

Estrategia de Gestión de Proyectos Dirigida por la Arquitectura

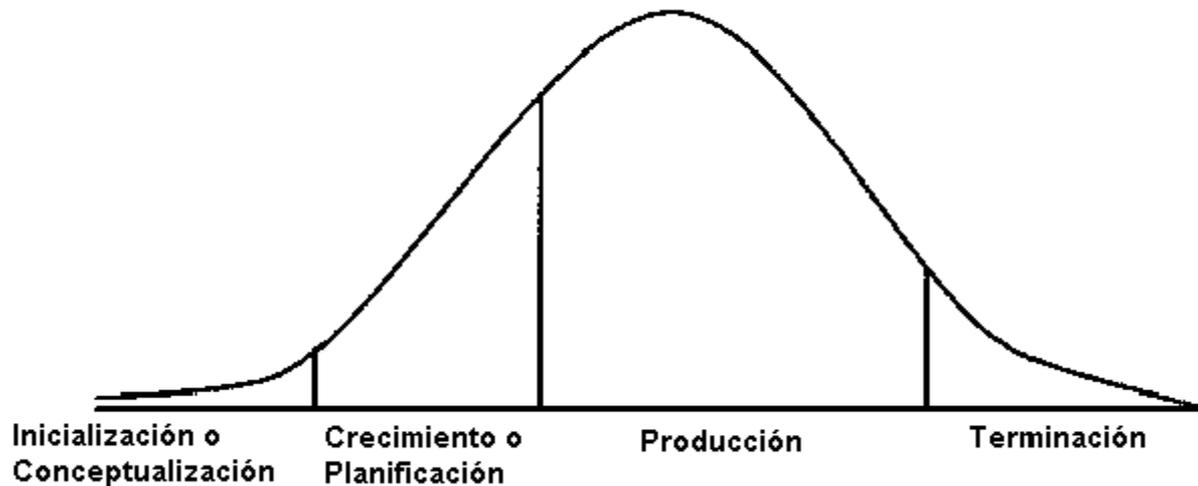
Ing. Mariel Feder, MC, PMP

Qué es la gestión de un proyecto?

- Gestionar un proyecto consiste en asegurar que se entregará un producto que cumpla con las especificaciones dentro del plazo y costo acordado, con el nivel de calidad pactado
 - Entorno de Naturaleza Cambiante
 - Incertidumbre (RRHH, tecnológicos, negocio)

Ciclo de vida del proyecto

Ciclo de Vida de un Proyecto



Inicialización

- Análisis de consecuencias
- Definición de magnitudes
- Análisis de alternativas
- Análisis técnicos (viabilidad, mat, rec, etc)
- Análisis de inversión y rentabilidad
- Análisis de riesgo

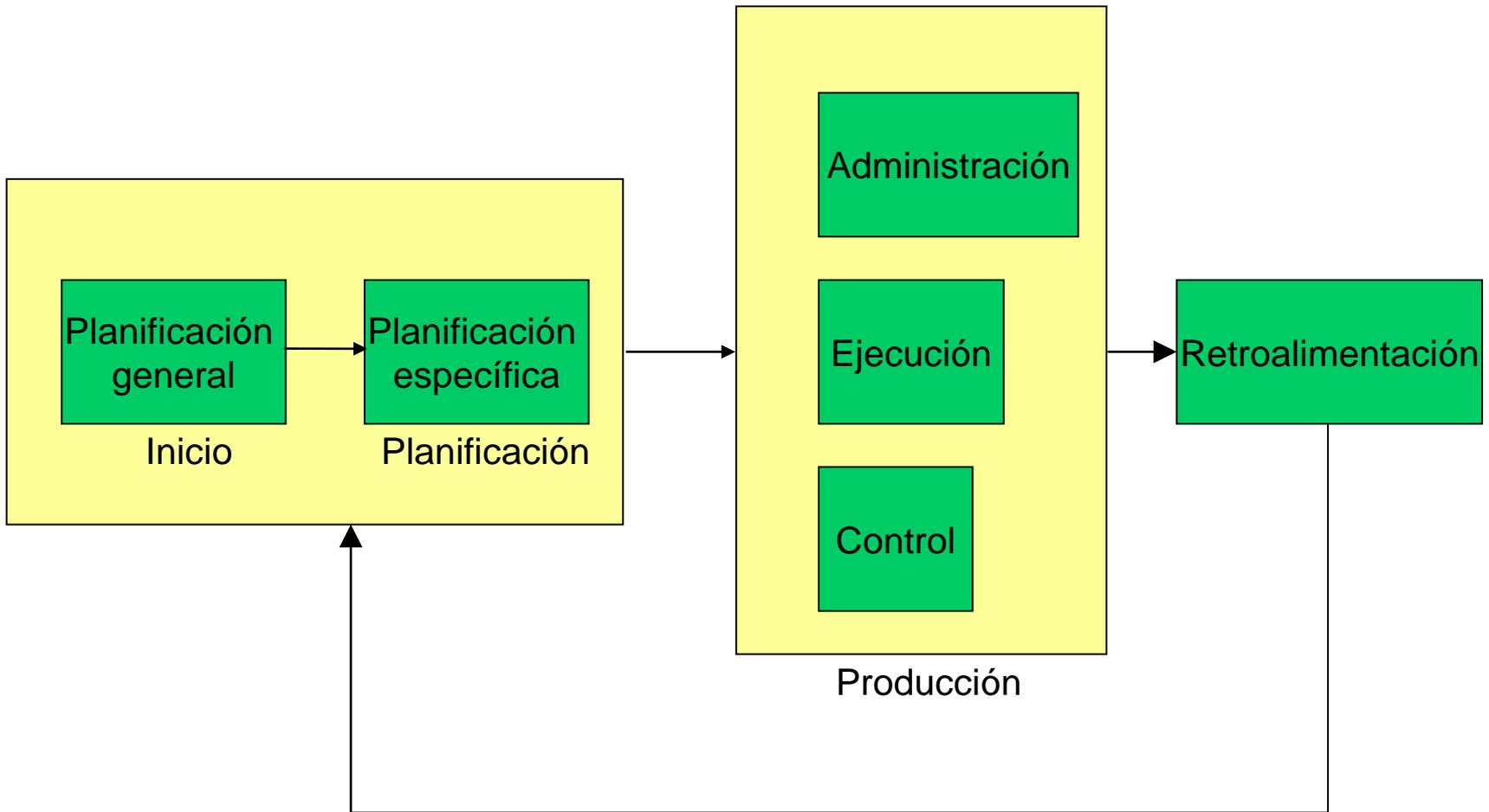
Planificación

- Esquema de trabajo para alcanzar el objetivo
- Estimaciones
- Se itera durante el proyecto

Producción o Implementación

- Administración
- Ejecución
- Control

Cierre



Etapas en la gestión de un proyecto

Arquitectura de Software

- Disciplina joven
- Bass, Clements y Kazman: ***La arquitectura de un programa o sistema de computación es la estructura o estructuras del sistema, que comprenden sus componentes de software, las propiedades externas de los componentes, y la relación entre ellos***, que coincide con la propuesta por Soni, Nord y Hofmeister

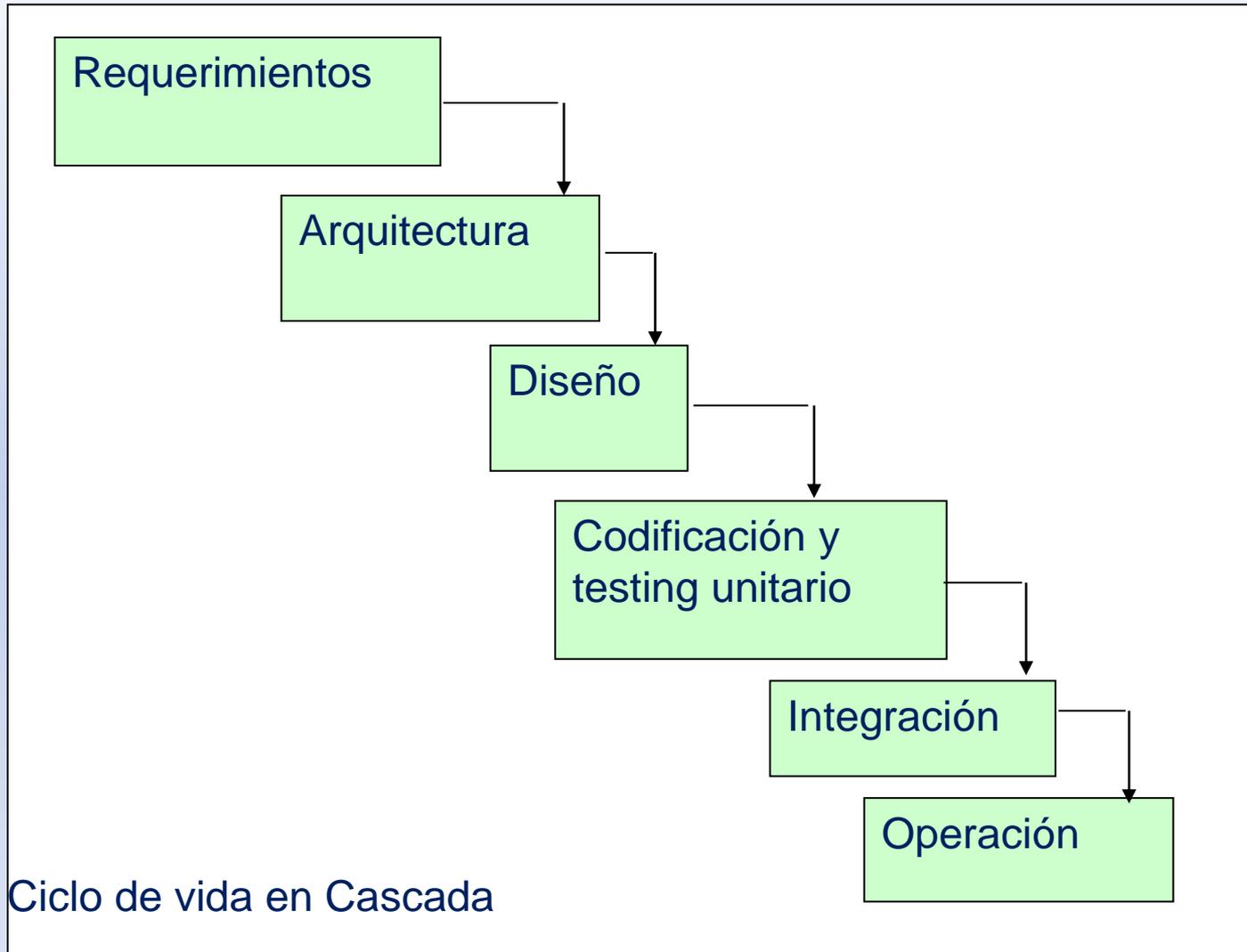
Ciclo de vida del Desarrollo del Producto

- Orden de las actividades que ocurren durante el desarrollo

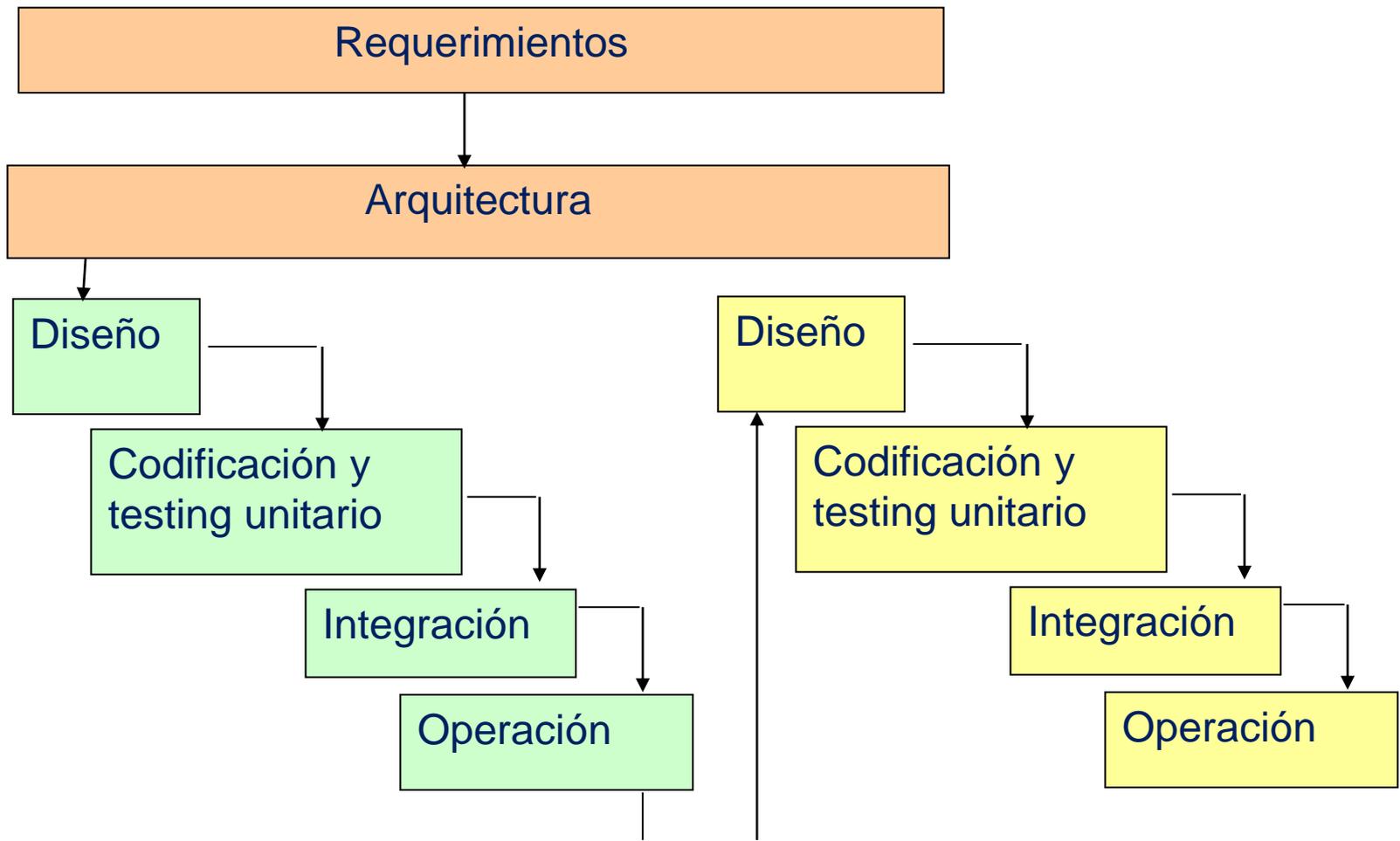
Cascada (Royce, 1970)

- Secuencia de fases que no se repite:
 - Planificar el proyecto antes de comenzar
 - Definir el comportamiento externo esperado del sistema antes de definir su funcionamiento interno.
 - Documentar los resultados de cada actividad
 - Diseñar el sistema antes de codificarlo
 - Testear el sistema una vez que está construido.

Cascada

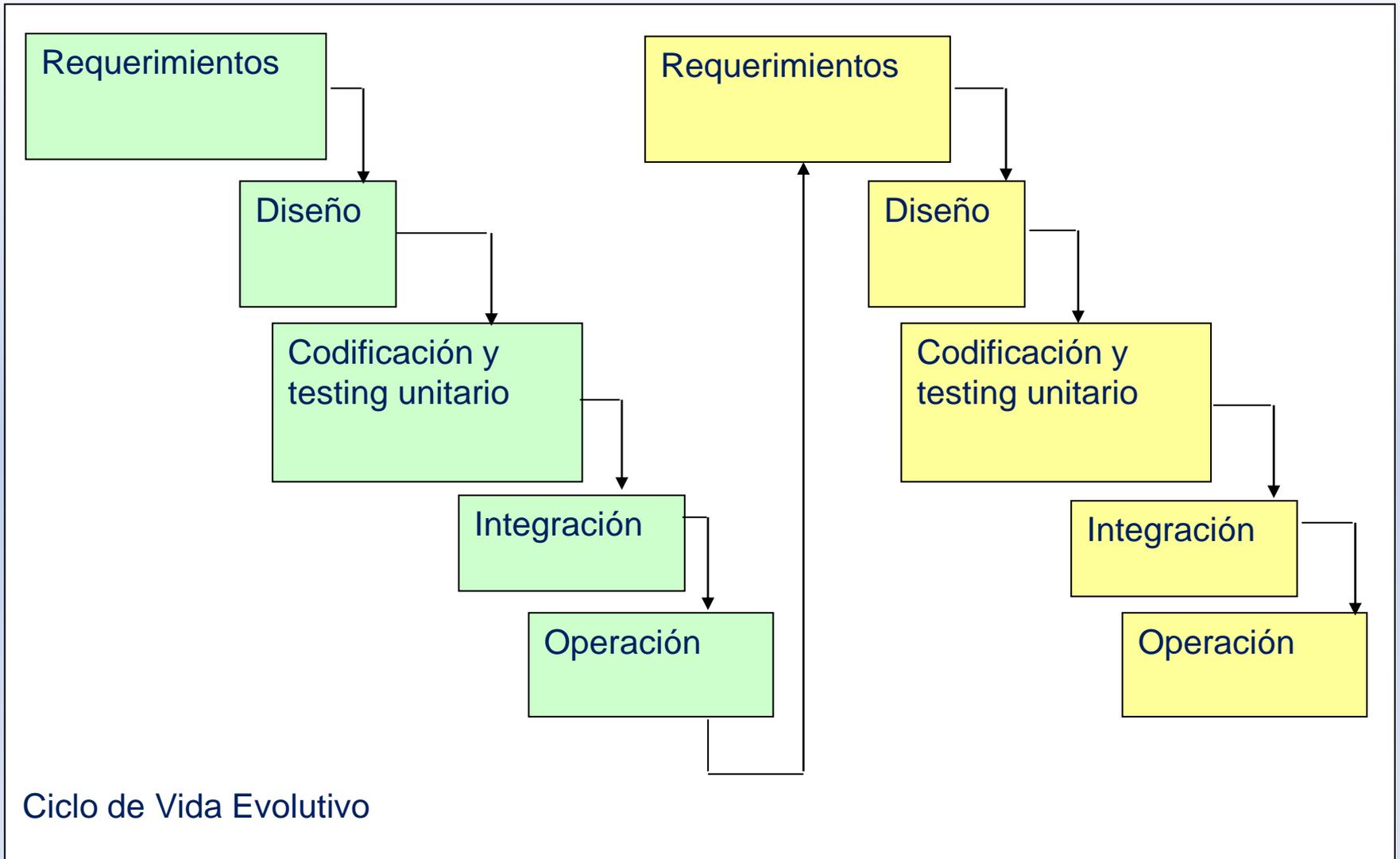


Desarrollo Incremental (Hirsch, 1985)



Ciclo de vida de desarrollo incremental

Evolutivo (Basili - Turner, 1975)



Otros

- Espiral (Boehm 1988)
- Prototipación
- Etc

Proceso para Gestión de Proyectos dirigidos por la arquitectura

Ciclo de vida para PDA

- Autores como Paulish, importancia a Arquitectura previa.
- Necesaria para PDA
- Ciclo cascada/incremental

Planificación del Proyecto

- Proyectos bien gestionados comienzan como proyectos bien planificados.
- Cuándo: en paralelo con la definición de la arquitectura.
- Presión externa. Problema de estimaciones tempranas.
- Cronograma viable, metas intermedias.

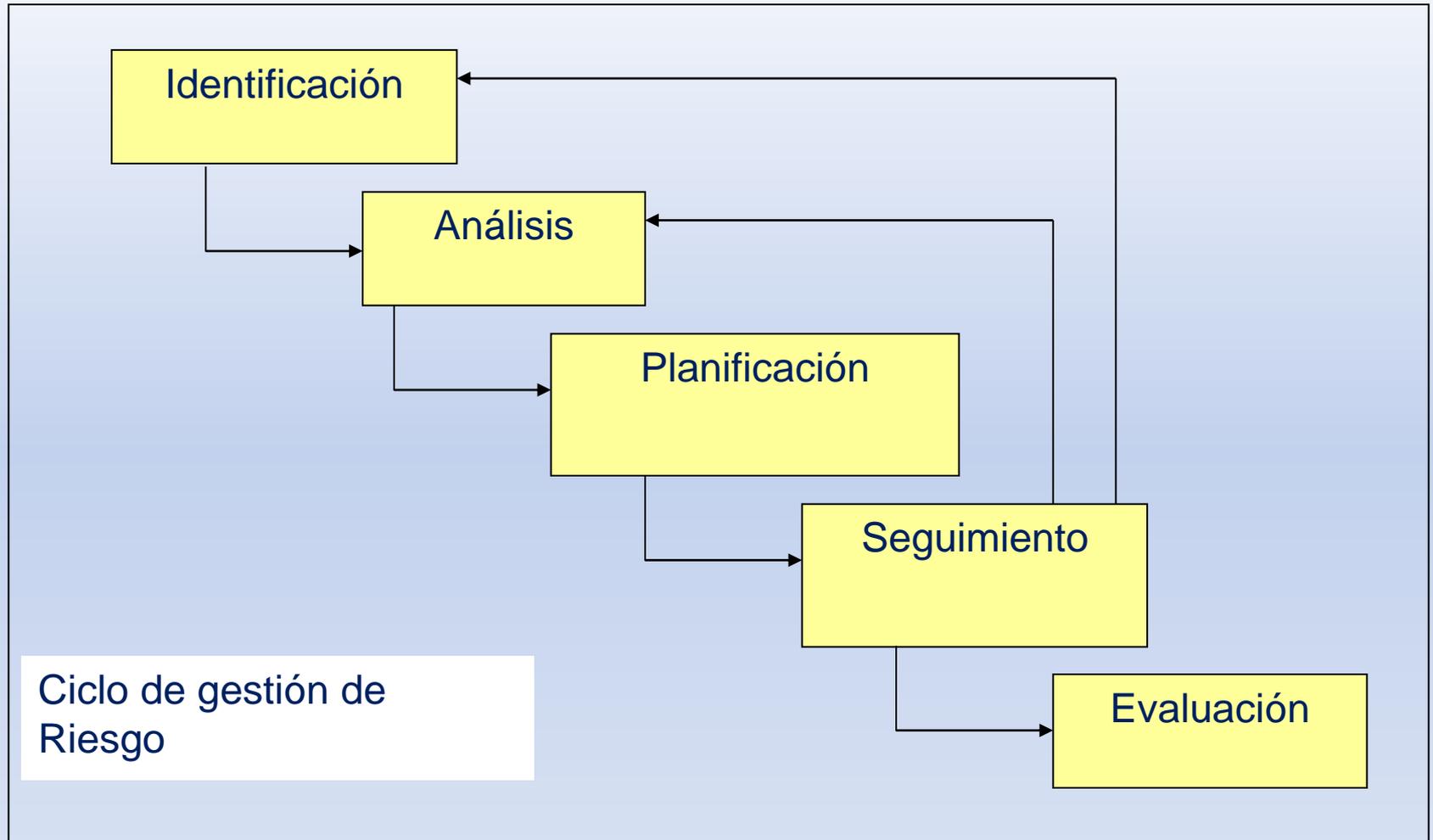
Estimación

- Pequeño grupo arq. de alto nivel
- Mecanismos de estimación top-down (Cocomo, Puntos Funcion, Slim, Price-S)
- Basarse en componentes para estimación Bottom Up (vista lógica de Krutchen).
Responsabilidad del arquitecto
- Integración de ambas estimaciones.

Top-Down vs Bottom Up

- Bottom up: mayor precision en cada componente
- Top down: incluye otras tareas
- Integración entre ambas
- Cota mínima: Bottom up

Análisis de Riesgo



Riesgos y Arquitectura

- Arquitectura como entrada importante
- Riesgos técnicos

Riesgos y Arquitectura – Check list

- Experiencia de los equipos de desarrollo en las plataformas, lenguajes y herramientas seleccionadas
- Estabilidad de las herramientas y plataformas
- Disponibilidad de las herramientas y plataformas
- Integrabilidad de las plataformas seleccionadas
- Interoperabilidad entre los componentes diseñados

Riesgos y Arquitectura – Check list

- Nivel de complejidad, testeabilidad y acoplamiento de los componentes
- Identificación de los componentes críticos ya sea por su importancia para el funcionamiento del producto o por su complejidad.
- Atributos de calidad requeridos que resulten críticos para el producto (ej.: performance, seguridad, amigabilidad, etc.) y que la arquitectura debe garantizar

Plan de Versiones

- Desarrollo simultáneo?
- Desarrollos en paralelo y en secuencia
- Identificación de paquetes o componentes y precedencias
- Puntos de control, plan de versiones
- Arquitectura como input para este proceso

Criterios para orden y contenido de las versiones

- Dirigido por el riesgo
- Dirigido por las necesidades del usuario/mercado
- Secuencia lógica
- Dirigido por la capacitación
- Combinación

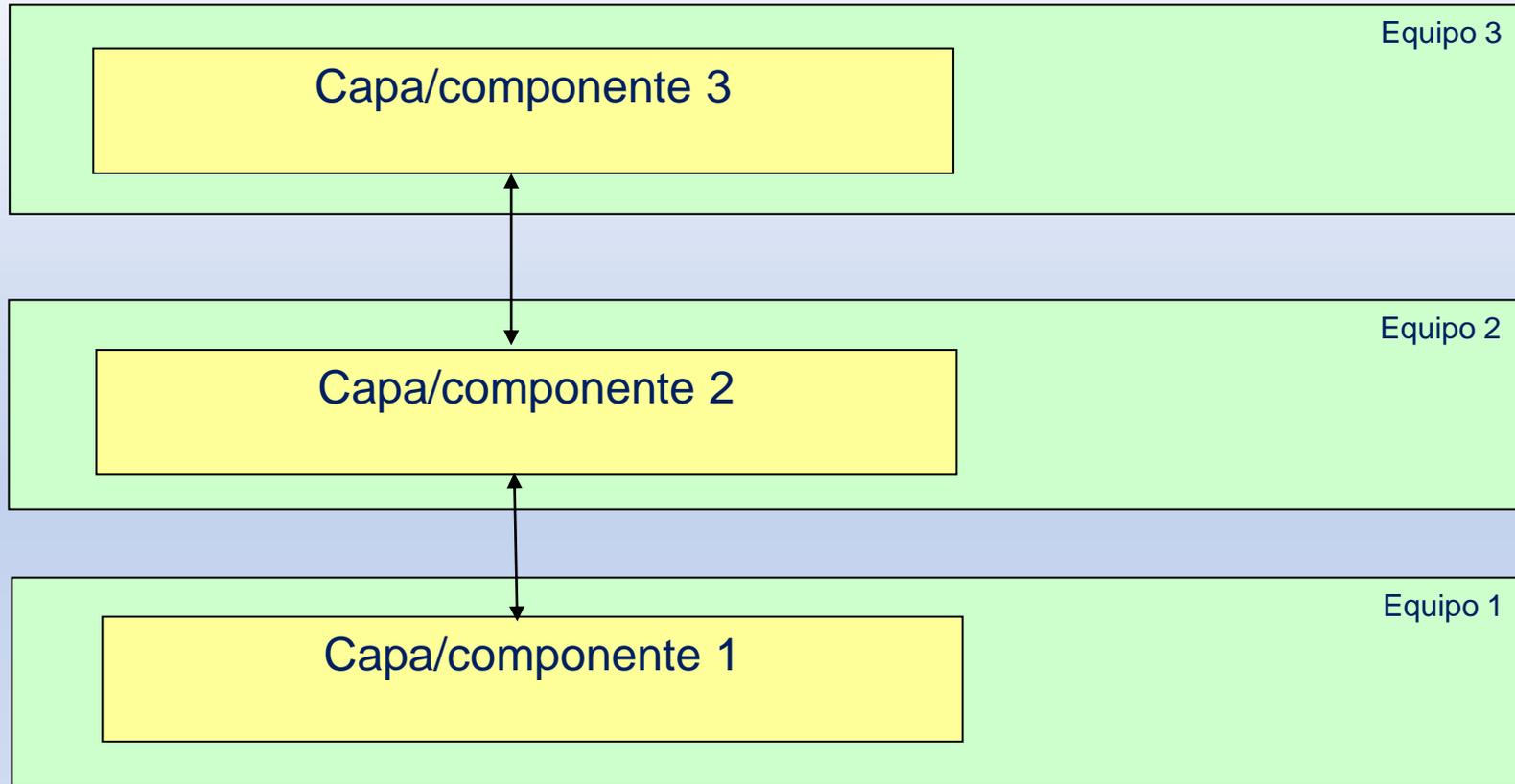
Arquitectura

- Identificación de componentes o paquetes por funcionalidad
- Complejidad y criticidad de los componentes (entrada también para la estrategia de testing).

Recursos Humanos

- Reclutamiento: Perfiles necesarios
- Organización de los equipos: Vertical, Horizontal

Organización de equipos horizontal

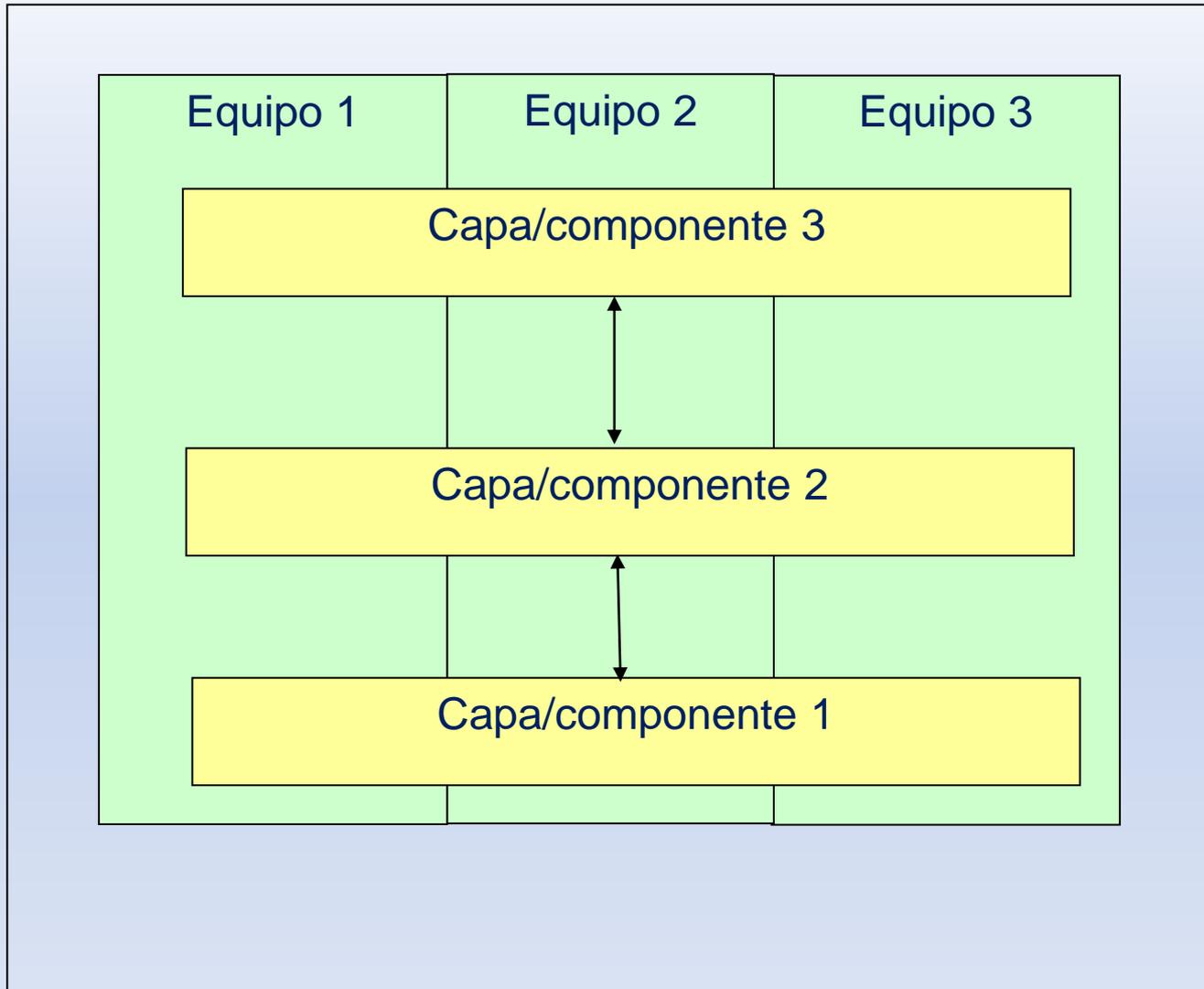


Organización de equipos horizontal

Org. Equipos horizontal

- Expertos en el componente
- Uniformidad en el componente
- Equipos simultáneos
- Visibilidad parcial

Org. Equipos Vertical



Org. Equipos vertical

- Posibilidad de un solo equipo (serialización)
- Evitar problemas de comunicación.
- Visión completa de la funcionalidad
- Componentes menos cohesivos o redundancia

Proyectos de gran porte

● Combinaciones:

- Equipos horizontales y sub-equipos verticales o viceversa, en base a la arquitectura.

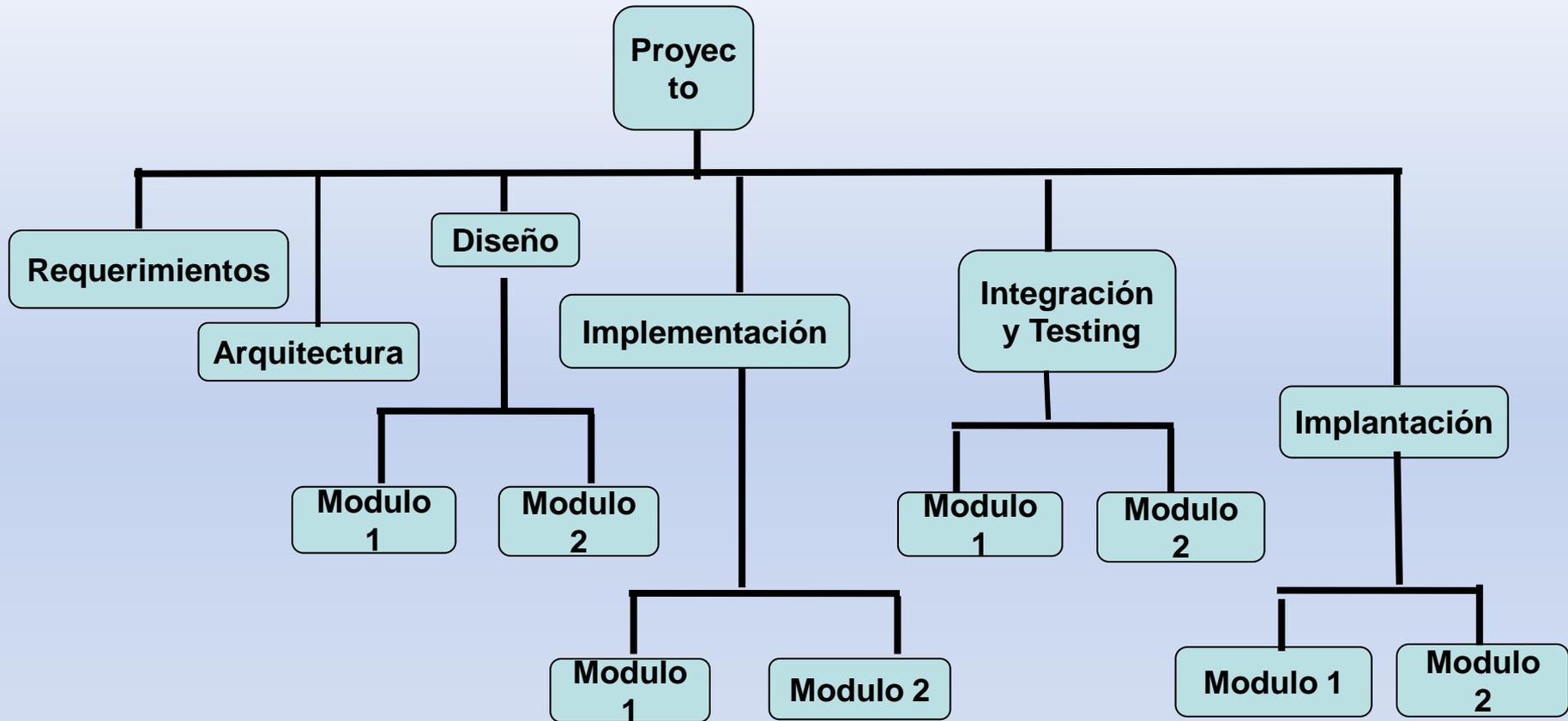
● Integrar participantes del diseño

- Visión más global
- Compromiso con la estimación

Organización del cronograma

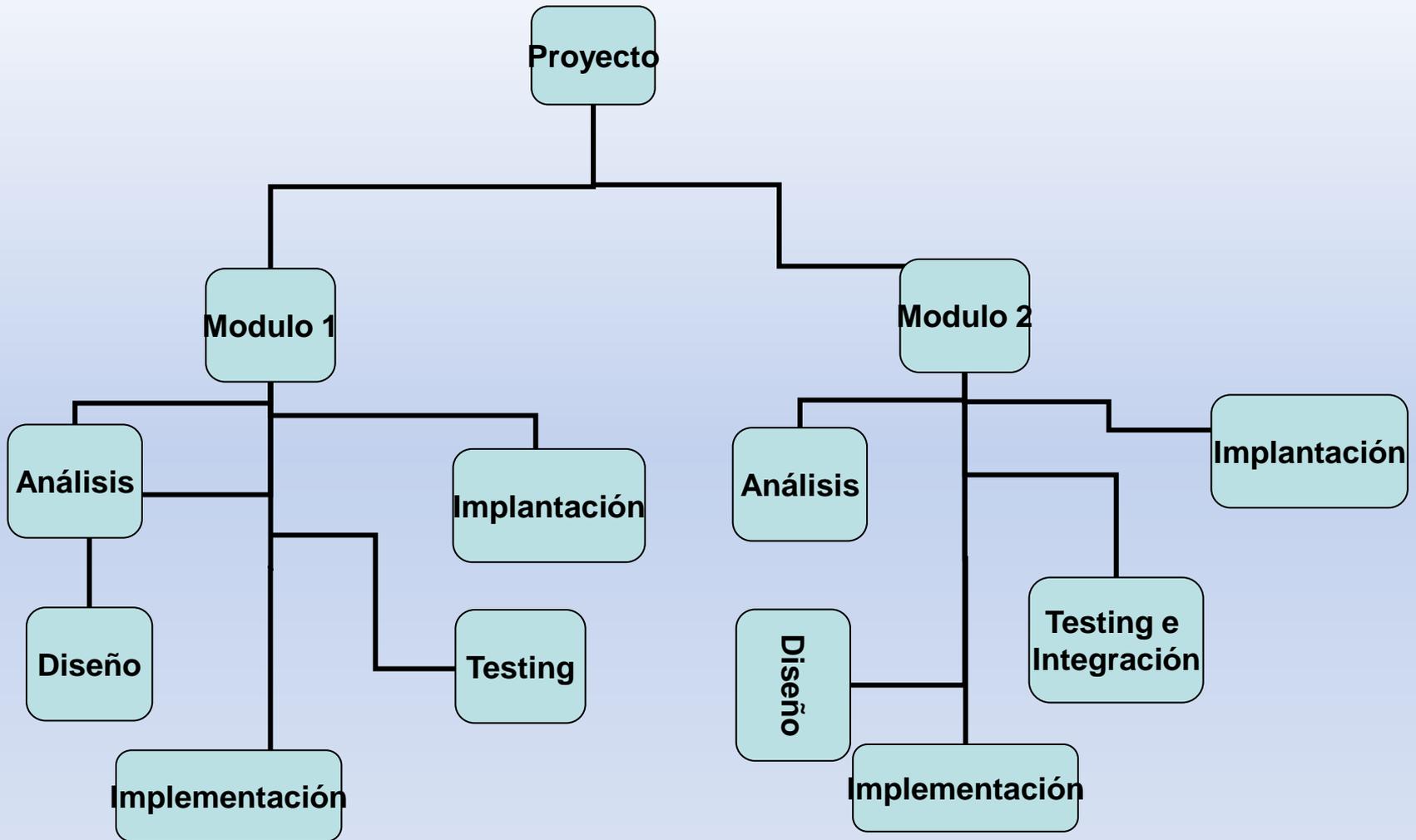
- WBS – Work BreakDown Structure
 - Orientadas a la función
 - Orientadas a la fase
 - Orientadas al producto
 - Híbridos

WBS Orientada a la Fase



WBS Orientada a la fase

WBS Orientada al Producto

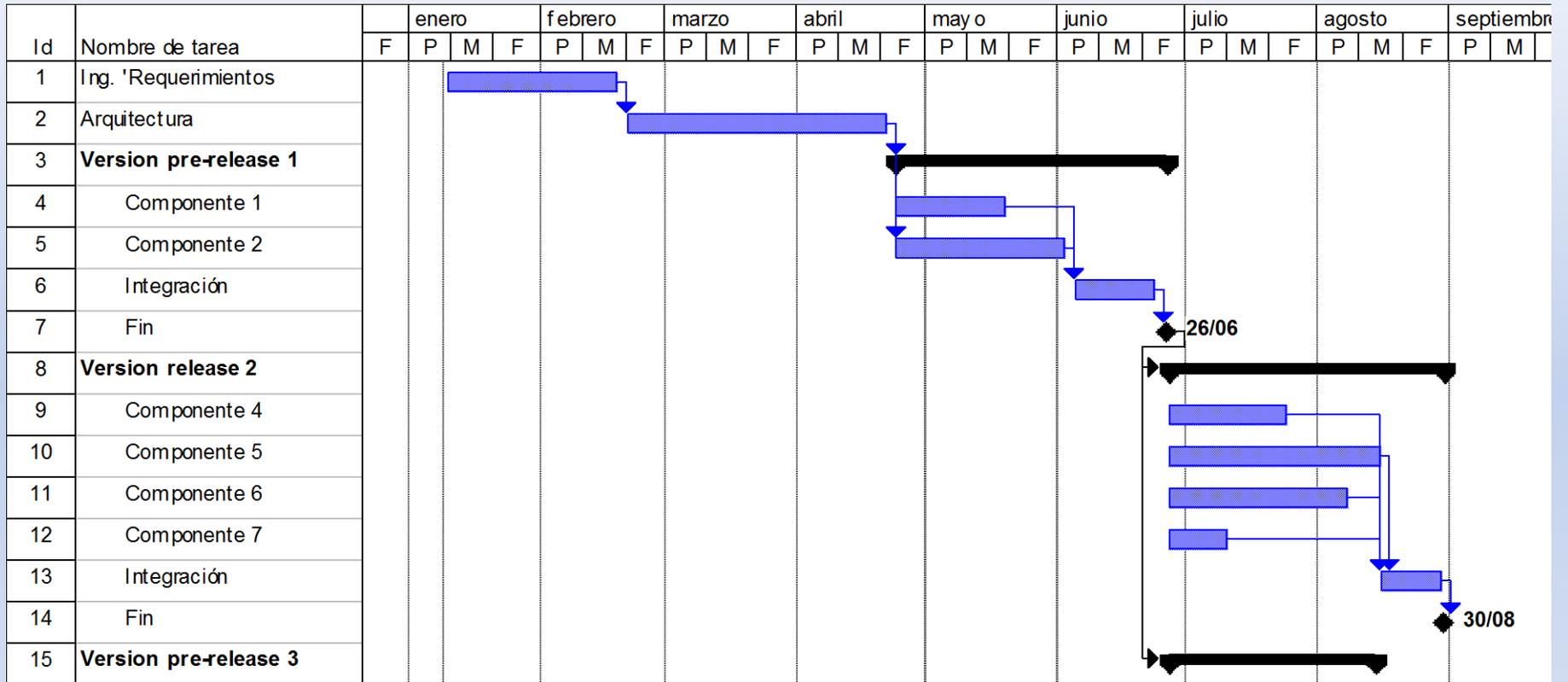


WBS orientada al producto

Cronograma

- Sugerencia: imitar el ciclo de vida.
- Hitos según plan de versiones
- Componentes de las iteraciones surgen del arquitectura

Cronograma

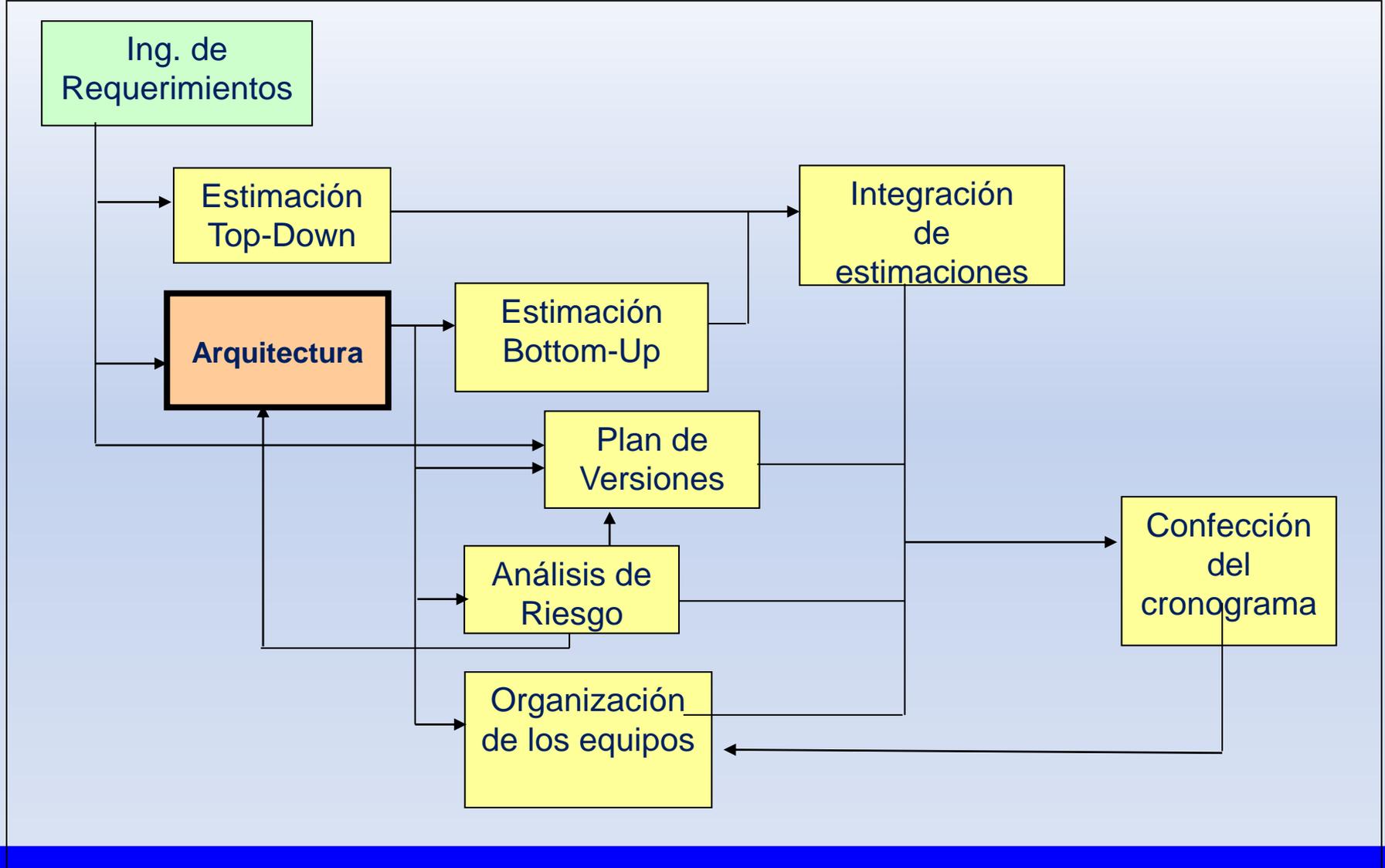


Factores a considerar

- Estimación del esfuerzo de cada componente
- Cantidad de recursos humanos, perfiles y requerimientos de cada componente
- Secuencia de desarrollo según el plan de versiones.
- Otras restricciones:
 - Tiempo máximo de desarrollo
 - Presupuesto
 - Flujo de caja, etc.

Resumen del Proceso

Proceso de planificación y organización del proyecto



Resumen Proceso

- A partir de los requerimientos funcionales y no funcionales se diseña la arquitectura del producto mientras en paralelo se realiza la estimación top-down del proyecto.
- A partir de la arquitectura se realiza una estimación bottom-up de cada uno de los componentes identificados.
- Se integran ambas estimaciones en la que se utilizará para la planificación del proyecto.
- Se realiza el análisis de riesgo, incorporando a la arquitectura como fuente de posibles riesgos que pueden afectar al proyecto. Algunos de los riesgos identificados pueden ocasionar la toma de decisiones que afecten la arquitectura con lo que se repite el ciclo.

Resumen del Proceso

- A partir de la arquitectura, los requerimientos o necesidades del usuario y del análisis de riesgo, se confecciona el plan de versiones.
- A partir de la arquitectura, se organizan los equipos que participarán en el proyecto.
- A partir del plan de versiones, del resultado del proceso de estimación, de las tareas de prevención y de las decisiones tomadas a partir del análisis de riesgo, y de los equipos previstos se confecciona el cronograma. Algunas consideraciones en el momento de confeccionar el cronograma para tener en cuenta todas las restricciones del proyecto pueden afectar la organización de los equipos, con lo que se vuelve a iterar el ciclo hasta encontrar el punto de equilibrio.
- Una vez definidos el plan de versiones y el cronograma, puede comenzar el desarrollo del proyecto, el cual se gestionará para mantenerlo dentro de lo previsto en los documentos obtenidos como resultado del proceso descrito

Referencias Bibliográficas

- Feder, Mariel. Gestión de proyectos dirigida por la Arquitectura. Congreso Argentino de Ciencias de la Computación. 2003
- *A guide to the Project Management Body of Knowledge (PMBOK)*, Project Management Institute, www.pmi.org.
- Humphrey, W.S, *Managing the software process*
- Bass, Clements y Kazman 1998. *Software Architecture in Practice*. Massachusetts, USA. Addison Wesley.
- Soni D, Nord R y Hofmeister C, *Software Architecture in Industrial Applications*, en Proceedings of the 17th International Conference on Software Engineering, New York: ACM Press, 1995.
- Davis Alan M., *Software Life Cycle Models. Software Engineering project management*, editado por Richard H Thayer, 2ª edición. IEEE Computer Society Press, 1988
- Royce W.W., *Managing the Development of Large Software Systems: Concepts and Techniques*, Proc. 1970 WESCON Technical Papers, Vol. 14, 1970
- Hirsch E., *Evolutionary Acquisition of Command and Control Systems*, Program Manager, Nov. Dec. 1985.
- Giddings R.V, *Accommodating Uncertainty in Software Design*, Comm ACM, Vol 27, N° 5, May 1984.

Referencias Bibliográficas

- Gooma H y Scott D, *Prototyping as a Tool in the Specification of User Requirements*, Proc. 5th IEEE International Conference on Software Engineering, IEEE CS Press, Los Alamitos, California, 1981.
- Boehm B.W. *A Spiral Model of Software Development and Enhancement*, Computer, May 1988.
- Paulish D.J, *Architecture-Centric Software Project Management, A practical guide*, Carnegie Mellon, Software Engineering Institute, 2002.
- Boehm B, *Software Engineering Economics*, E Englewood Cliffs, NJ, Prentice Hall 1981.
- Boehm B. et al, *Software Cost Estimation with Cocomo II*, Upper Saddle River, NJ, Prentice Hall, 2000.
- Paulish D.J, *Architecture-Centric Software Project Management, A practical guide*, Carnegie Mellon, Software Engineering Institute, 2002.
- KRUTCHEN Philippe. Noviembre 1995. *The 4+1 View Model of Architecture*. IEEE Software, 12(6), pp 42-50.
- Higuera R.P, *Software Risk Management*, CMU/SEI-96-TR-012 ESC-TR-96-012, Software Engineering Institute (SEI), www.sei.cmu.edu

Referencias Bibliográficas

- Grey S, *Practical Risk Assessment for Project Management*, John Wiley & Sons
- Karolak D.W, *Software Engineering Risk Management*, IEEE Computer Society Press
- Michaels J.V, *Technical Risk Management*, Prentice Hall
- BASS L. et al. *Quality Attribute Design Primitives*. Technical Note CMU/SEI-2000-TN-017. Diciembre 2000
- BASS L., et al. *Quality Attribute Design Primitives and the Attribute Drive Design Method*. SEI, Carnegie Mellon University, Pittsburgh.
- MOUSQUES Gastón. 2002. *Transparencias Curso de Arquitectura*. Ingeniería de Sistemas, Universidad ORT, Uruguay.
- KAZMAN R. et al. Octubre 1999. *Attribute Based Architectural Styles*. Technical Report CMU/SEI-99-TR022.
- Thayer R, *Software Engineering Project Management*
- Rees F, *Equipos de trabajo*, Prentice Hall 1997
- Simons, Lucarelli, *Work Breakdown Structures*
- Gido, Clements, *Network Planning and Scheduling*
- Pinto J, *Project Management Handbook*, PMI