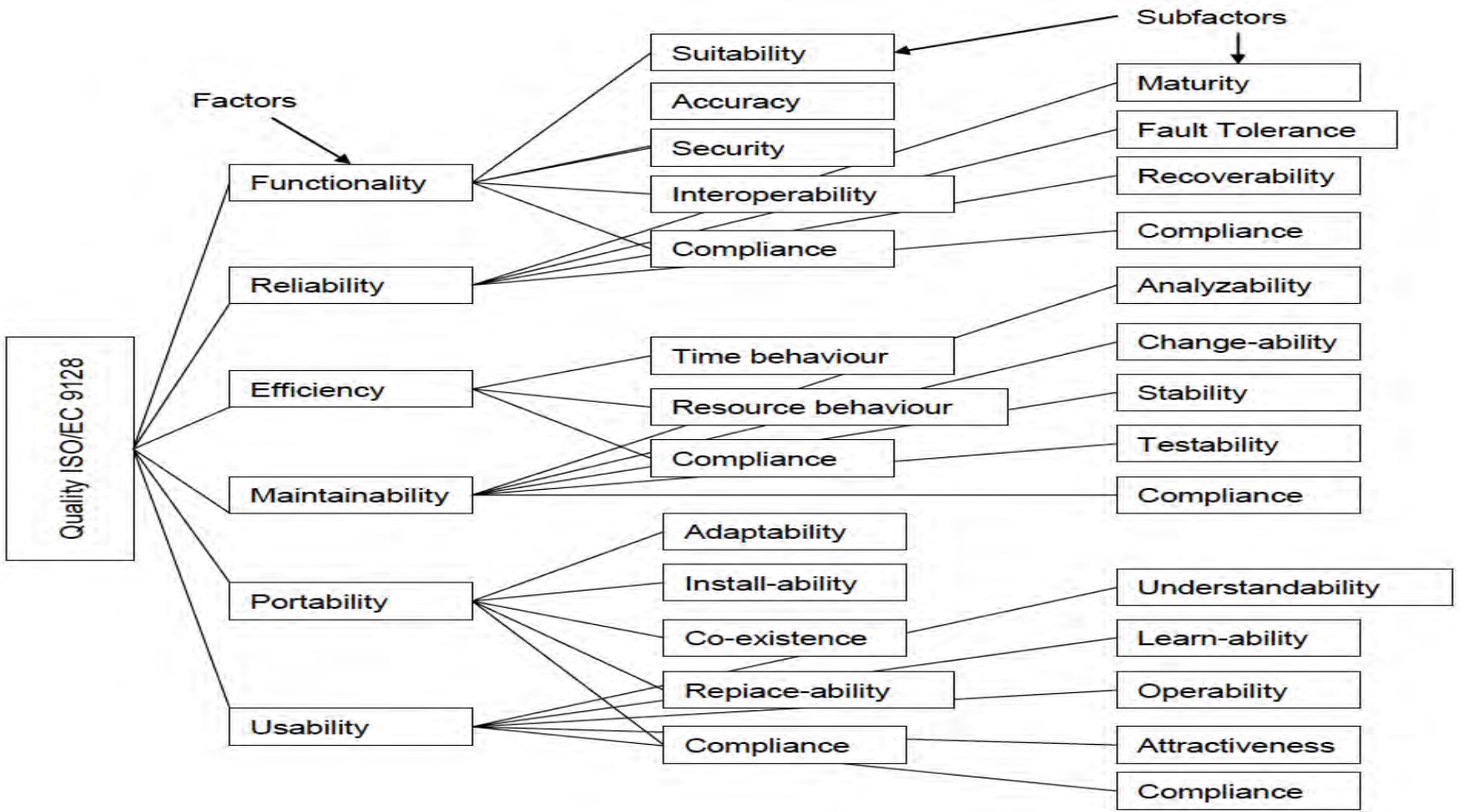
Product Quality Models

McCall's Quality Model



Boehm's Software Quality Characteristics

Product Quality Models



Quality Attributes

External quality attributes:

They describe characteristics that are observed when the software is executing.

Internal quality attributes:

They are not directly observable during execution of the software. They are properties that a developer or maintainer perceives while looking at the design or code to modify it, reuse it, or move it to another platform.

Quality Attributes

External quality	Brief description
Availability	The extent to which the system's services are available when and where they are needed
Installability	How easy it is to correctly install, uninstall, and reinstall the application
Integrity	The extent to which the system protects against data inaccuracy and loss
Interoperability	How easily the system can interconnect and exchange data with other systems or components
Performance	How quickly and predictably the system responds to user inputs or other events
Reliability	How long the system runs before experiencing a failure
Robustness	How well the system responds to unexpected operating conditions
Safety	How well the system protects against injury or damage
Security	How well the system protects against unauthorized access to the application and its data
Usability	How easy it is for people to learn, remember, and use the system

Internal quality	Brief description
Efficiency	How efficiently the system uses computer resources
Modifiability	How easy it is to maintain, change, enhance, and restructure the system
Portability	How easily the system can be made to work in other operating environments
Reusability	To what extent components can be used in other systems
Scalability	How easily the system can grow to handle more users, transactions, servers, or other extensions
Verifiability	How readily developers and testers can confirm that the software was implemented correctly

* Fuente: Software Requirements, 3rd Edition, by Karl E. Wiegers, Microsoft Press, 2013.

Quality Attributes

Category	Quality attribute	Description
Design Qualities	*Conceptual Integrity*	Conceptual integrity defines the consistency and coherence of the overall design. This includes the way that components or modules are designed, as well as factors such as coding style and variable naming.
	Maintainability	Maintainability is the ability of the system to undergo changes with a degree of ease. These changes could impact components, services, features, and interfaces when adding or changing the functionality, fixing errors, and meeting new business requirements.
	Reusability	Reusability defines the capability for components and subsystems to be suitable for use in other applications and in other scenarios. Reusability minimizes the duplication of components and also the implementation time.

Category	Quality attribute	Description
Run-time Qualities	*Availability*	Availability defines the proportion of time that the system is functional and working. It can be measured as a percentage of the total system downtime over a predefined period. Availability will be affected by system errors, infrastructure problems, malicious attacks, and system load.
	Interoperability	Interoperability is the ability of a system or different systems to operate successfully by communicating and exchanging information with other external systems written and run by external parties. An interoperable system makes it easier to exchange and reuse information internally as well as externally.
	Manageability	Manageability defines how easy it is for system administrators to manage the application, usually through sufficient and useful instrumentation exposed for use in monitoring systems and for debugging and performance tuning.
	Performance	Performance is an indication of the responsiveness of a system to execute any action within a given time interval. It can be measured in terms of latency or throughput. Latency is the time taken to respond to any event. Throughput is the number of events that take place within a given amount of time.
	Reliability	Reliability is the ability of a system to remain operational over time. Reliability is measured as the probability that a system will not fail to perform its intended functions over a specified time interval.
	Scalability	Scalability is ability of a system to either handle increases in load without impact on the performance of the system, or the ability to be readily enlarged.
	Security	Security is the capability of a system to prevent malicious or accidental actions outside of the designed usage, and to prevent disclosure or loss of information. A secure system aims to protect assets and prevent unauthorized modification of information.

Quality Attributes

Category	Quality attribute	Description
System Qualities	*Supportability*	Supportability is the ability of the system to provide information helpful for identifying and resolving issues when it fails to work correctly.
	Testability	Testability is a measure of how easy it is to create test criteria for the system and its components, and to execute these tests in order to determine if the criteria are met. Good testability makes it more likely that faults in a system can be isolated in a timely and effective manner.
User Qualities	*Usability*	Usability defines how well the application meets the requirements of the user and consumer by being intuitive, easy to localize and globalize, providing good access for disabled users, and resulting in a good overall user experience.

Tradeoffs

	Availability	Efficiency	Installability	Integrity	Interoperability	Modifiability	Performance	Portability	Reliability	Reusability	Robustness	Safety	Scalability	Security	Usability	Verifiability
Availability									+		+					
Efficiency	+				−	−	+	−			−		+		−	
Installability	+								+					+		
Integrity			−		−		−			−		+		+	−	−
Interoperability	+		−	−			−	+	+		+	−		−		
Modifiability	+		−				−		+	+			+			+
Performance		+			−	−		−			−		−		−	
Portability		−			+	−	−			+				−	−	+
Reliability	+	−		+		+	−				+	+		+	+	+
Reusability		−		−	+	+	−	+						−		+
Robustness	+	−	+	+	+		−		+			+	+	+	+	
Safety		−		+	+		−				+			+	−	−
Scalability	+	+		+			+	+	+		+					
Security	+			+	+		−	−	+		+	+			−	−
Usability		−	+				−	−	+		+	+				−
Verifiability	+		+	+		+			+	+	+	+		+	+	

Key:

[-] Increasing the attribute in that row generally adversely affects the attribute in the column.

[+] Increasing the attribute in that row generally positively affects the attribute in the column.

[ ] The attribute in the row has little effect on the attribute in the column.

Quality Attributes

A criteria to consider a quality attribute as architectural includes:

a) it is critical to achieve the business goals that motivate the development of the system.

b) it implies a high level of technical difficulty

Scenarios and Business Goals

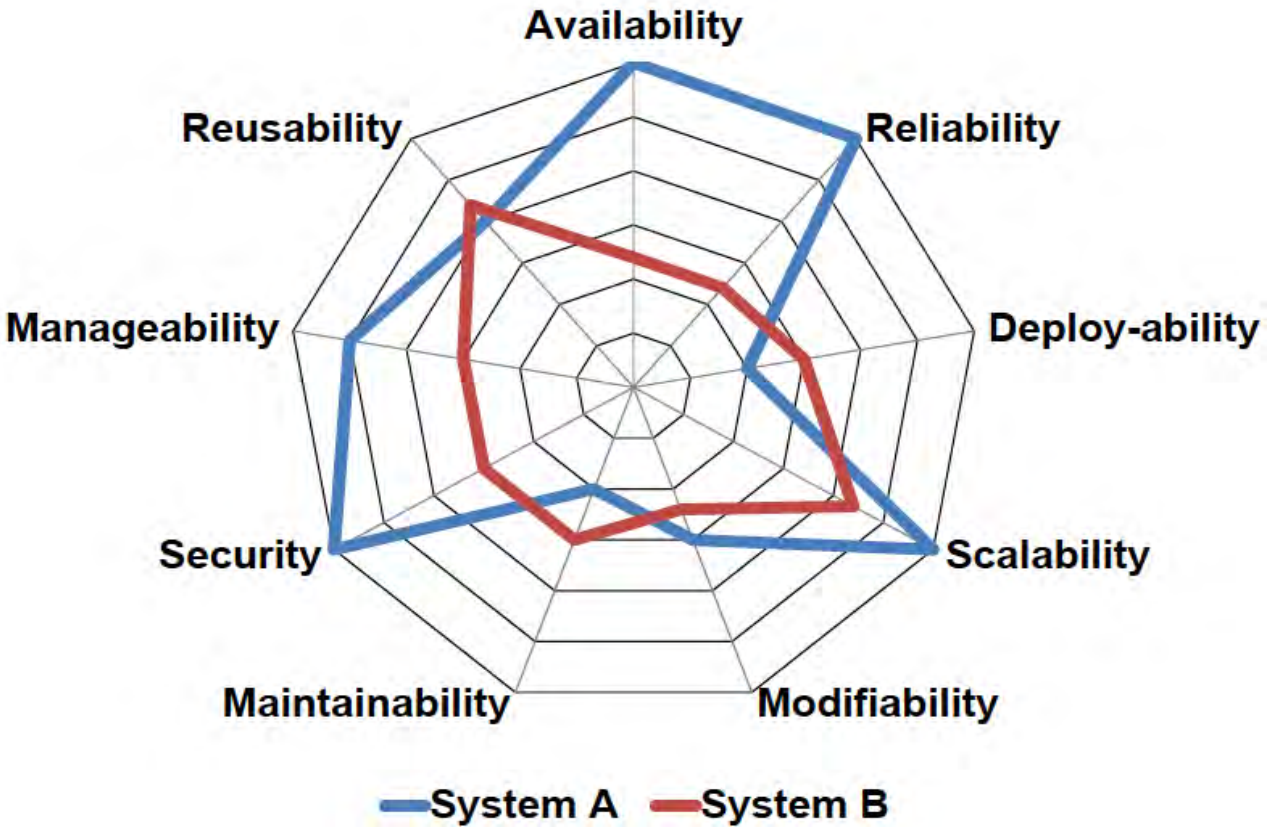
Table 1. Business goals and quality attribute scenarios for the software management system.

Business goal	Goal refinement	Quality attribute	Quality attribute scenario	Priority
Open new sales channels in the form of VARs	Support hardware devices from different manufacturers	Modifiability	Two developers integrate a new device into the system in 320 person-hours.	High, High
	Support conversions of nonstandard units the different devices use	Modifiability	A system administrator configures the system to handle units from a newly plugged-in field device in less than three hours.	High, Medium
Expand by entering new and emerging geographic	Support international languages	Modifiability	A developer packages a version of the system with new language support in 80 person-hours.	High, Medium
	Support regulations that require life-critical systems, such as fire alarms, to operate within specific latency constraints	Performance	A life-critical alarm reports to concerned users within three seconds of the event.	High, High

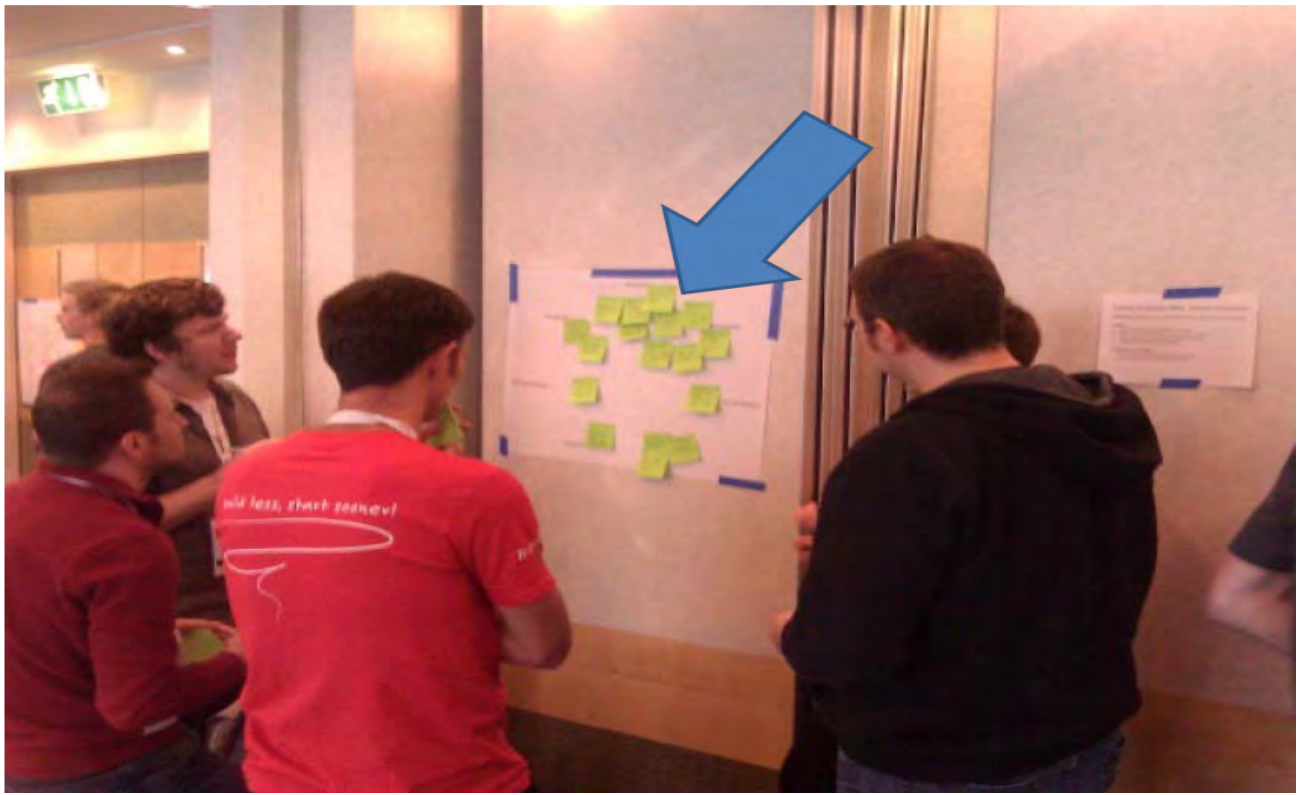
Outline

1. Quality Attributes Defined
2. Walk the System Properties Web
3. Writing Quality Attributes as Scenarios
4. Writing Quality Attributes as Acceptance Criteria
5. Summary

Same Quality Attributes, Different Systems



Walk System Properties Web




Upgrade-ability
Reusability
Availability
Manageability
Reliability
Security
Build-ability
Deploy-ability
Maintainability
Scalability
Modifiability

Walking the System Properties Web

- Goal: Guide stakeholders in identifying highly desirable quality attributes.
- Who: Key stakeholders, project managers, IT, user champions, subject experts, development team.
- Outcome: Raw quality attribute specs.
- Timeframe: Depends on stakeholders, risk, complexity. It should be a time boxed activity, ends when time runs out.

Walk System Properties Web

Objective: Identify and prioritize raw quality attribute requirements.

Time Limit: [30 minutes to 2-3 hours]

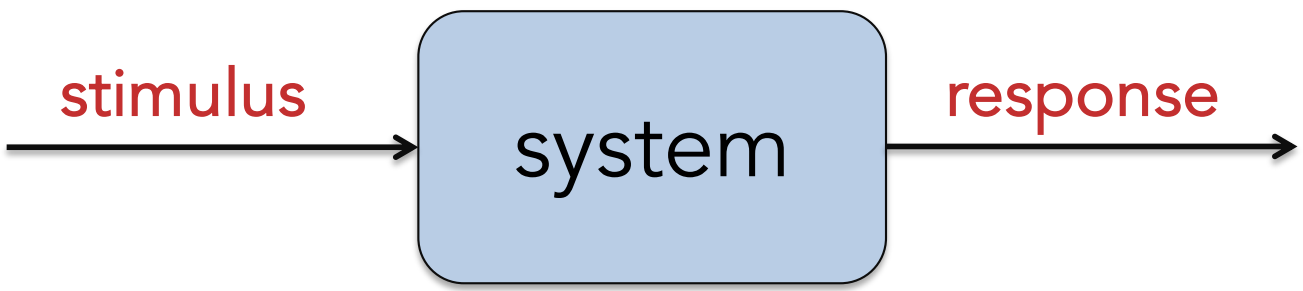
Guidelines and hints:

- Put the sticky close to related attributes
- What are you worried about?
- Don't worry about creating detailed specs, but think about stimulus, response

- **Look at the user requirements!**

Stimulus, Response, Environment

- A good approach to discuss quality attribute requirements.
- In its most basic form, describes the system's response to some stimulus




Upgrade-ability
Reusability
Availability
Manageability
Reliability
Security
Build-ability
Deploy-ability
Maintainability
Modifiability
Scalability

Structured Brainstorming

Process

- 3 - 5 minutes ideation using any method (e.g. silent, round robin, etc) + time for refinement
- Capture ideas directly on the quality attributes web

Pros

- Fast – About 30 - 45 minutes for raw scenario generation

Cons

- May leave areas unexplored
- Requires experienced stakeholders

Taxonomy Questionnaire

Process

- Introduce each quality attribute
  - "Is this quality attribute relevant to your system?"
  - Yes – ask follow up questions
- When time runs out, the activity is over.

Pros

- Thorough, very repeatable.

Cons

- You need a taxonomy
- Workshop runs longer (allow ~2+ hours)

Walk System Properties Web

Objective: Identify and prioritize raw quality attribute requirements.

Time Limit: [30 minutes to 2-3 hours]

Guidelines and hints:

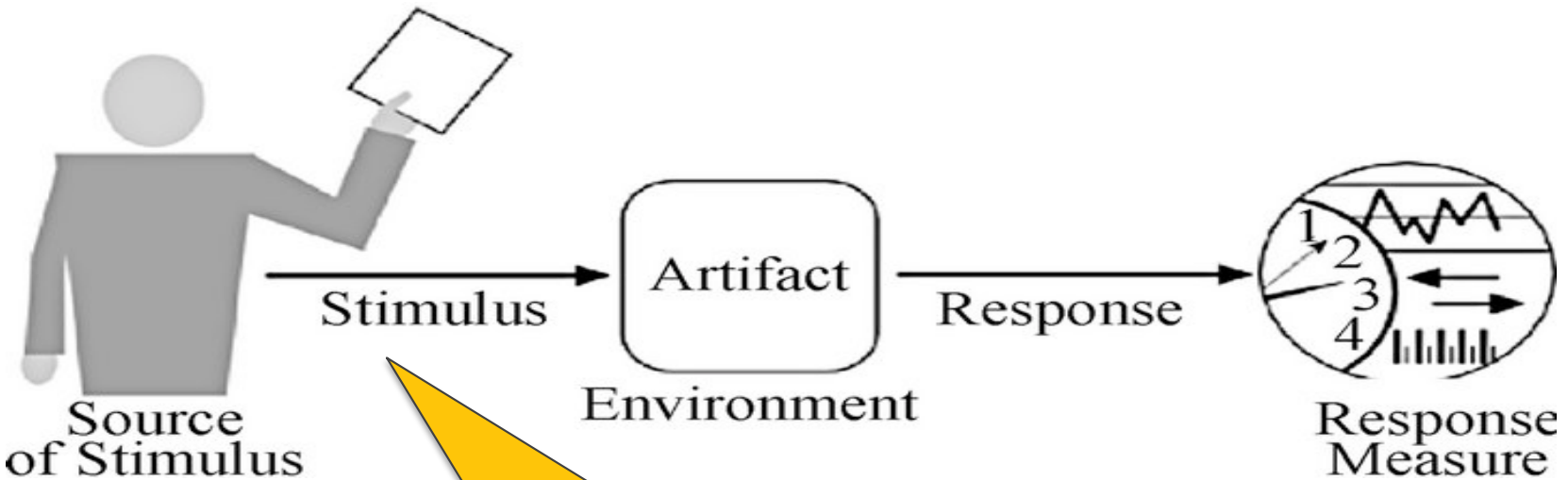
- Put the sticky close to related attributes
- What are you worried about?
- Don't worry about creating detailed specs, but think about stimulus, response

- **Look at the user requirements!**

Outline

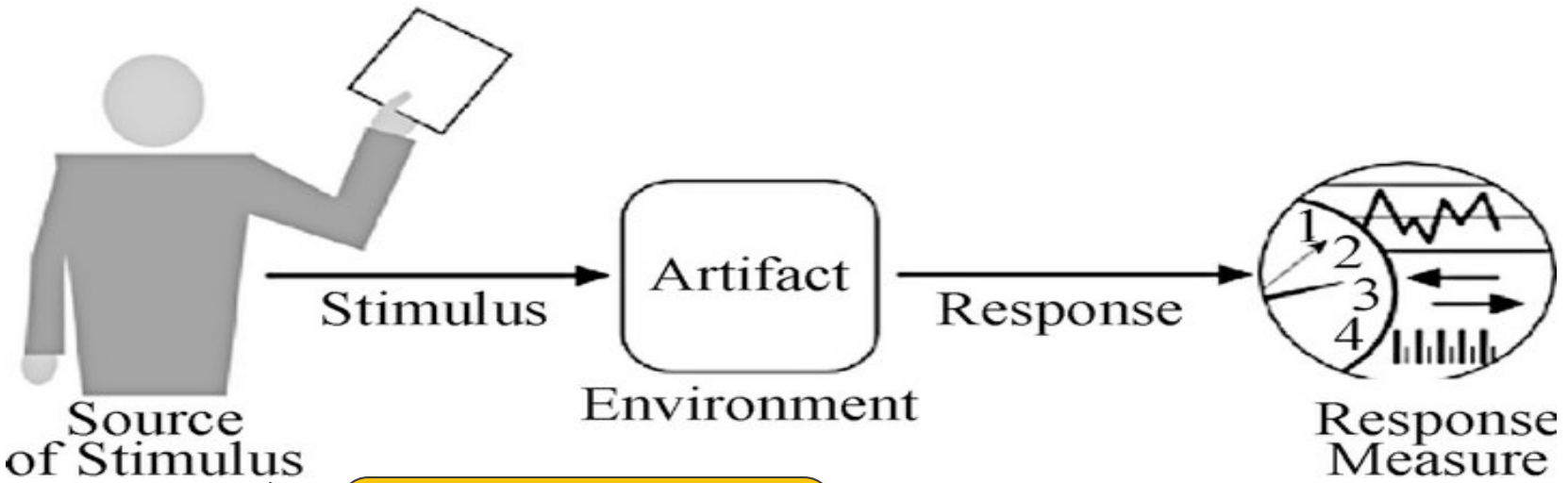
1. Quality Attributes Defined
2. Walk the System Properties Web
3. Writing Quality Attributes as Scenarios
4. Writing Quality Attributes as Acceptance Criteria
5. Summary

Quality Attribute Scenarios

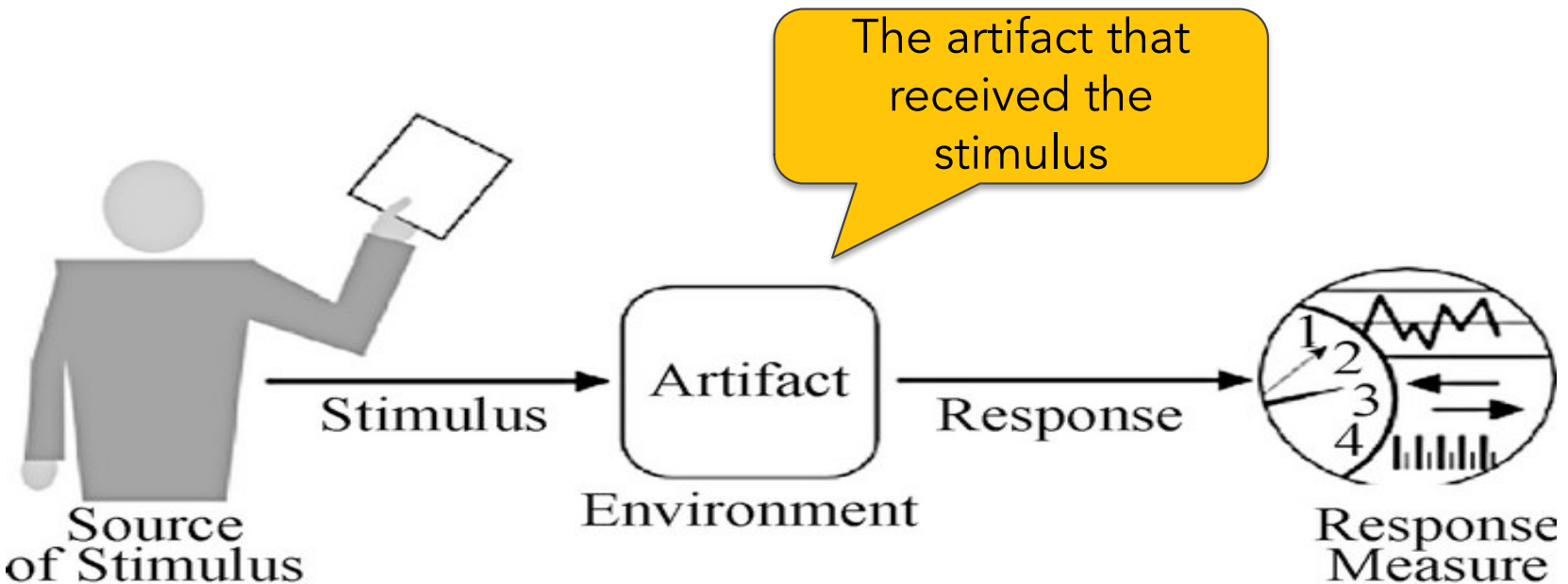


The condition (stimulus) that affects the system

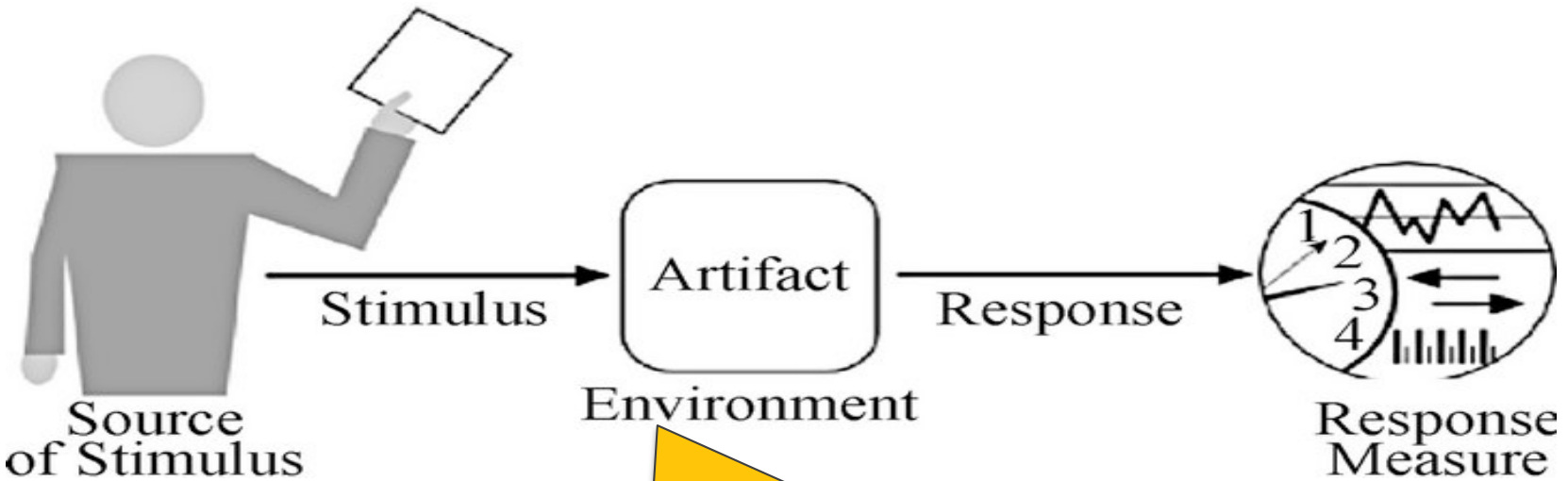
Quality Attribute Scenarios



Quality Attribute Scenarios

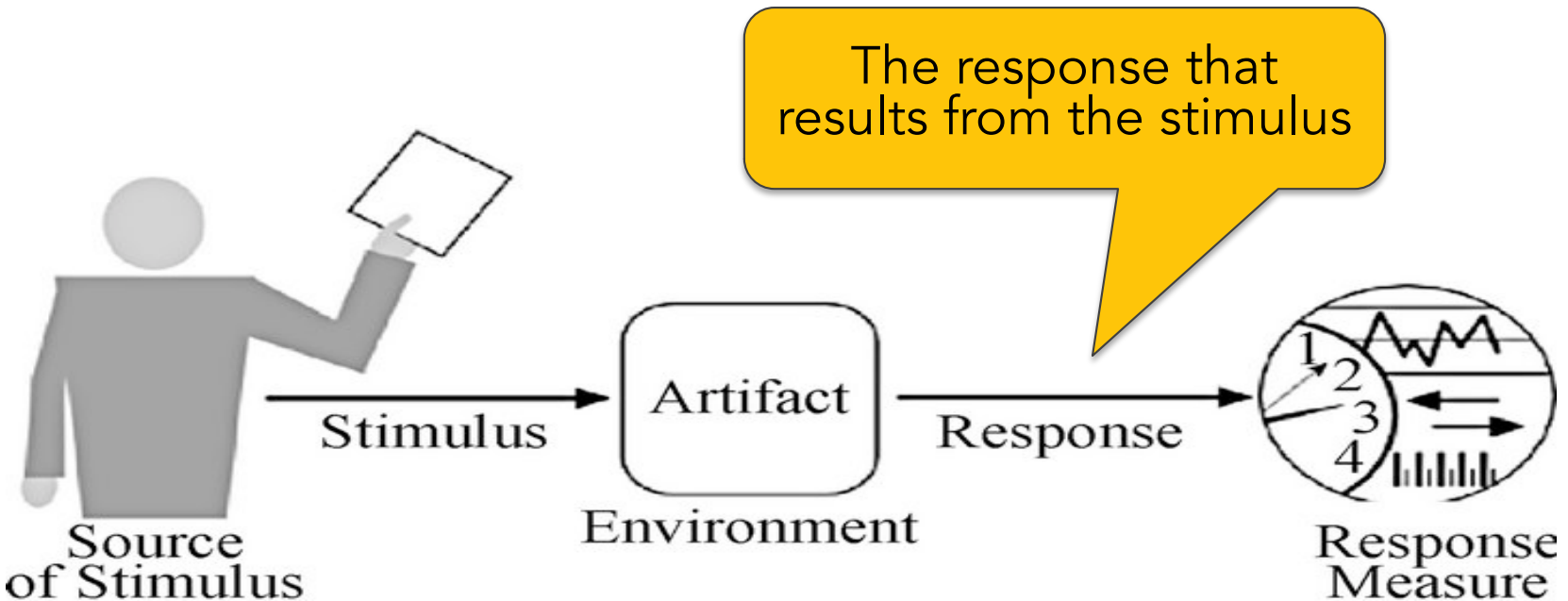


Quality Attribute Scenarios

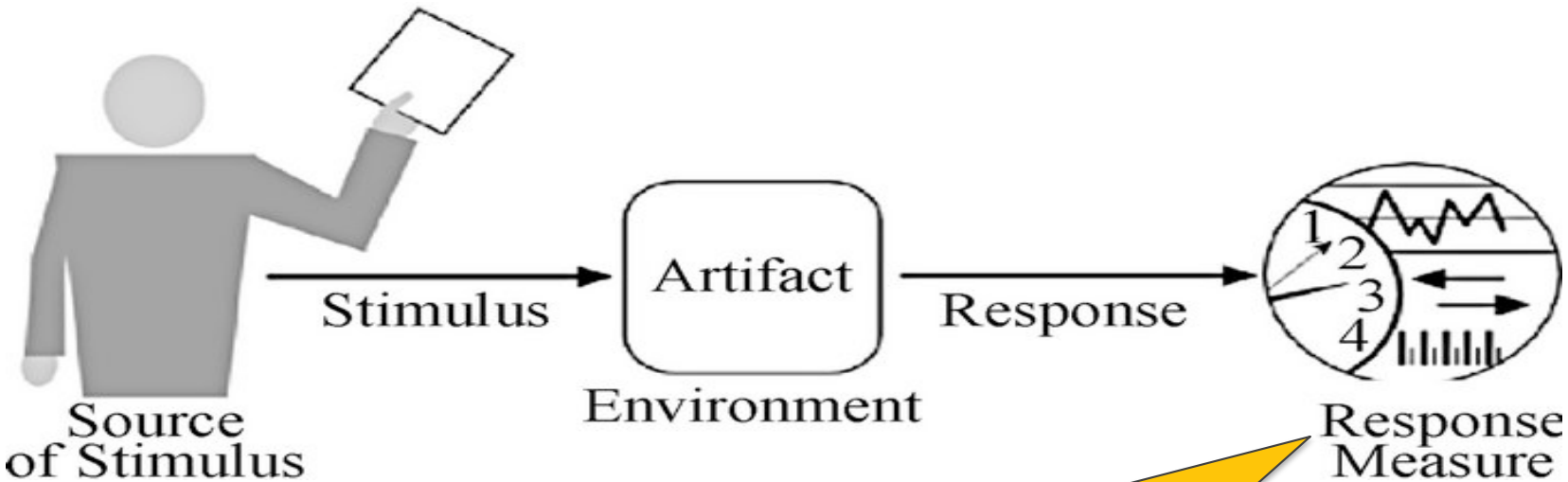


The condition under which the stimulus occurred

Quality Attribute Scenarios

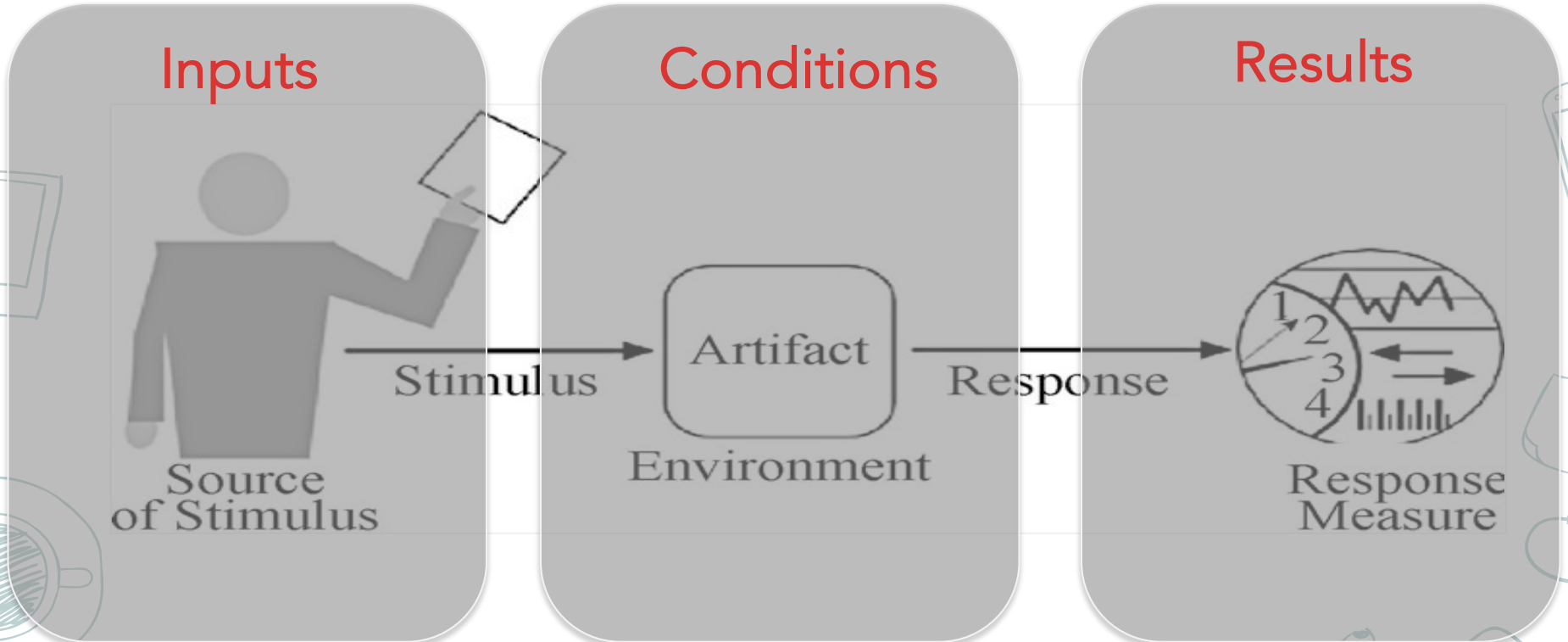


Quality Attribute Scenarios



The measure by which the system's response will be evaluated

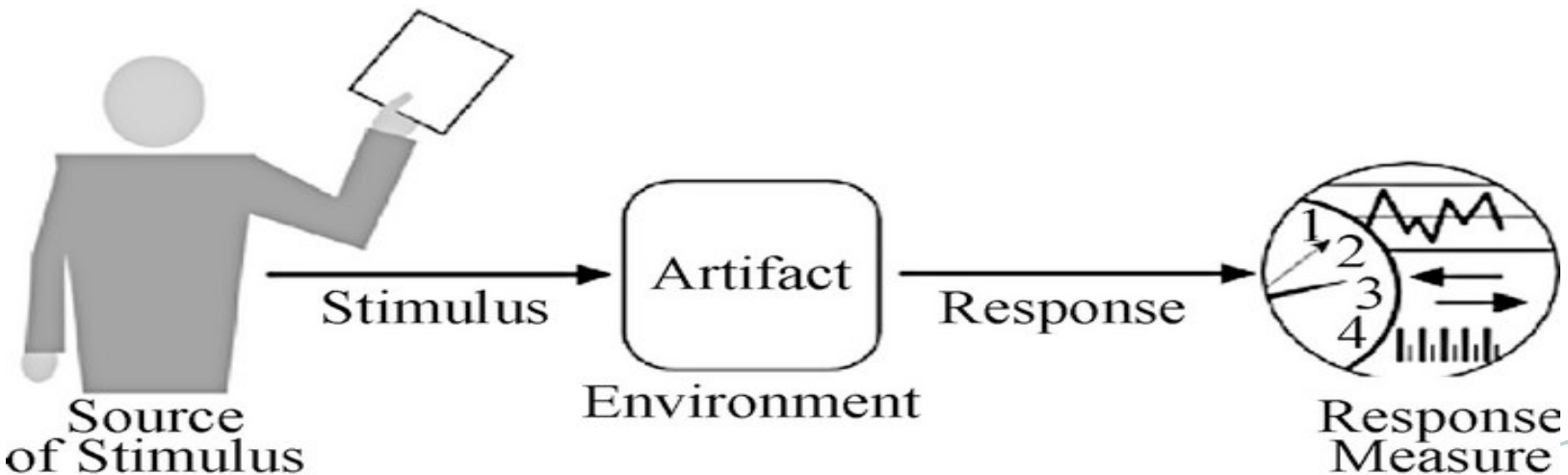
Quality Attribute Scenarios



Quality Attribute Scenario Specification

Performance

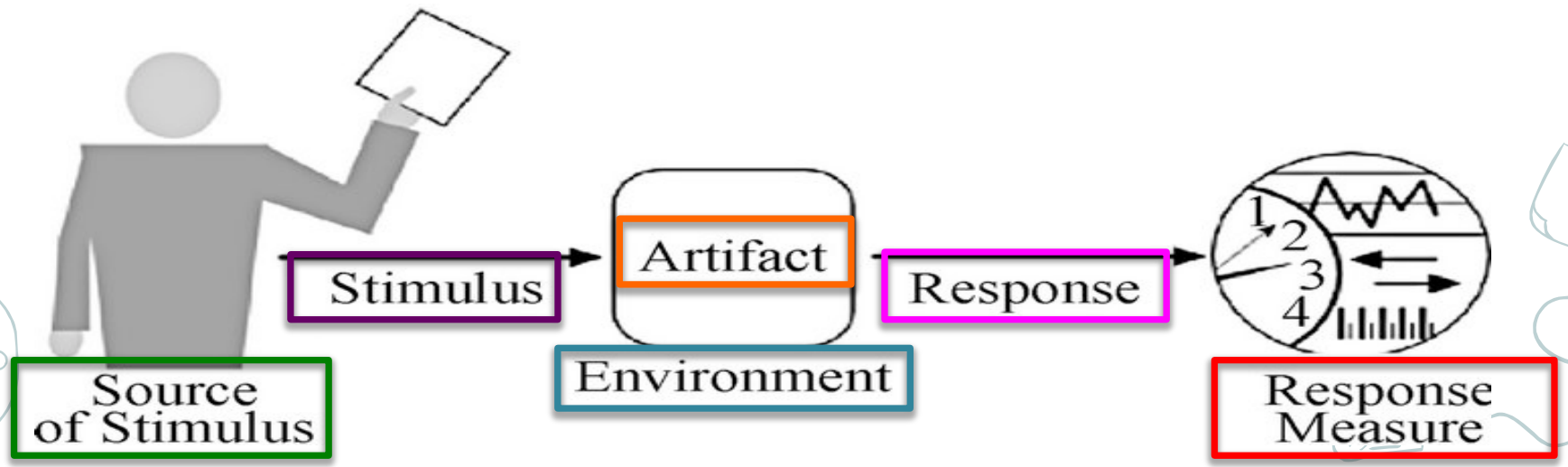
"Users initiate transactions in the system under normal operations. The system processes the transactions with an average latency of two seconds"



Quality Attribute Scenario Specification

Performance

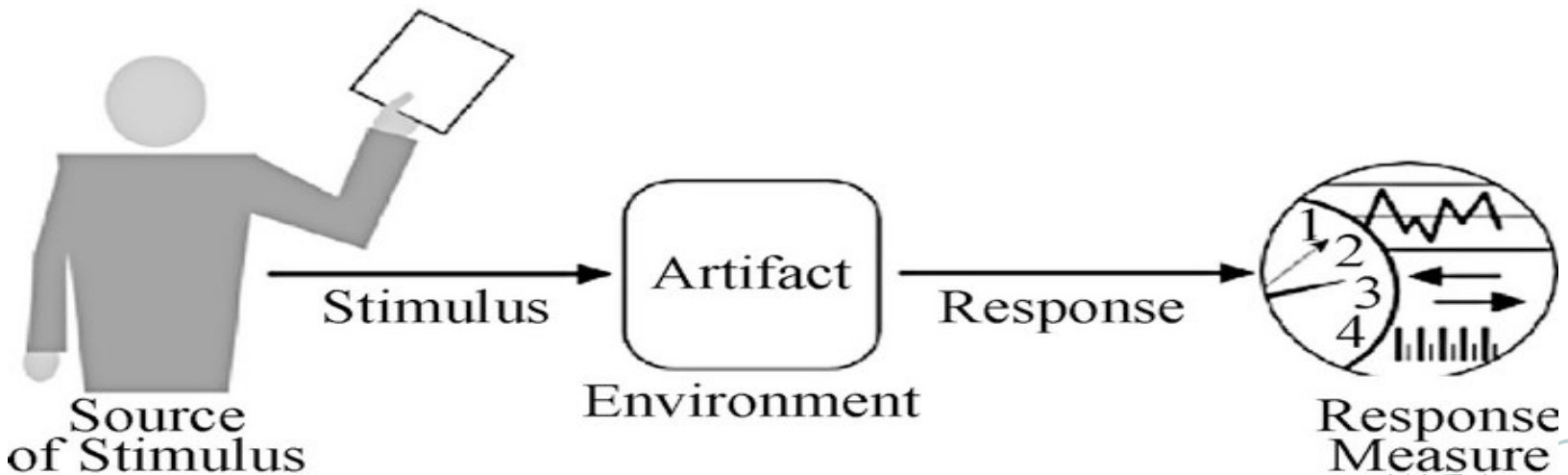
"Users initiate transactions in the system under normal operations. The system processes the transactions with an average latency of two seconds"



Quality Attribute Scenario Specification

Security/Integrity

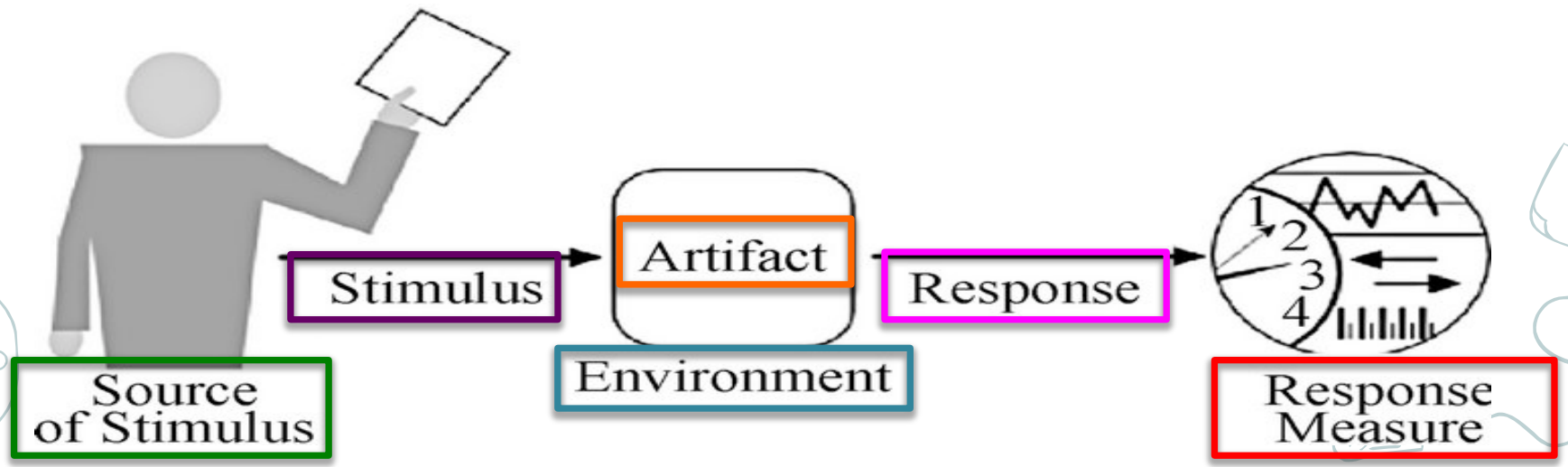
"A disgruntled employee from a remote location attempts to modify the pay rate table during normal operations. The system maintains an audit trail and the correct data is restored within a day"



Quality Attribute Scenario Specification

Security/Integrity

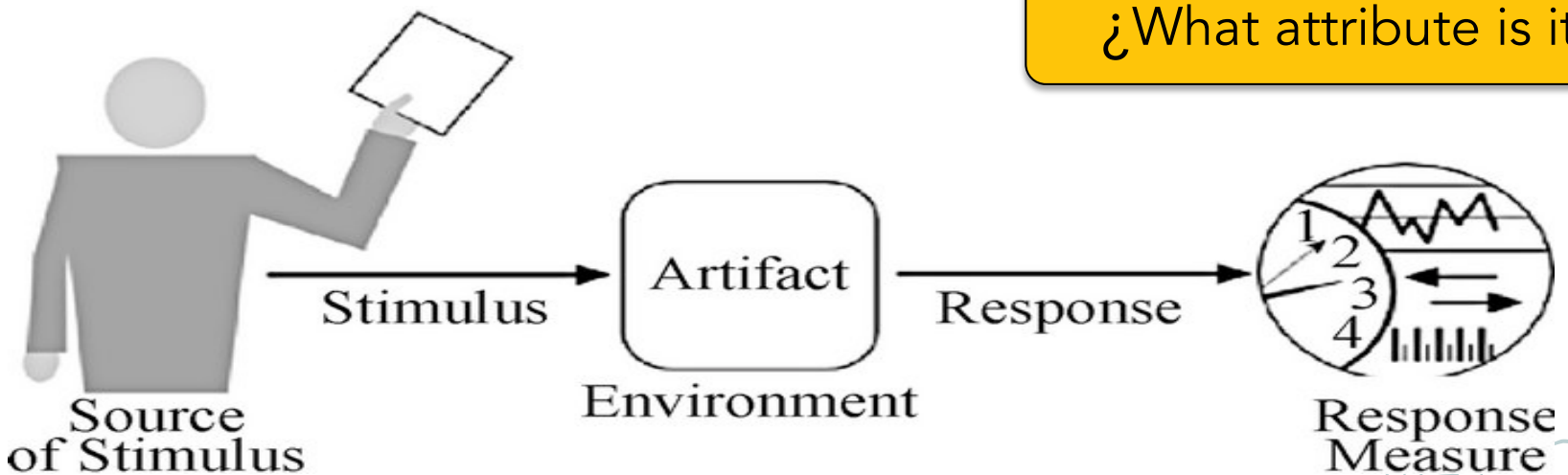
"A disgruntled employee from a remote location attempts to modify the pay rate table during normal operations. The system maintains an audit trail and the correct data is restored within a day"



Quality Attribute Scenario Specification

"The unit tester completes a code unit during development and performs a test sequence whose results are captured and that gives 85% path coverage within 3 hours of testing"

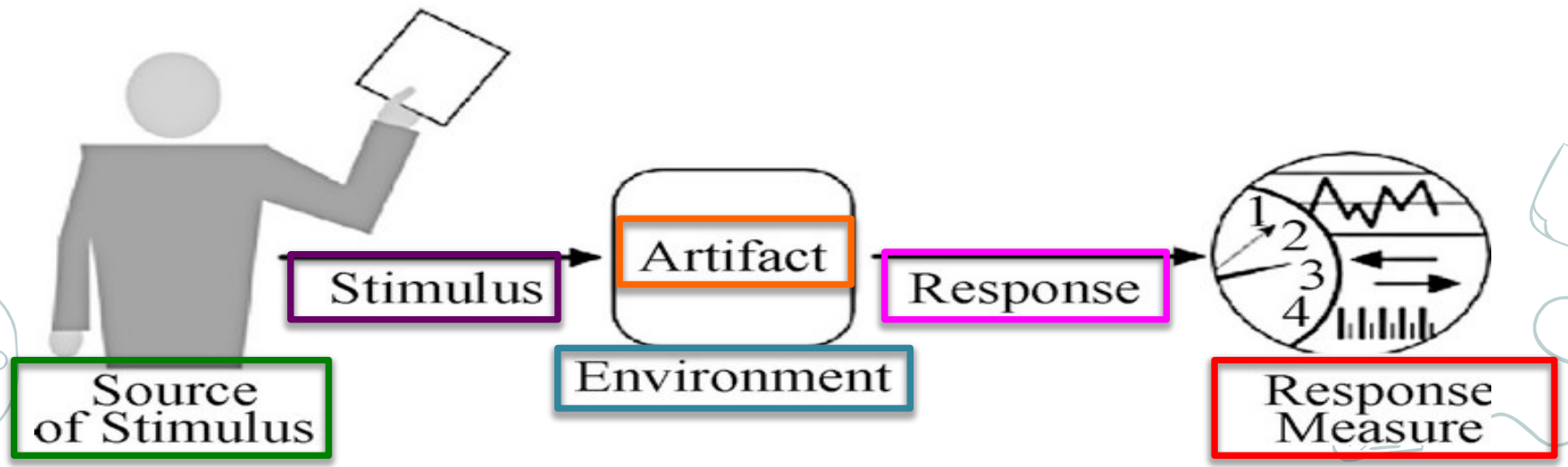
¿What attribute is it?



Quality Attribute Scenario Specification

Testability

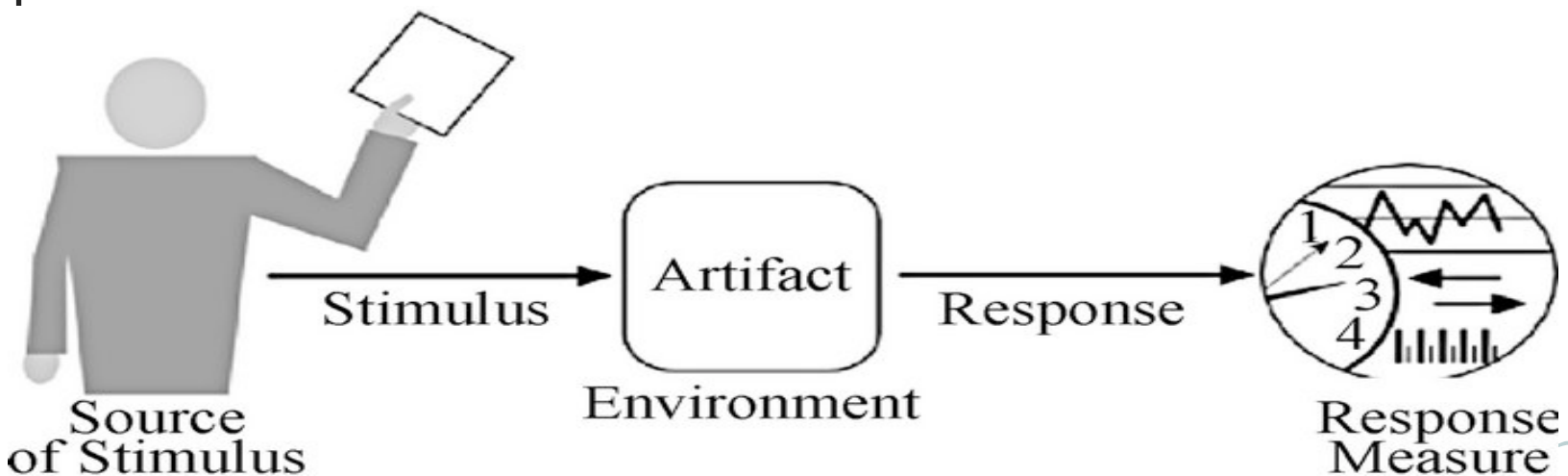
"The unit tester, during development, performs a test sequence a code unit whose results are captured and that gives 85% path coverage within 3 hours of testing"



Quality Attribute Scenario Specification

Usability

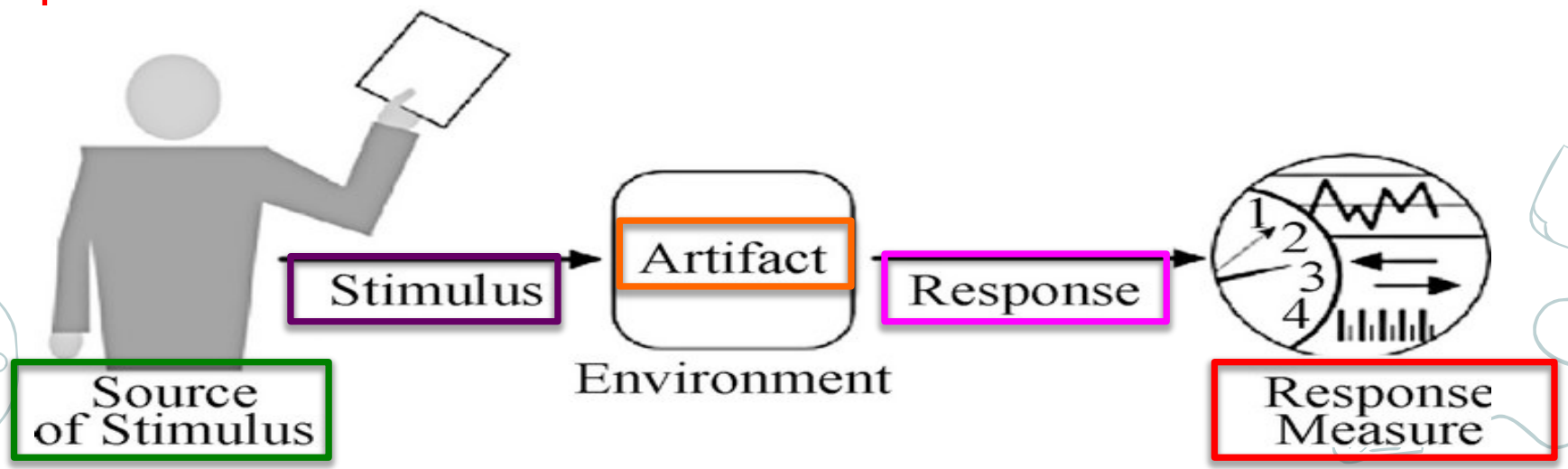
"The user downloads a new application and is using it productively after two minutes of experimentation"



Quality Attribute Scenario Specification

Usability

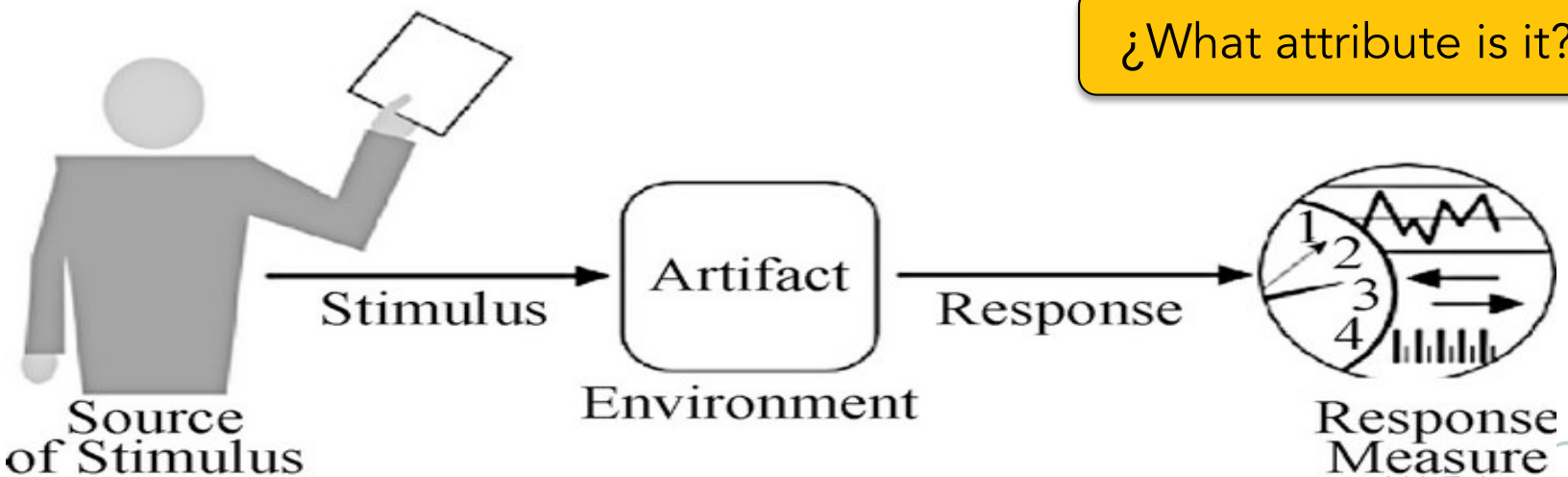
"The user downloads a new application and is using it without mistakes after two minutes of experimentation"



Quality Attribute Scenario Specification

"Our vehicle information system sends our current location to the traffic monitoring system. The traffic monitoring system combines our location with other information, overlays this information on a Google Map, and broadcasts it. Our location information is correctly included with a probability of 99.9%"

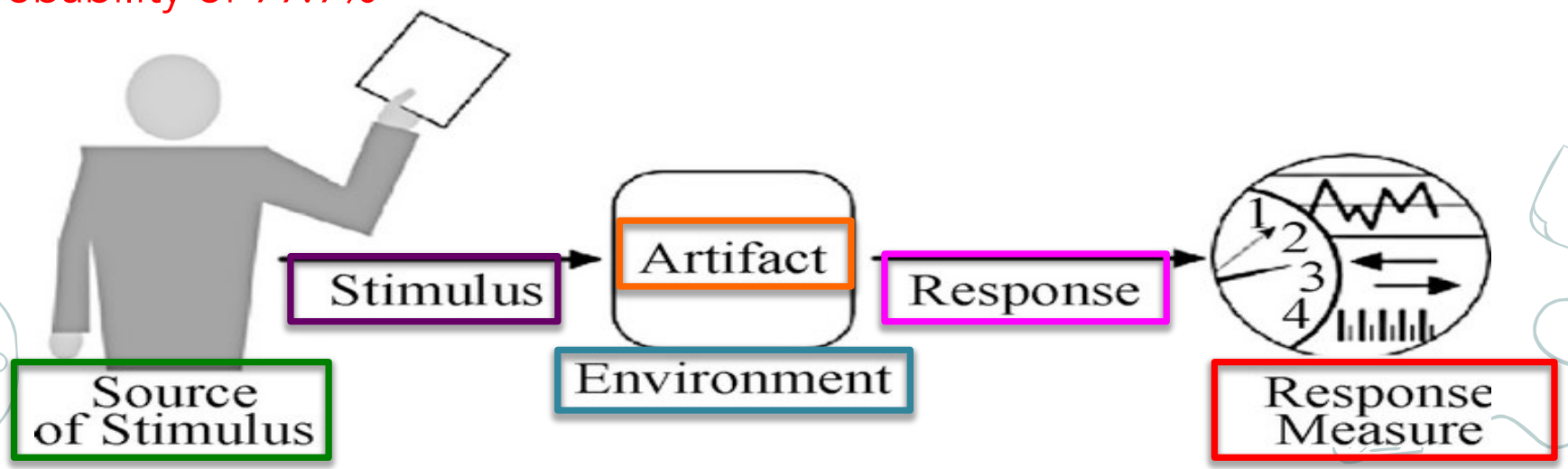
¿What attribute is it?



Quality Attribute Scenario Specification

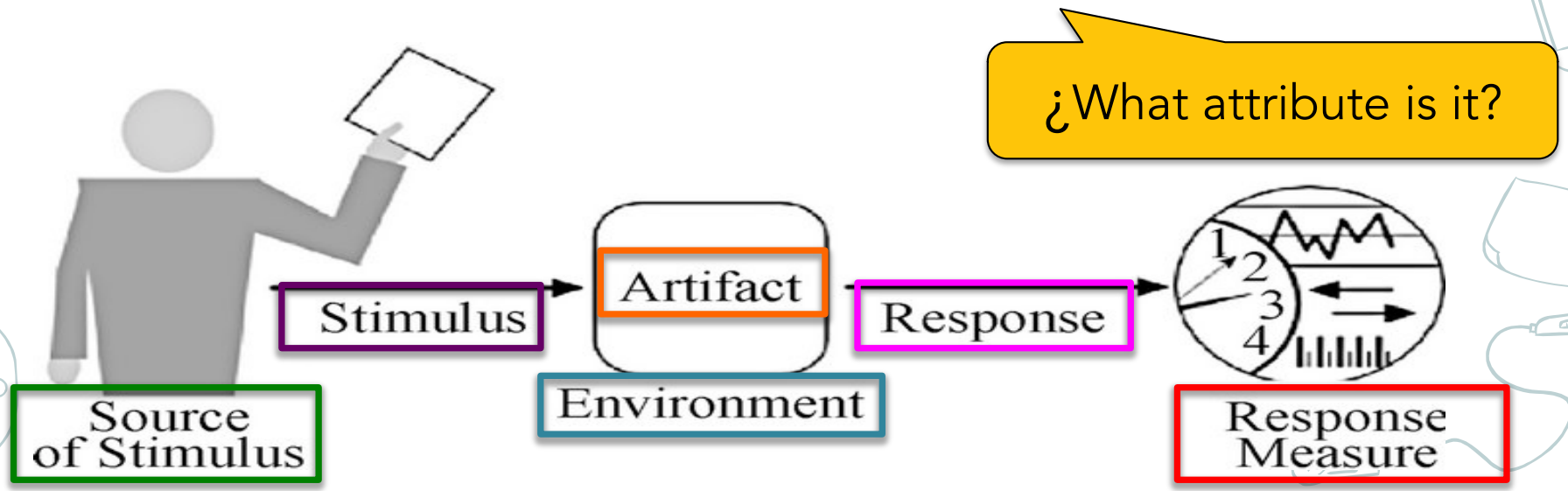
Interoperability/Reliability

"The vehicle information **system** sends its current location to our traffic monitoring system during peak time. Our traffic monitoring system combines our location with other information, overlays this information on a Google Map, and broadcasts it. The location information is correctly received with a probability of 99.9%"



Quality Attribute Scenario Specification

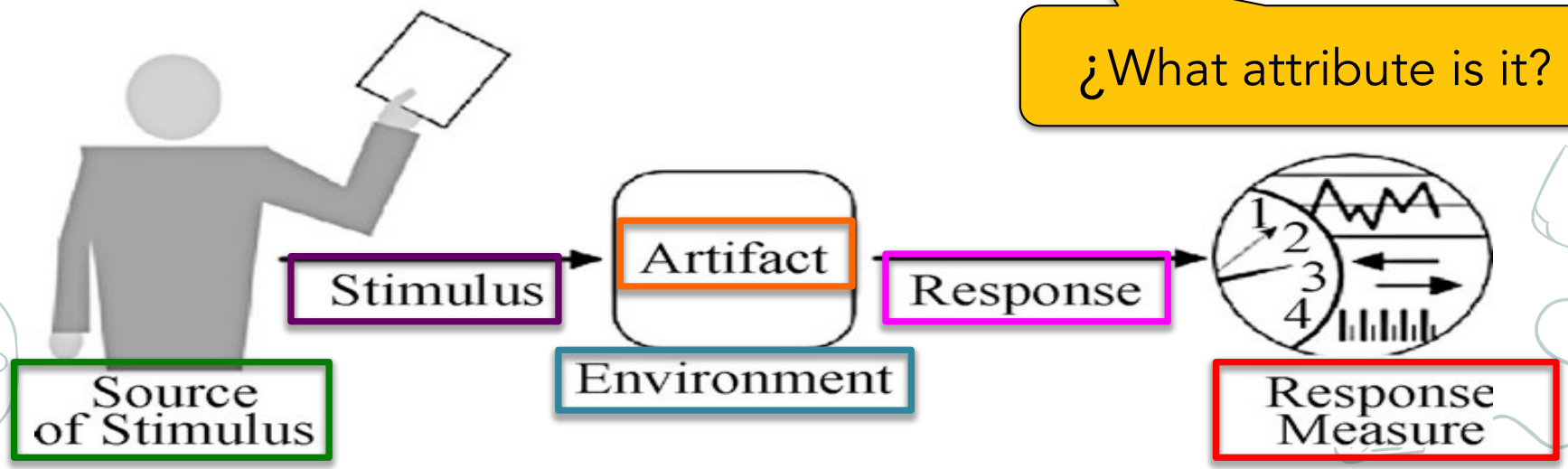
"The developer wishes to change the user interface by modifying the code at design time. The modifications are made with no side effects within three hours"



Quality Attribute Scenario Specification

Maintainability/Modifiability

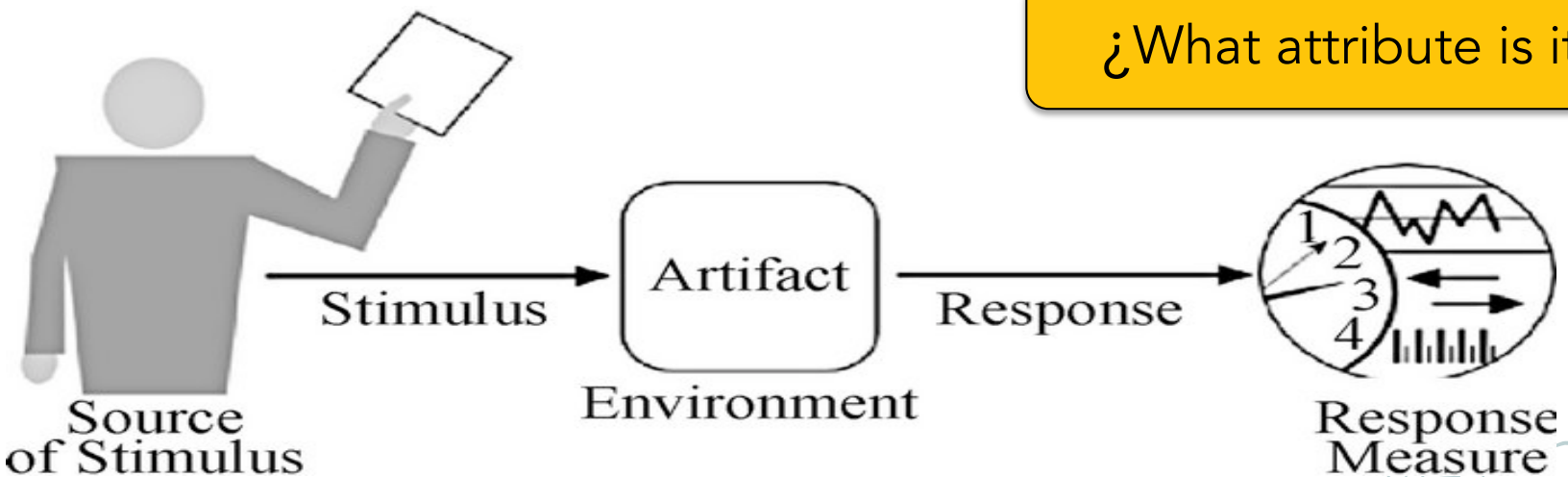
"The developer wishes to change the user interface by modifying the code at design time. The modifications are made with no side effects within three hours"



Quality Attribute Scenario Specification

"The heartbeat monitor determines that the server is nonresponsive during normal operations. The system informs the operator and continues to operate with no downtime"

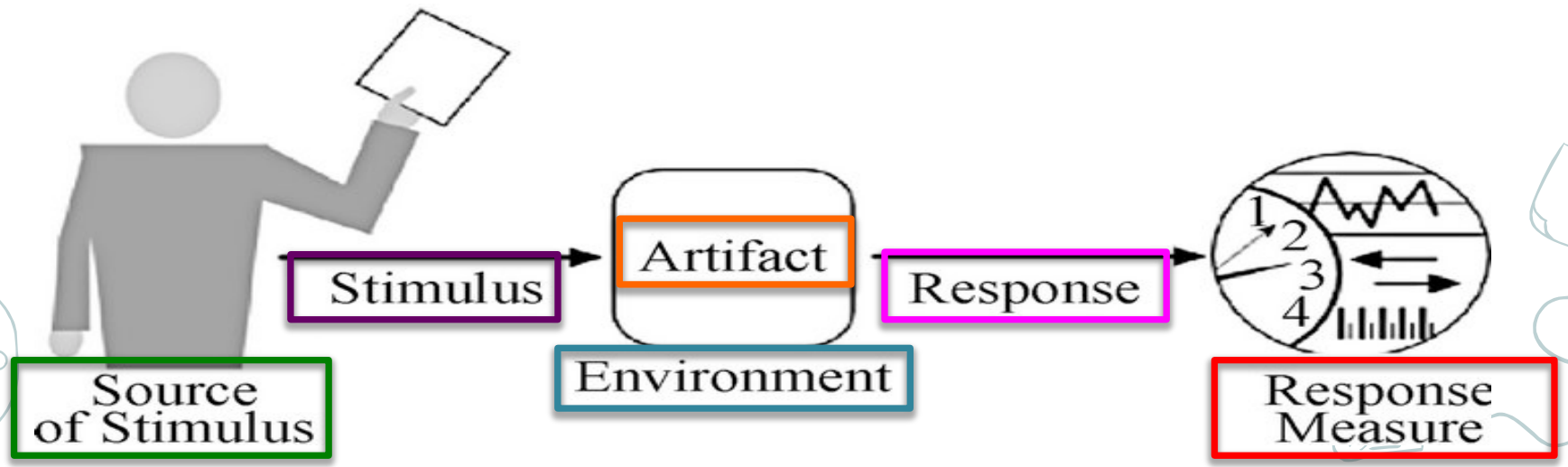
¿What attribute is it?



Quality Attribute Scenario Specification

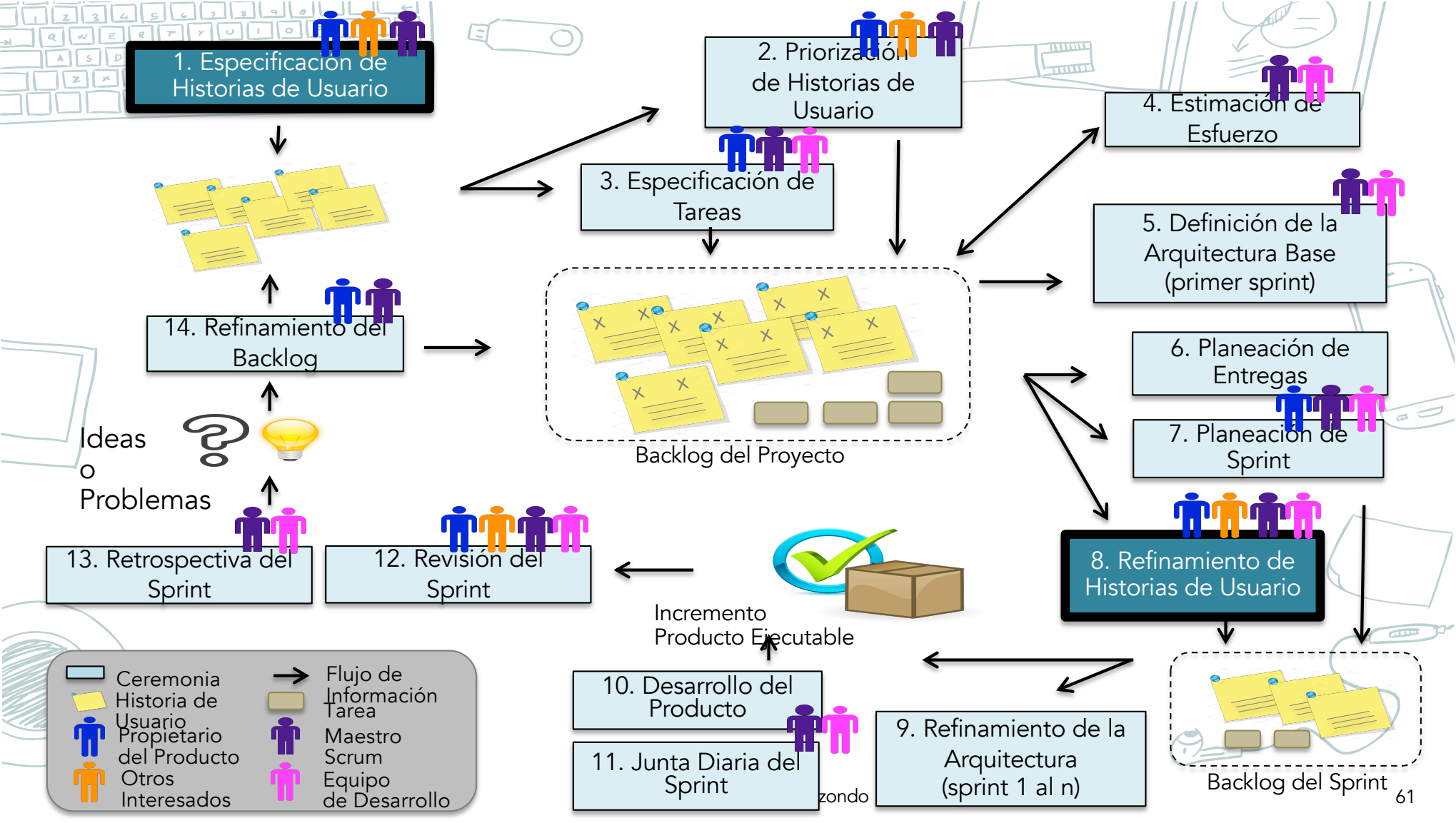
Availability

"The heartbeat monitor determines that the server is nonresponsive during normal operations. The system informs the operator and recovers in no more than 2 minutes"



Outline

1. Quality Attributes Defined
2. Walk the System Properties Web
3. Writing Quality Attributes as Scenarios
4. Writing Quality Attributes as Acceptance Criteria
5. Summary


1. Especificación de Historias de Usuario
2. Priorización de Historias de Usuario
3. Especificación de Tareas
4. Estimación de Esfuerzo
5. Definición de la Arquitectura Base (primer sprint)
6. Planeación de Entregas
7. Planeación de Sprint
8. Refinamiento de Historias de Usuario
9. Refinamiento de la Arquitectura (sprint 1 al n)
10. Desarrollo del Producto
11. Junta Diaria del Sprint
12. Revisión del Sprint
13. Retrospectiva del Sprint
14. Refinamiento del Backlog
Backlog del Proyecto
Backlog del Sprint
Incremento Producto Ejecutable
Ideas o Problemas
Ceremonia
Historia de Usuario
Propietario del Producto
Otros Interesados
Flujo de Información
Tarea
Maestro Scrum
Equipo de Desarrollo

User Stories Refinement

- Performed by PO, SM and Team
- It is about defining the acceptance criteria for user stories

Acceptance Criteria

Given/IF [an initial context]
and [some more context]…
When [some event happens]
and [another event]…
Then [an expected outcome]
and [another outcome]…

Acceptance Criteria

IF se está capturando información

WHEN se observa un comportamiento anormal del sistema

THEN el sistema entrará en modo de operación degradado mientras se realizan las acciones reparación.

¿Qué atributo de calidad es?

Acceptance Criteria

IF no existe información previa

and el archivo *no excede* en tamaño 5M

WHEN presiona el botón "importar archivo"

THEN el sistema importará la información

and no deberán pasar más de 5 segundos para esto

¿Qué atributo de calidad es?

Outline

1. Quality Attributes Defined
2. Walk the System Properties Web
3. Writing Quality Attributes as Scenarios
4. Writing Quality Attributes as Acceptance Criteria
5. Summary

Summary

- Quality Attributes Defined
- Walk the System Properties Web
- Writing Quality Attributes as Scenarios
- Writing Quality Attributes as Acceptance Criteria

Questions?
Comments?