

Departamento de Ingeniería de
Sistemas y Computación

Arquitectura de Software

Arquitectura y Procesos de Desarrollo

CSOF 5204



Agenda

1. Introducción

2. Metodologías

3. Arquitectura y Procesos de Desarrollo

Agenda



1. Introducción

2. Metodologías

3. Arquitectura y Procesos de Desarrollo

Introducción

¿Qué hace un Arquitecto de Software?

¿Qué es Arquitectura de Software?



Introducción (II)

¿Qué hace un Arquitecto de Software?

(Tomado de www.monster.com)

- Design enterprise Web applications using MVC framework, Hibernate and Service Oriented Architecture (SOA) principles. Responsible for functional and technical designs of cross-platform applications & services. Perform business analysis and requirement analysis with the end user. The candidate must demonstrate progressive leadership in the full life cycle software development environment.
- Architect enterprise applications using Agile Methodology. This includes creating detailed design using UML, process flows, sequence diagrams, and some times pseudo-code level details.
- Leverage enterprise service layer framework SOA for the architecture. Create designs involving XML/Web-Services that integrate various distributed applications.

Introducción (III)

¿Qué hace un Arquitecto de Software?

(Tomado de www.monster.com)

- Perform business analysis and requirement analysis with the help of end-user in the Cable/Telecom BSS/OSS environment.
- Develop prototype and architecture for new concepts and technology that would eventually be used by the development teams.
- Develop functional designs including Use Cases while interacting with *business partners* as well as *end customers*.
- Develop designs of the User Interfaces (UI) especially Java based UI using struts and JSPs. Create conceptual data model for the database interaction.
- Closely work with Enterprise Architects, Developers, and Business Partners while developing design solutions.
- Perform other duties as assigned.

Introducción

¿Qué hace un Arquitecto de Software?

- Ingeniería de Requerimientos
- Maneja equipos de trabajo
- Es un líder técnico
- Toma decisiones de diseño
- Evalúa la arquitectura
- Guía la implementación
- Administra la evolución del sistema/producto
- Punto de contacto con el negocio

Introducción

“Architecting is about Balancing”

Randy Stafford



Introducción

Habilidades de un Arquitecto

- Conoce el dominio del problema
- Gerencia técnica y de desarrollo
- Habilidades en diseño e implementación
- Habilidades de comunicación oral y escrita
- Negociación
- Liderazgo

Introducción

¿Qué es Arquitectura de Software?

“The software architecture of a program or computing system is the **structure or structures** of the system, which comprise the **software elements**, the externally **visible properties** of those elements, and the **relationships** among them.”

Len Bass

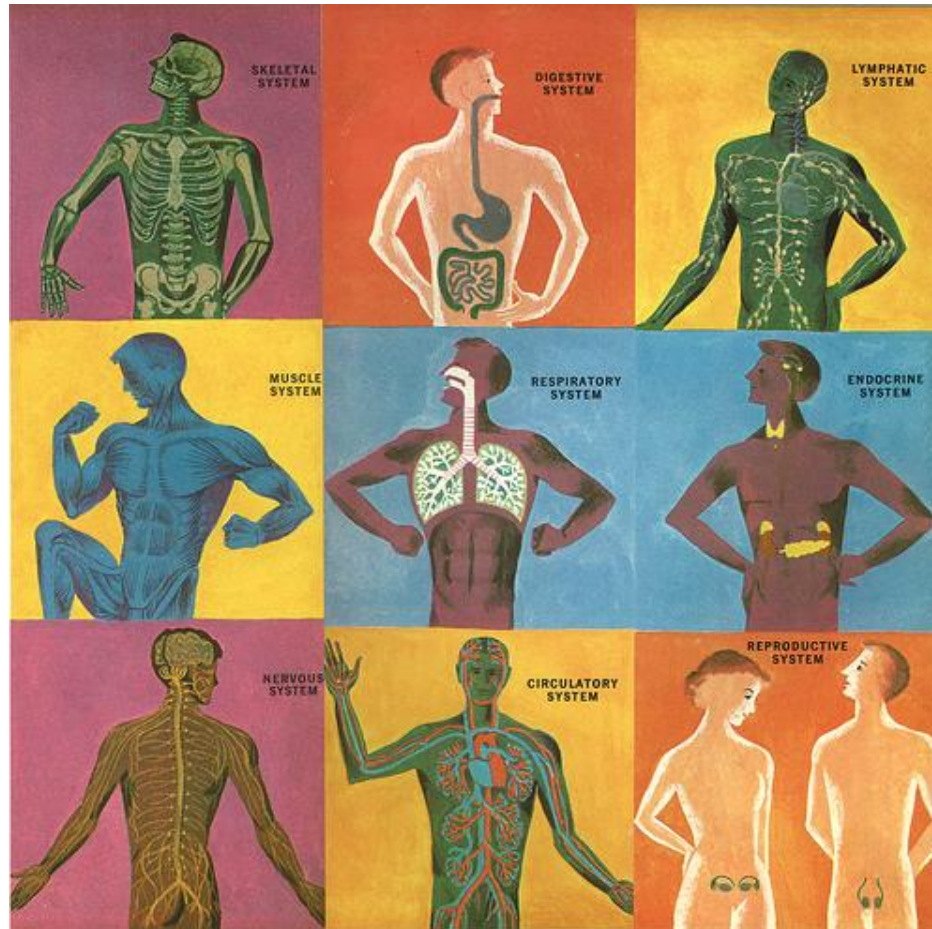
Introducción

¿Qué es Arquitectura de Software?

¿Es esto una arquitectura?



Introducción



¿Cómo hacer las propiedades visibles y describir las relaciones entre elementos de software?

Introducción

¿Qué es Arquitectura de Software?

“Effective Software Architects understand not only **technology** but also the **business domain** of a problem space. Without business domain knowledge, it is difficult to understand the business problem, goals, and requirements, and therefore difficult to **design an effective architecture** to meet the requirements of the business”

Mark Richards

Introducción

¿Qué es Arquitectura de Software?

“In the context of business enterprise applications development, an architect must act as a bridge between the **business** and **technology** communities of an organization, **representing the interest** of each party to the other, often mediating between the two, but allowing the business to drive.”

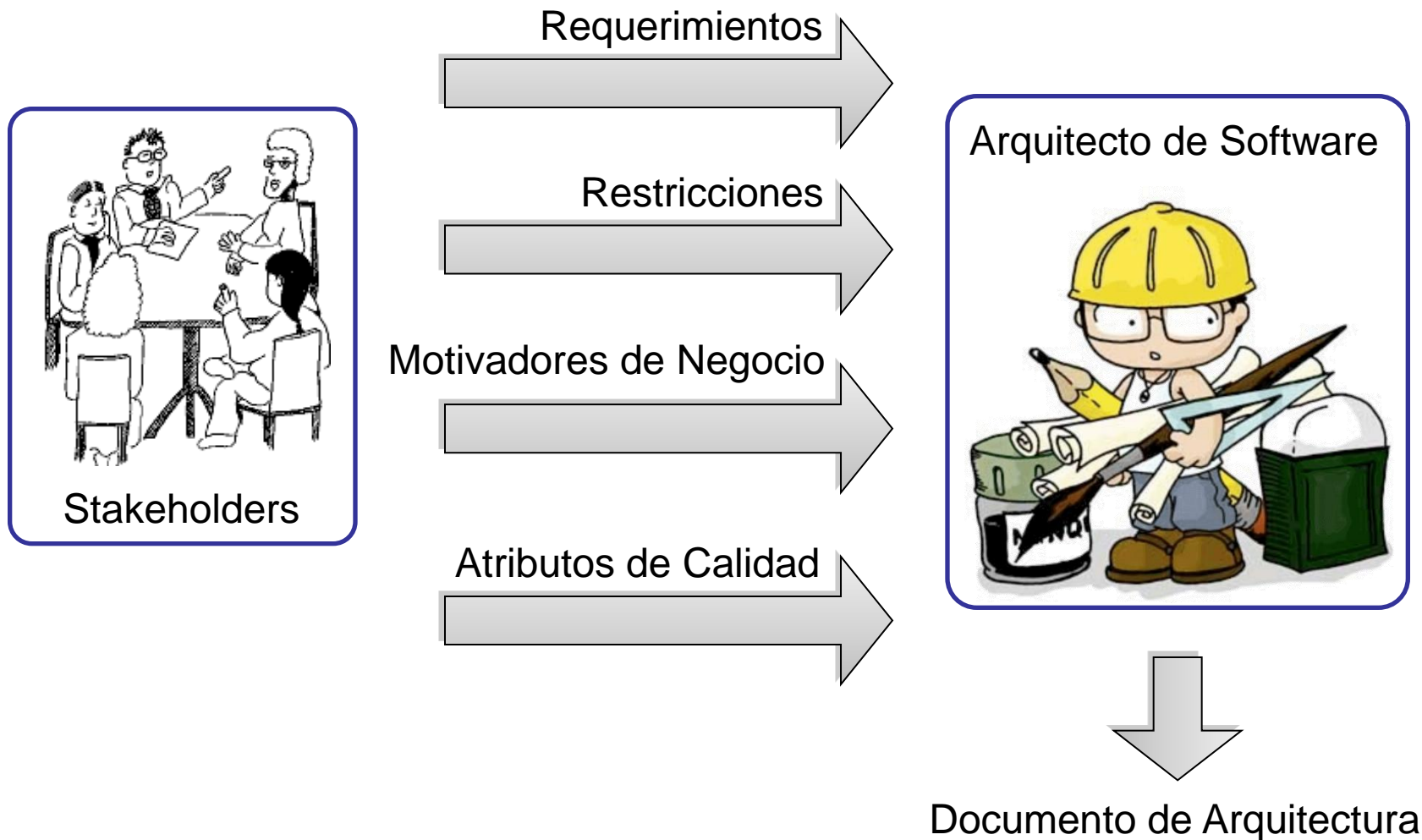
Dave Muirhead

Introducción

La arquitectura está directamente relacionada con la organización

- ¿Cómo la impacta el contexto de negocio?
 - Presupuestos, tiempos de mercado, estrategias de negocio, visión, tecnologías, ...
- ¿De dónde provienen tantos factores?
 - Diferentes stakeholders tienen diferentes intereses y preocupaciones
- Y viceversa ¿Cómo la arquitectura impacta la organización?
 - Recursos?
 - Visión del negocio?
 - Infraestructura?

Introducción



Agenda

1. Introducción



2. Metodologías

3. Arquitectura y Procesos de Desarrollo

Metodologías

- ADD (Attribute Driven Design)
- ACDM (Architecture Centric Design Method)

Metodologías

- ADD (Attribute Driven Design)
 - Toma como entrada escenarios de calidad
 - Se basa en un proceso de descomposición recursivo
 - Como salida se obtiene una descomposición modular

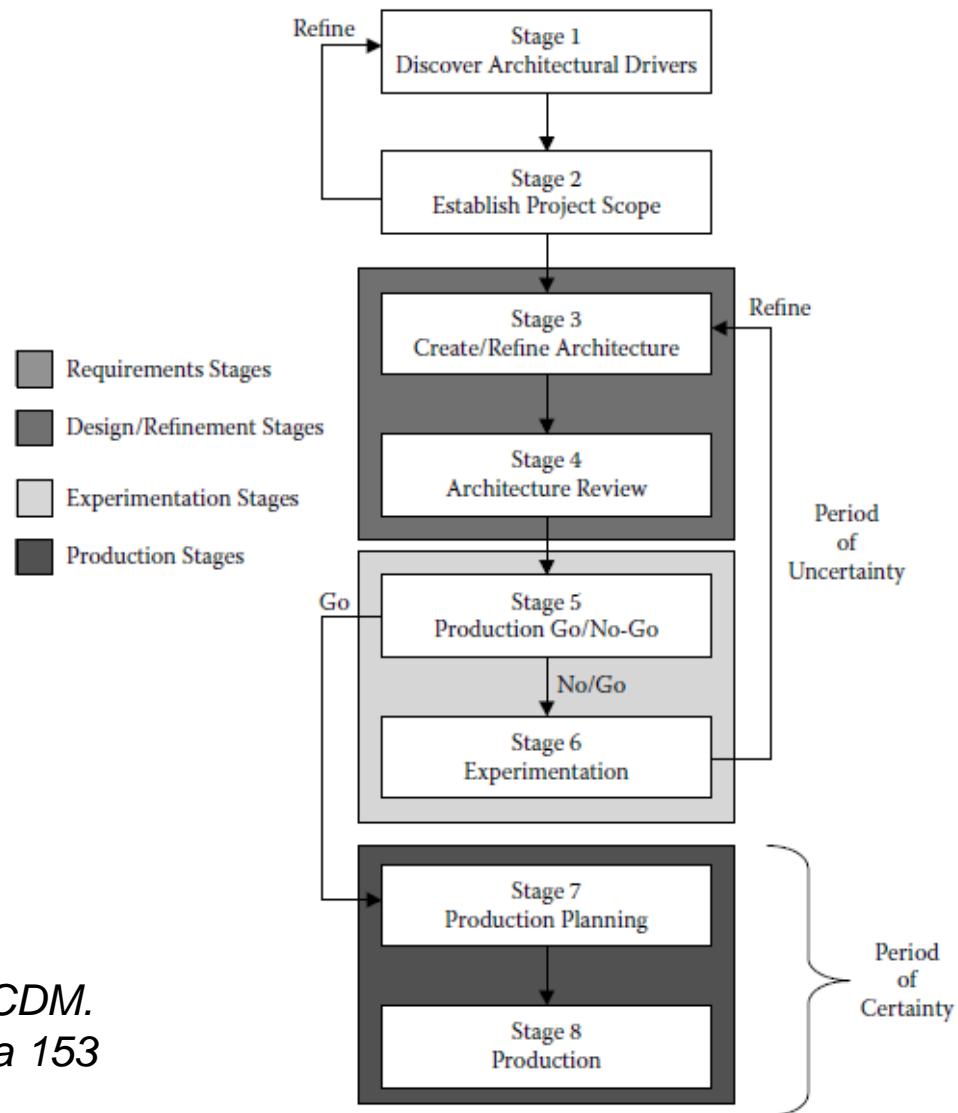
Metodologías

- Pasos ADD (Attribute Driven Design)
 - Seleccionar el módulo a descomponer
 - Refinar el módulo
 - Seleccionar motivadores arquitectónicos relevantes
 - Seleccionar estilos arquitectónicos
 - Asignar funcionalidad
 - Definir interfaces
 - Verificar los resultados

Metodologías

- ACDM (Architecture Centric Design Method)
 - Propuesto por A. Lattanze – CMU
 - Utilizado para estimar, planificar y seguir la construcción del producto

- ACDM (Architecture Centric Design Method)



*Etapas del método ACDM.
Tomado de [2], página 153*

Agenda

1. Introducción

2. Metodologías



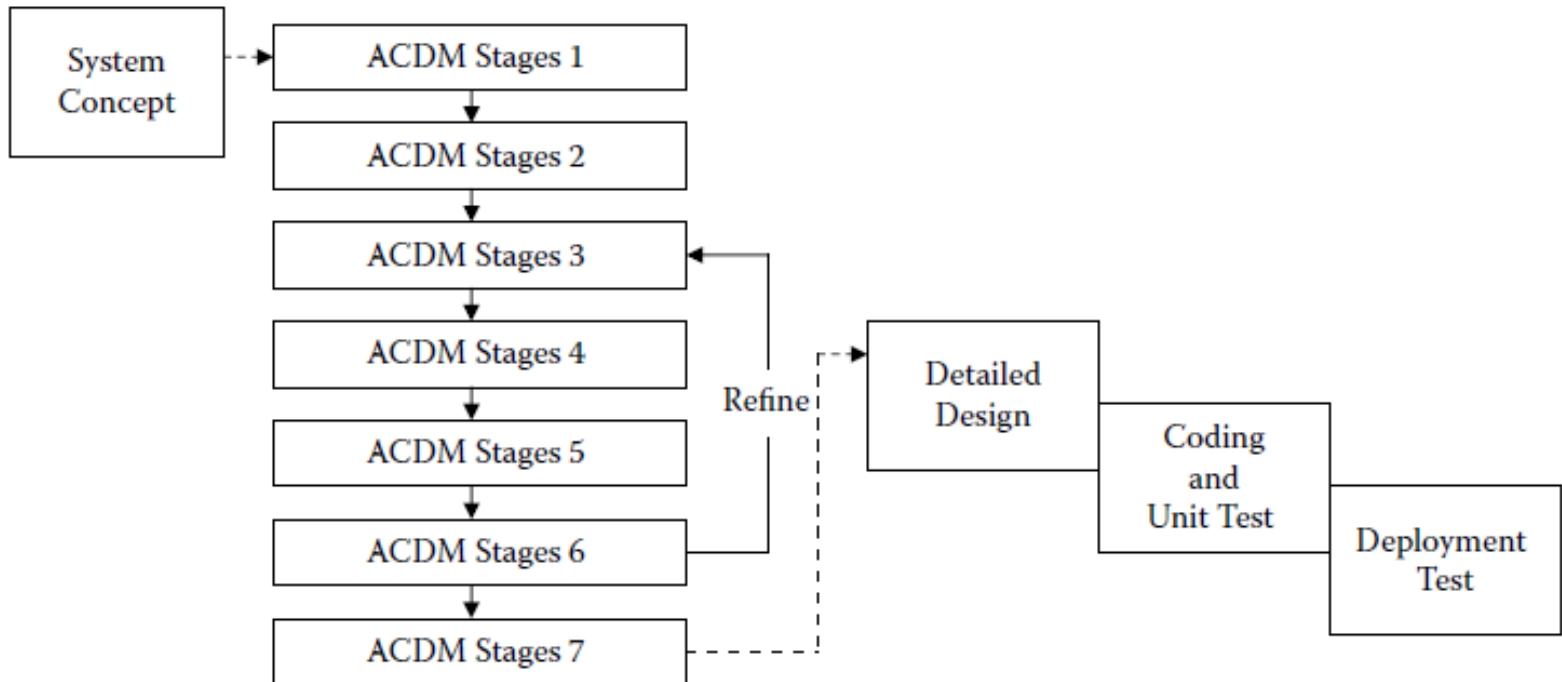
3. Arquitectura y Procesos de Desarrollo

Procesos de Desarrollo

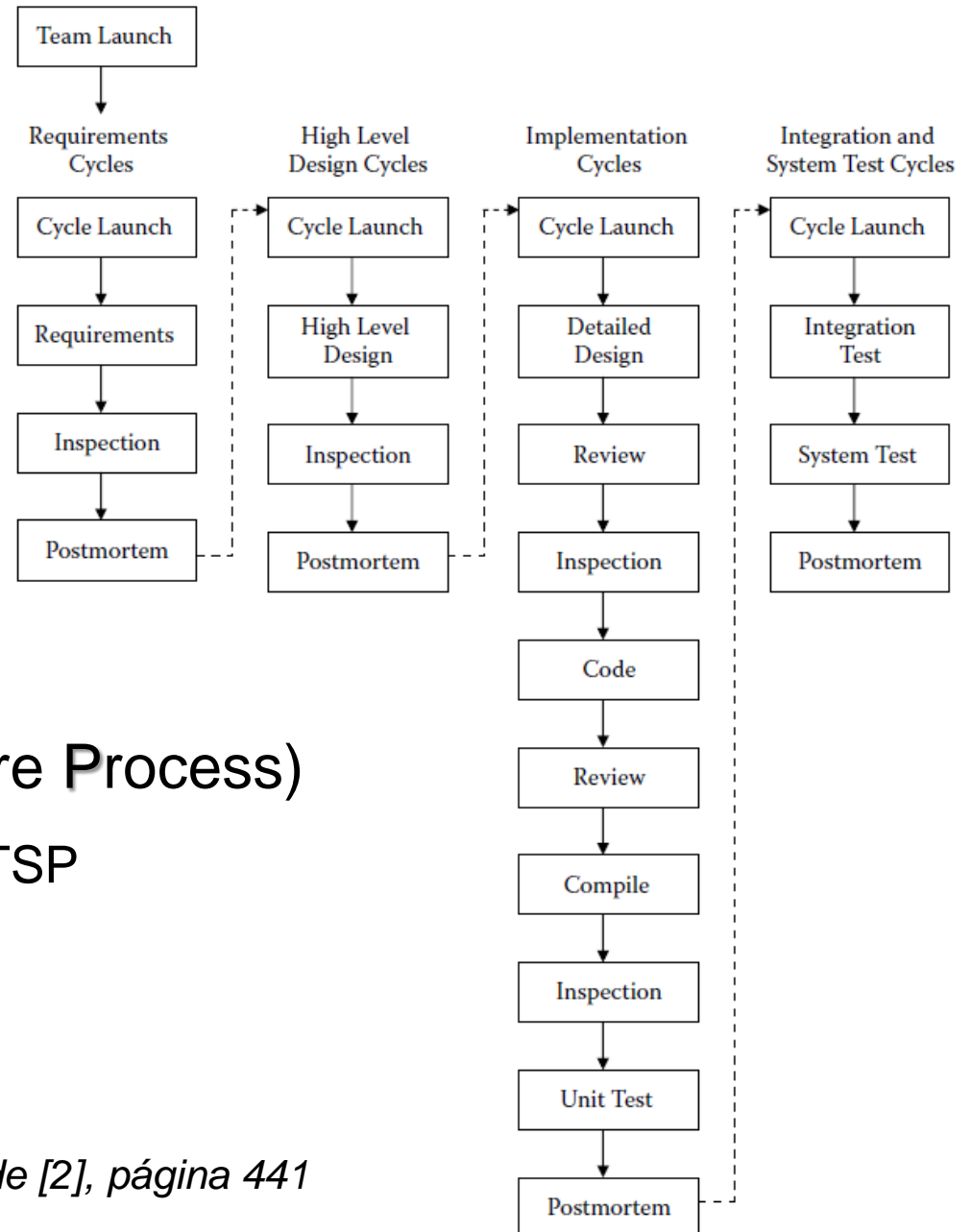
- Cascada
- TSP (Team Software Process)
- RUP (Rational Unified Process)
- Metodologías Ágiles

Procesos de Desarrollo

- Cascada



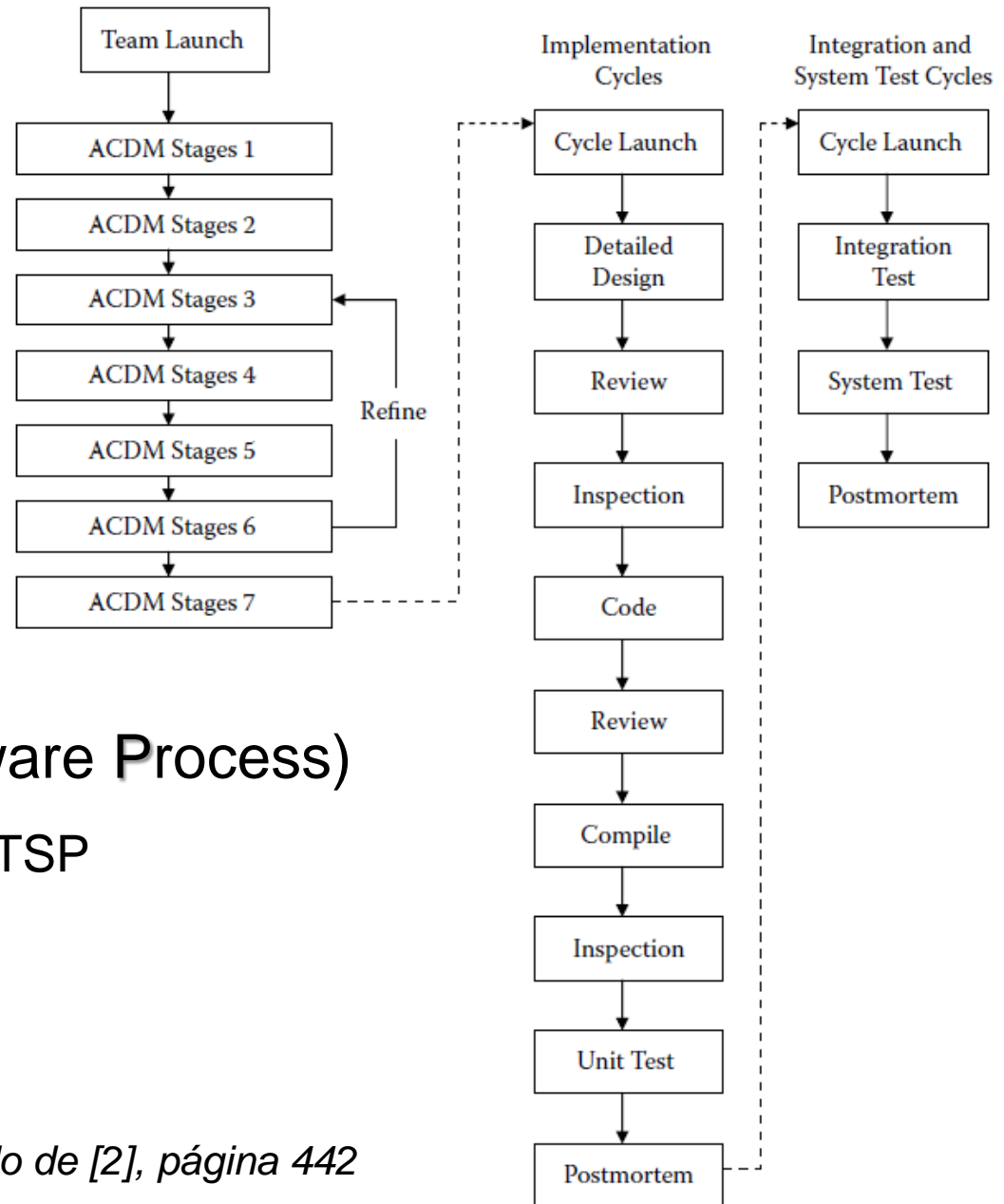
Tomado de [2], página 432



- **TSP (Team Software Process)**

Estructura general de TSP

Tomado de [2], página 441



- **TSP (Team Software Process)**

Integrando ACDM y TSP

Tomado de [2], página 442

Procesos de Desarrollo

- RUP (Rational Unified Process)

RUP	ACDM
Inception	Etapas 1 – 2
Elaboration	Etapas 3 – 7
Construction	Etapas 8
Transition	N / A

Procesos de Desarrollo

- Metodologías Ágiles
 - Diversidad de Opiniones
 - No existe en general un acuerdo sobre su integración

Referencias

- [1] Bass, L., Clements, P., Kazman, R. “Software Architecture in Practice”, Addison-Wesley, Second Edition, 2006.
- [2] Lattanze, A. “Architecting Software Intensive Systems: A practitioner's guide”, CRC Press, 2009.

¿Preguntas?

