

Ingeniería de Software II

Primer Cuatrimestre de 2010

Clase 5: Especificación de Atributos de Calidad y QAW

Buenos Aires, 8 de Abril de 2010

Una historia real

- Reunión por una gran licitación entre el contratante y 5 oferentes
- O1: "¿Los cientos de puntos de acceso en todo el país, van a tener conectividad?"
- C: "Si, van a tener todos banda ancha"
- O2: "¿Qué tipo de arquitectura están esperando?"
- C: "Bueno, nosotros queremos un sistema Web"
- O1: "Pero... ¿Siempre van a tener conectividad?"
- C: "Y... en algunos lugares es complicado, así que hay que tener en cuenta eso. Podría ser una parte Web y una parte no-web"
- O2: "Claro... una parte cliente servidor y otra parte Web."
- C: "Si, no hace falta que funcione todo si no hay conexión, así que la parte cliente servidor podría ser más chica que la parte Web"
- ... varios minutos más de discusión...
- **O3: "Perdón, ¿Por qué quieren un sistema Web?"**

Los Atributos de Calidad

- La funcionalidad es sólo una parte de lo que un sistema debe hacer.
- Además, están los atributos de calidad (“ilities”), que hablan de características específicas que debe tener el sistema (anteriormente llamados “requerimientos no funcionales”).
 - Ejemplo: portabilidad, flexibilidad, usabilidad
- Necesitamos conocerlos para definir una arquitectura
- En muchos casos, los atributos de calidad se afectan entre si. Por ejemplo, portabilidad vs. performance o flexibilidad vs. Performance
- *“Software quality is the degree to which software possesses a desired combination of attributes.”*

[IEEE Std. 1061]

Algunas realidades sobre los atributos de calidad

- ▶ Suelen estar pobremente especificados, o directamente no especificados (“un requerimiento que no es testeable no es implementable”)
- ▶ En general no se analizan sus dependencias
- ▶ La importancia de estos atributos varía con el dominio para el cual se construye el software
- ▶ Además de requerimientos funcionales y atributos de calidad, el ingeniero de software debe identificar correctamente restricciones.
- ▶ Las “tácticas” de arquitectura no son fines en si mismas, son formas de alcanzar atributos de calidad deseados
- ▶ El atributos de calidad que suele ser más importante: la flexibilidad (“facilidad de cambios”)

Ingeniería de Requerimientos

- ▶ Una forma disciplinada y sistemática de llegar desde las necesidades de los usuarios a una especificación
- ▶ Lo que hay que conocer para definir bien una arquitectura (y por lo tanto para hacer bien un sistema):

Requerimientos
Funcionales
“de negocio”

Otros Atributos
de Calidad
requeridos

Restricciones

Distintas clasificaciones de atributos de calidad

‣ **IEEE Std 1061 / ISO 9126**

- Efficiency
- Functionality
- Maintainability
- Portability
- Reliability
- Usability

‣ **Mitre**

- Efficiency
- Reliability
- Usability
- Maintainability
- Expandability
- Interoperability

- Reusability
- Integrity
- Survivability
- Correctness
- Verifiability
- Flexibility
- Portability

Atributos de calidad: Apertura en ISO 9126

Características

Subcaracterísticas

Funcionalidad

Adecuación

Corrección

Interoperabilidad

Seguridad

Conformidad

Fiabilidad (reliability)

Madurez

Tolerancia a Fallas

Recuperabilidad

Usabilidad

Aprendibilidad

Comprensibilidad

Operabilidad

Atractividad

Eficiencia

Comportamiento Temporal

Utilización de Recursos

Mantenibilidad

Analizabilidad

Cambiabilidad

Estabilidad

Facilidad de Prueba

Portabilidad

Adaptabilidad

Instalabilidad

Coexistencia

Reemplazabilidad

Atributos de calidad que vamos a discutir

- ▶ Relacionados con la calidad del sistema:
 - ▶ Disponibilidad
 - ▶ Facilidad de cambios
 - ▶ Performance
 - ▶ Escalabilidad
 - ▶ Seguridad
 - ▶ Facilidad de testeo
 - ▶ Usabilidad

Atributos de calidad – Disponibilidad

- ▶ Relacionada con fallas (“failures”) en el sistema y sus consecuencias asociadas.
- ▶ Un “failure” ocurre cuando un sistema no entrega más un servicio de acuerdo con su especificación.
- ▶ Esas “failures” son observables por los usuarios (personas u otros sistemas).
- ▶ Error <> Defecto (defect) <> Fault <> Failure
- ▶ Tiempo de reparación = tiempo hasta que la falla no es más observable
- ▶ Disponibilidad = probabilidad de que un sistema esté disponible cuando se lo necesite
 - ▶ $D = \text{Mean Time to Failure} / (\text{Mean Time to Failure} + \text{Mean Time to Repair})$
- ▶ Los “downtimes” programados no se consideran
- ▶ Relativamente fácil de especificar, difícil de verificar

Atributos de calidad – Facilidad de cambios

- ▶ Relacionada con el costo de los cambios. Uno de los atributos de calidad más difíciles de expresar.
- ▶ Temas importantes:
 - ▶ ¿Qué puede cambiar?
 - ▶ Funcionalidad
 - ▶ Plataforma
 - ▶ Otros atributos de calidad
 - ▶ Interfaces
- ▶ ¿Quién y dónde se hace el cambio?
 - ▶ Usuarios, desarrolladores, administradores
 - ▶ Código, configuración, parametrización
- ▶ Una vez que un cambio se especifica, debe ser diseñado, implementado, probado y liberado. Todo esto cuesta dinero.

Escuchando a los que saben...



David Parnas: "The part of the game of designing good software is designing for change"

Performance

- ▶ Relacionada con el tiempo que le lleva al sistema responder a un evento que ocurre (interrupciones, mensajes, pedidos de usuarios o paso del tiempo).
 - ▶ Latencia: tiempo entre la llegada del estímulo y el inicio de la respuesta del sistema
 - ▶ “Jitter”: variación en la latencia
 - ▶ Deadlines: límites de tiempo para un proceso
 - ▶ Throughput: cantidad de transacciones que el sistema puede procesar en un período de tiempo
 - ▶ Eventos no procesados
- ▶ Difícil de expresar. Depende de volúmenes del sistema, equipamiento en uso y versiones de sistema operativo y otros software de base.

Seguridad

- ▶ Habilidad de un sistema para resistir usos no autorizados y seguir proveyendo sus servicios a usuarios legítimos. Incluye:
 - ▶ Nonrepudiation: mecanismos para asegurar que quienes hicieron algo no puedan negarlo
 - ▶ Condifencialidad: propiedad por la cual datos o servicios son protegidos de accesos no autorizados
 - ▶ Integridad: propiedad por la cual datos o servicios se brindan como fue previsto.
 - ▶ Disponibilidad (en el contexto de seguridad): que un sistema esté disponible para su uso legítimo
 - ▶ Auditabilidad: habilidad de un sistema para hacer un seguimiento de actividades realizadas

Facilidad de testeo

- ▶ Facilidad que presenta un sistema para que se ejecuten sobre él actividades de testing.
- ▶ Para eso se debe poder controlar el estado interno de un componente y poder ver sus outputs. Normalmente se utilizan los llamados “test harness” (hay herramientas comerciales que los implementan).

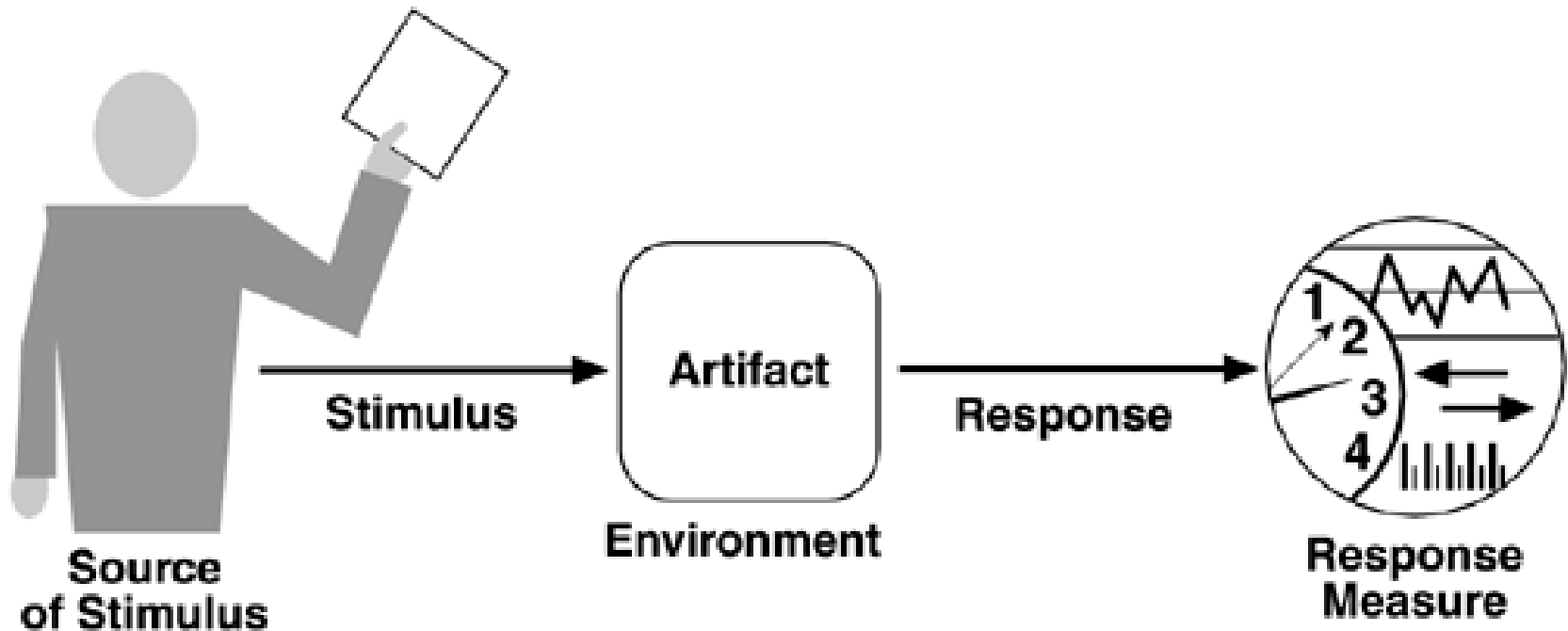
Usabilidad

- ▶ Usabilidad: Relacionada con la facilidad con la cual un usuario puede cumplir una tarea o utilizar un servicio ofrecido por el sistema y el tipo de soporte que provee el sistema.
 - ▶ Aprender la funcionalidad del sistema
 - ▶ Usar el sistema eficientemente
 - ▶ Minimizar el impacto de los errores
 - ▶ Adaptar el sistema a las necesidades de los usuarios
 - ▶ Aumentar confianza y satisfacción

Otros Atributos

- ▶ Escalabilidad: una medida de qué tan bien una solución sigue cumpliendo con sus requerimientos al cambiar los volúmenes del problema que resuelve
- ▶ Portabilidad: facilidad de un sistema para poder ser operado en distintas plataformas.

Especificación de Atributos de Calidad (SEI)



Quality Attribute Scenario, formado por:

Fuente del estímulo: Interna o externa

Estímulo: condición que debe ser tenida en cuenta al llegar al sistema

Entorno: condiciones en las cuales ocurre el estímulo

Artifact: el sistema o partes de él afectadas por el estímulo

Response: qué hace el sistema ante la llegada del estímulo

Response measure: cuantificación de un atributo de la respuesta

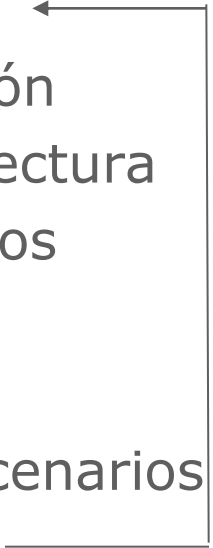
Ejemplos de escenarios de atributos de calidad

- ▶ Escenario de disponibilidad
 - ▶ “Un proceso del sistema recibe un mensaje externo no anticipado durante un modo de operación normal. El proceso informa al operador y continúa su operación sin caídas”.
 - ▶ Fuente: Sistema externo
 - ▶ Estímulo: Mensaje no anticipado
 - ▶ Entorno: Operación normal
 - ▶ Artefacto: Proceso interno
 - ▶ Respuesta: Informar al operador y seguir operando
 - ▶ Medición de la respuesta: sin caídas (downtime)

Ejemplos de escenarios de atributos de calidad (cont.)

- Escenario de facilidad de cambios
 - “Un desarrollador desea cambiar el framework de ORM de la aplicación. El cambio será hecho en el código, en tiempo de diseño. El cambio debe poder ser hecho sin efectos secundarios en menos de 500 horas persona”.
 - Fuente: Desarrollador
 - Estímulo: Cambio en el framework de ORM
 - Entorno: En tiempo de diseño
 - Artefacto: Código
 - Respuesta: Cambio hecho sin efectos secundarios
 - Medición de la respuesta: en menos de 500 horas persona

Quality Attribute Workshops

- ▶ El Quality Attribute Workshop (QAW) es un método facilitado que relaciona los stakeholders de un sistema de manera temprana en el ciclo de vida para descubrir los atributos de calidad clave en un sistema de software
 - ▶ Sus pasos son:
 - ▶ Presentación del método
 - ▶ Presentación del negocio / misión
 - ▶ Presentación del plan de arquitectura
 - ▶ Identificar drivers arquitectónicos
 - ▶ Brainstorming de escenarios
 - ▶ Consolidación de escenarios
 - ▶ Definición de prioridades de escenarios
 - ▶ Refinamiento de escenarios
- 
- Iterar mientras sea necesario agregando "stakeholders"

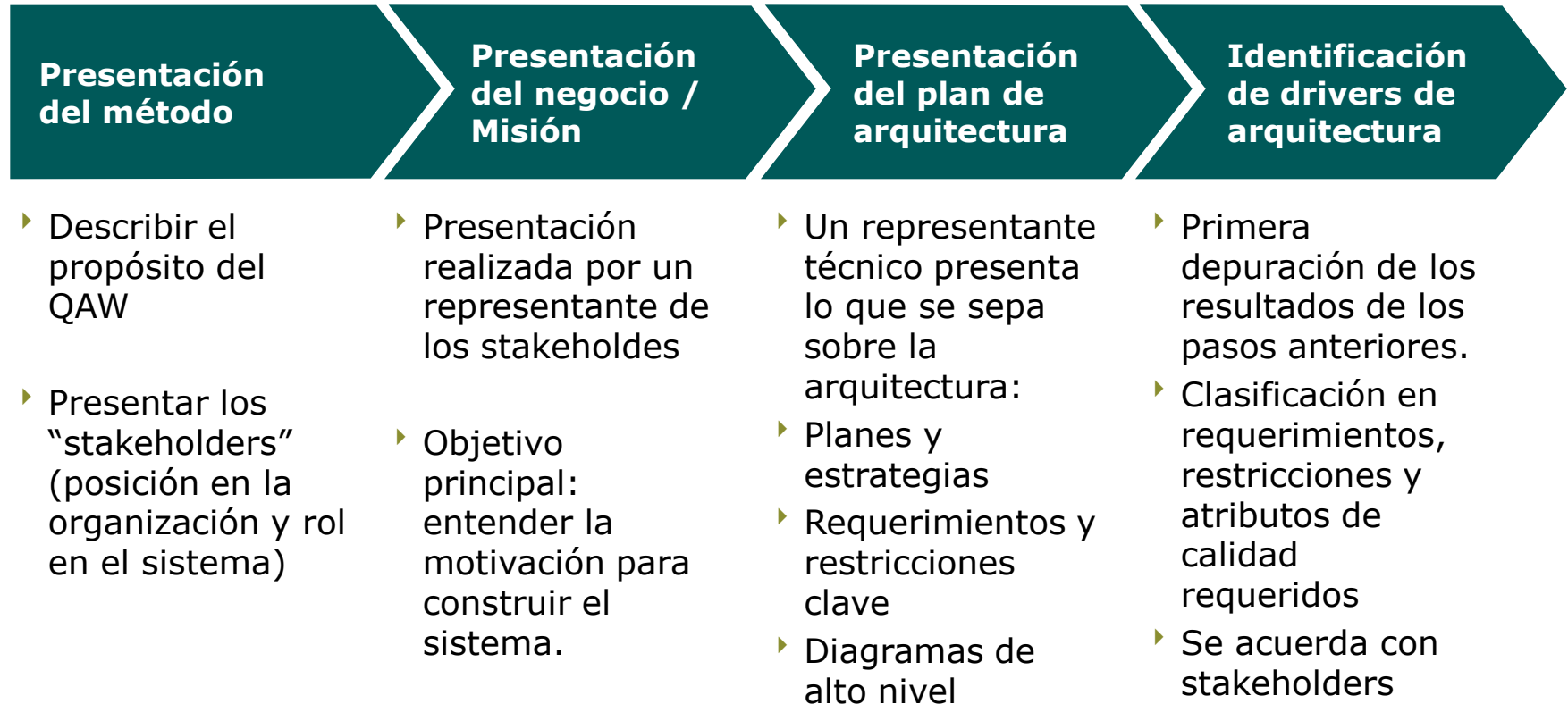
Motivación – ¿Qué problema queremos atacar?

- ▶ ¿Cuál es el significado preciso de los atributos de calidad, como modificabilidad seguridad, performance, y confiabilidad, en el contexto del sistema que se está construyendo?
- ▶ ¿Cómo se pueden descubrir, caracterizar, y dar prioridades a los atributos clave antes de que se construya el sistema?
- ▶ ¿Cómo se puede hacer que una comunidad dispersa geográficamente de “stakeholders” se involucren de una manera disciplinada y repetible en el descubrimiento y la caracterización de atributos de calidad?
- ▶ ¿Cómo se puede usar toda esta información?

Roles y duración

- ▶ Evento de 1 día ofrecido como servicio por el SEI
- ▶ Roles:
 - ▶ Líder de QAW: facilitador de las discusiones
 - ▶ Asistente: registra la información de las sesiones
- ▶ Existe un template del SEI para minutas de las sesiones

Detalle del proceso (1)



Detalle del proceso (2)

