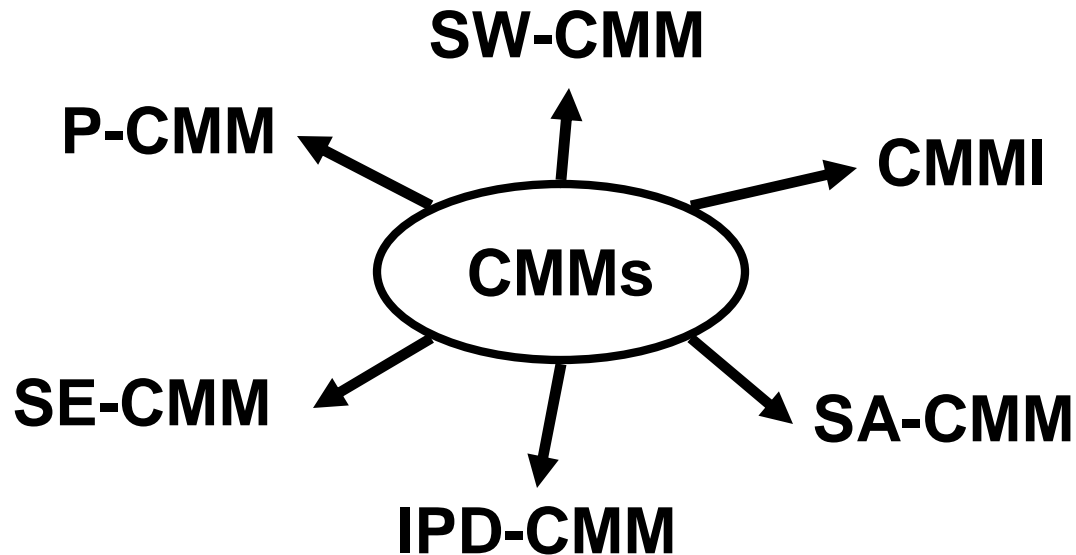


SW-CMM

**“Capability Maturity Model
for Software”**

Introducción

- **1986 – Comienzan Estudios.**
 - ❑ **SEI (Software Engineering Institute - UCM).**
- **1991 – Nace CMM v1.0**
- **1994 – CMM v1.1**



Conceptos Previos

➤ **Proceso de software:**

“Conjunto de actividades, métodos, prácticas, y transformaciones que se usan para desarrollar y mantener el software y sus productos asociados”.

➤ **Capacidad:**

“Describe el rango de resultados esperados que pueden ser logrados siguiendo un proceso de software”.

Conceptos Previos

➤ **Madurez:**

“Ámbito o contexto en el cual un proceso específico es explícitamente definido, administrado, medido, controlado, efectivo”.

➤ **Proceso maduro:**

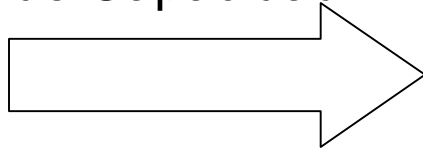
- Bien entendido a lo largo de toda la organización.
- Consistentemente implementado.
- Supervisado y en mejoramiento continuo.

Definición

“Modelo organizacional que describe las mejores prácticas de administración e ingeniería de software, para lograr un proceso maduro de software”.

Guiar en la selección de estrategias de mejora.

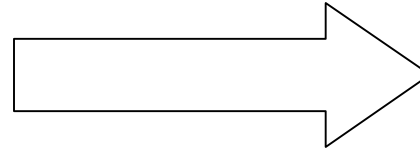
Evaluación
de Capacidad



Proceso de
Software

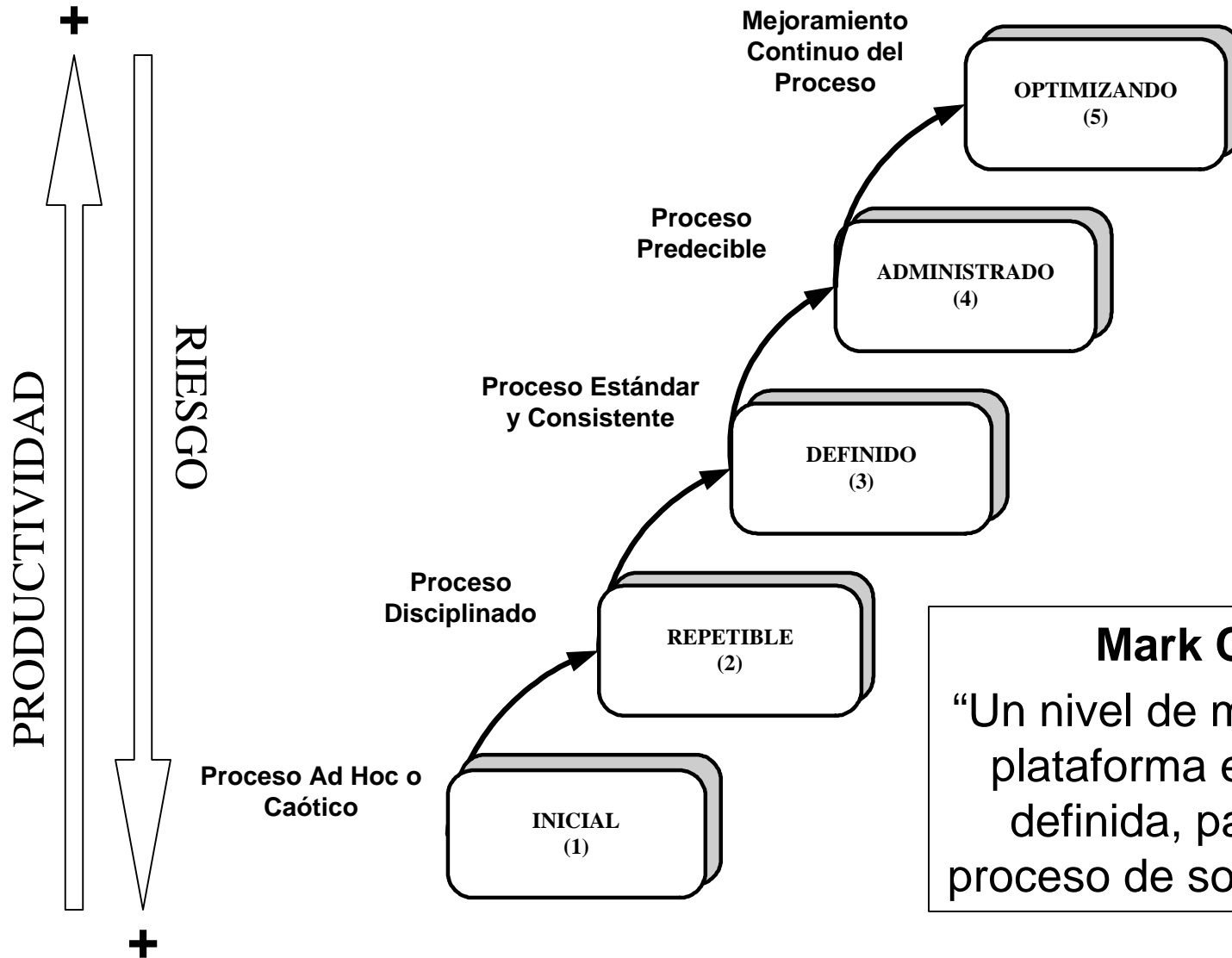


Nivel de Madurez



“Mejorar el proceso y la calidad del software”.

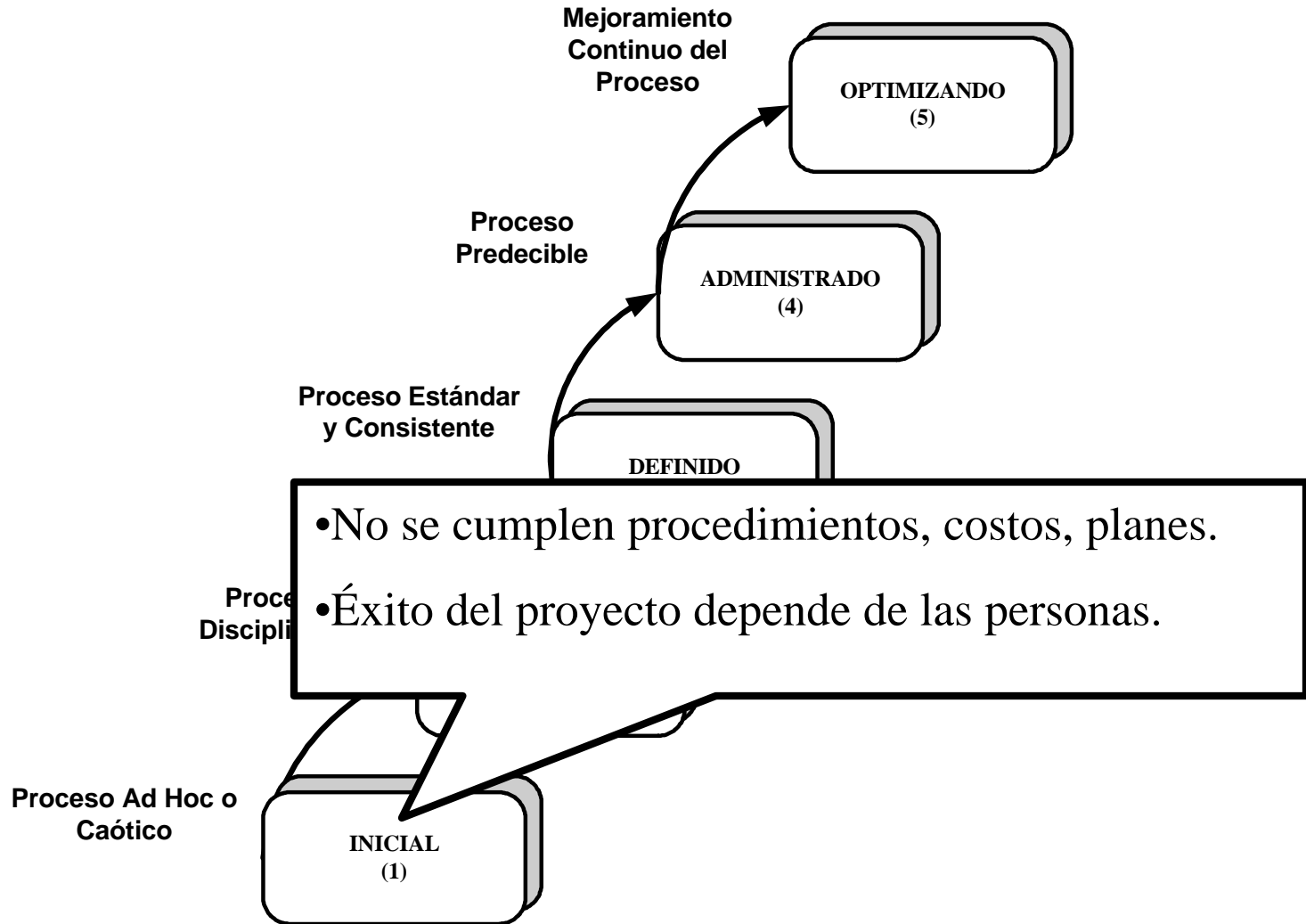
Niveles de Madurez



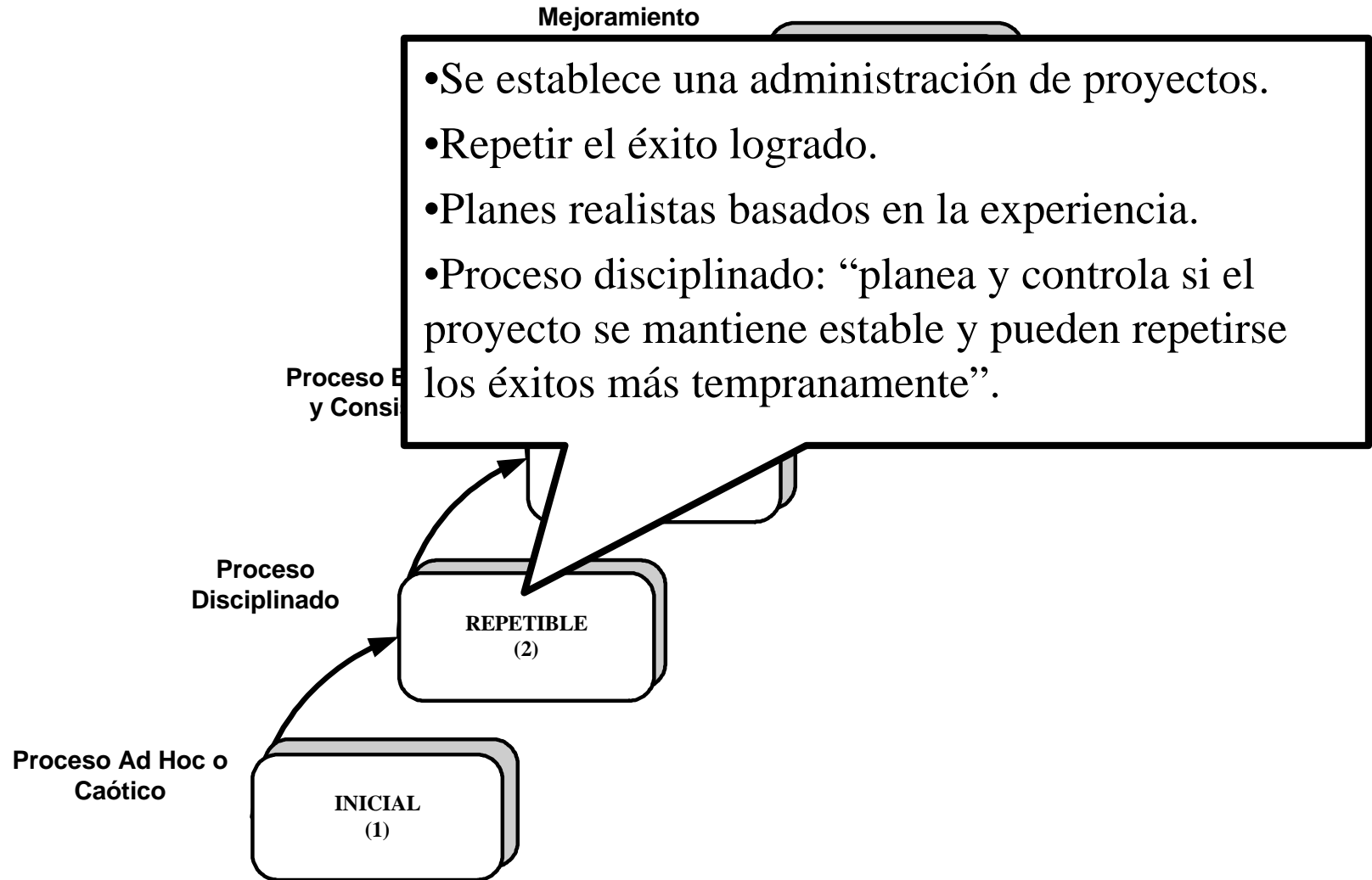
Mark C. Paulk

“Un nivel de madurez es una plataforma evolutiva bien definida, para lograr un proceso de software maduro”

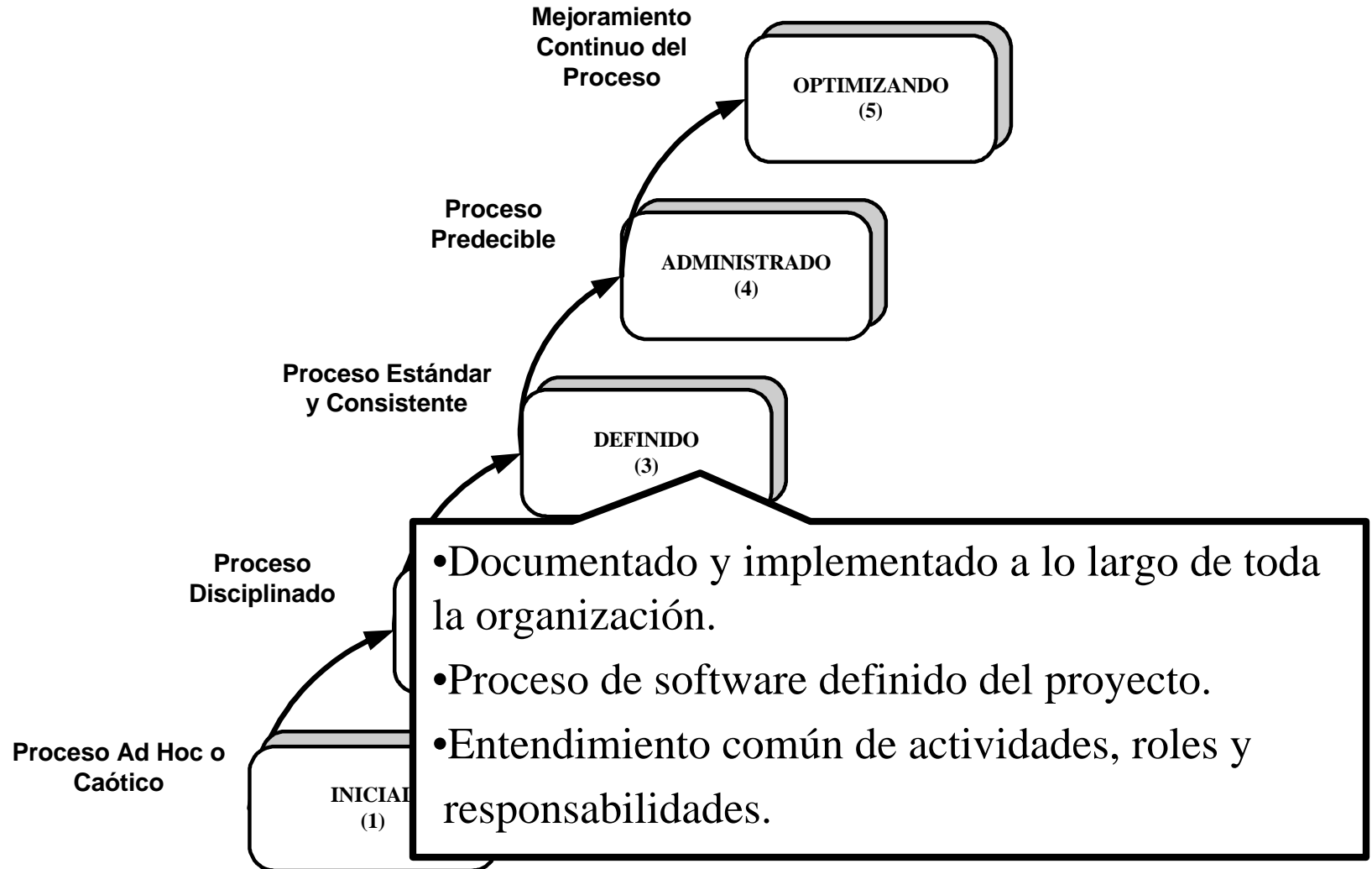
Nivel 1 - Inicial



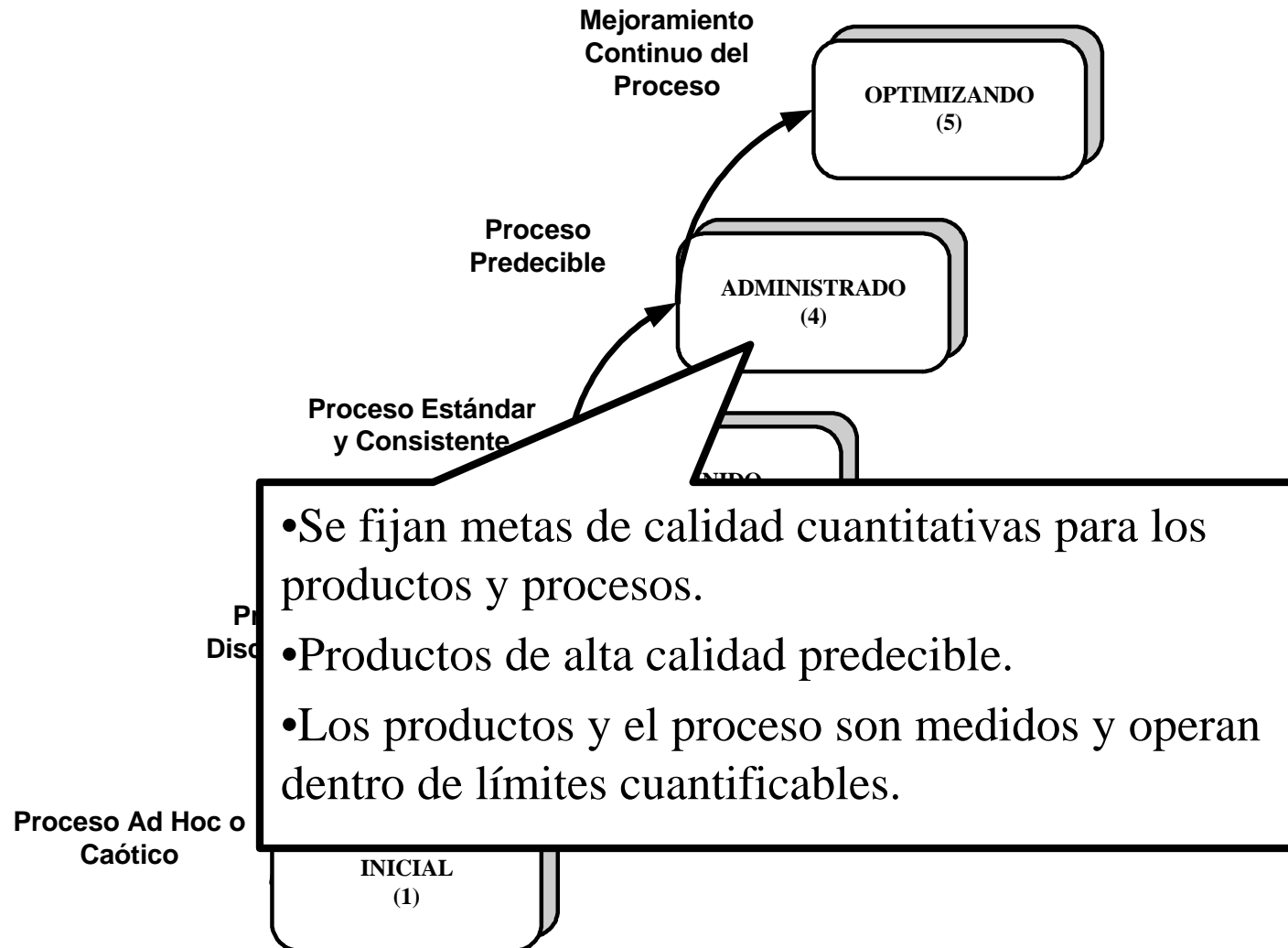
Nivel 2 - Repetible



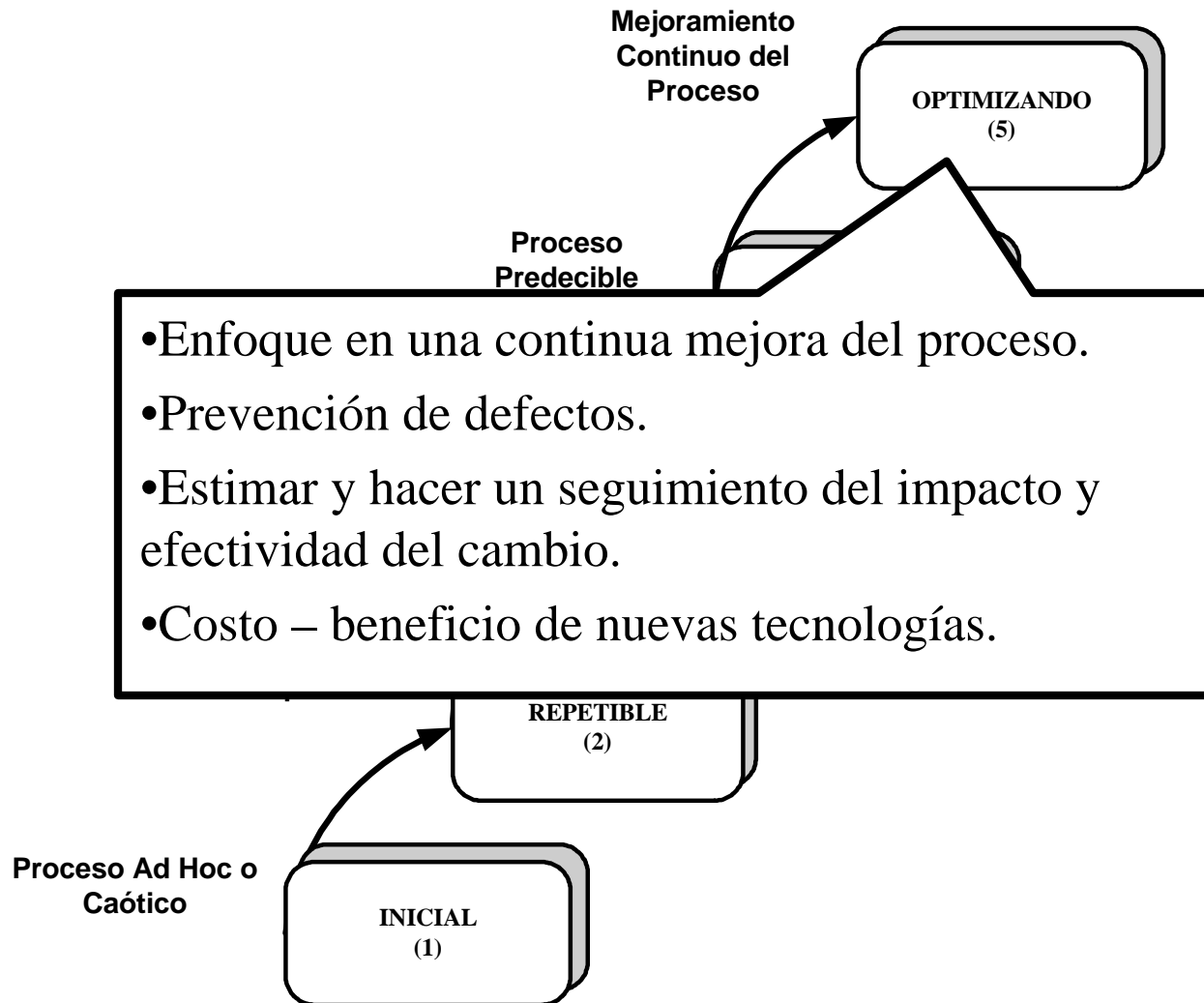
Nivel 3 – Definido



Nivel 4 - Administrado



Nivel 5 - Optimizando



Área clave de proceso (ACP)

Mark C. Paulk

“Grupo de actividades relacionadas que cuando se llevan a cabo en conjunto alcanzan un conjunto de metas consideradas importantes para aumentar la capacidad del proceso”.

ACPs de Nivel 2

➤ **Administración de Requerimientos.**

- ❑ Establecer un entendimiento común entre el cliente y el proyecto de software, acerca de los requerimientos del cliente.
- ❑ Base para estimar, planificar, ejecutar y seguir las actividades del proyecto de software.

➤ **Planificación del Proyecto de Software.**

- ❑ Establacer planes razonables para realizar las tareas de administración e ingeniería de software.

ACPs de Nivel 2

➤ **Seguimiento y Control del Proyecto de Software.**

- ❑ Proporcionar una adecuada visión del avance real del proyecto, de forma que la administración pueda tomar acciones efectivas cuando el rendimiento del proyecto de software se desvíe del plan de software.

➤ **Administración de Subcontratos de Software.**

- ❑ Seleccionar subcontratistas de software calificados y administrarlos efectivamente.

ACPs de Nivel 2

- **Aseguramiento de la Calidad del Software (SQA).**
- ❑ Su propósito es dar a la administración una visibilidad adecuada del proceso que está siendo usado y los productos que están siendo construidos.
- ❑ El aseguramiento de la calidad del software involucra revisar y auditar los productos y actividades de software, a fin de asegurar que ellos cumplan con los estándares y procedimientos aplicables

ACPs de Nivel 2

- **Administración de la Configuración del Software.**
- Establecer y mantener la integridad de los productos de software a través del ciclo de vida del proyecto.
 - Identificar productos.
 - Mantener la integridad de los productos.

Establecer una librería de líneas base del proyecto.

Línea Base de Software: “especificación o producto que ha sido revisado y acordado formalmente y que sirve como base para el desarrollo futuro.”

ACPs de Nivel 3

➤ **Administración Integrada de Software.**

- ❑ Integrar las actividades de ingeniería de software y de administración en un proceso coherente y definido (proceso de software definido del proyecto).
- ❑ Desarrolla el proceso de software definido del proyecto.
- ❑ Administra el proyecto usando este proceso.

➤ **Coordinación Intergrupala.**

- ❑ Acuerdo intergrupala de requisitos, objetivos y planes.
- ❑ Los compromisos entre los grupos, son acordados por los grupos afectados.

ACPs de Nivel 3

➤ **Enfoque en el Proceso de la Organización.**

- ❑ Desarrollar y mantener una comprensión de los procesos de software de la organización.
- ❑ Coordinar las actividades para evaluar, desarrollar, mantener y mejorar tales procesos.

➤ **Definición del Proceso de Software de la Organización.**

- ❑ Desarrolla y mantiene el proceso de software estándar de la organización y sus elementos.

ACPs de Nivel 3

➤ Programa de Capacitación.

- ❑ Desarrollar las habilidades y el conocimiento de las personas, para que puedan cumplir sus roles en forma eficaz y efectiva.

➤ Ingeniería de Productos de Software.

- ❑ Involucra ejecutar las actividades de ingeniería de para construir y mantener el software usando el proceso de software definido del proyecto, métodos y herramientas apropiadas.
- ❑ Análisis de requerimientos, diseño, construcción y prueba.

ACPs de Nivel 3

➤ **Revisiones de Pares.**

- ❑ Remover los defectos de los productos de trabajo de software temprana y eficientemente.
- ❑ los productos son revisados entre colegas.
- ❑ Desarrollar una mejor comprensión de los productos de trabajo de software y de los defectos que pueden ser prevenidos.

ACPs de Nivel 4

- **Administración Cuantitativa del Proceso.**

- ❑ Su propósito es controlar cuantitativamente el rendimiento del proceso en el proyecto de software.

- **Administración de la Calidad del Software.**

- ❑ Su propósito es desarrollar una comprensión cuantitativa de la calidad de los productos de software del proyecto y lograr metas específicas de calidad.

ACPs de Nivel 5

➤ **Prevención de Defectos.**

- ❑ Su propósito es identificar las causas de los defectos y prevenir que se repitan. El proyecto de software analiza los defectos, identifica sus causas, y cambia su proceso definido de software.

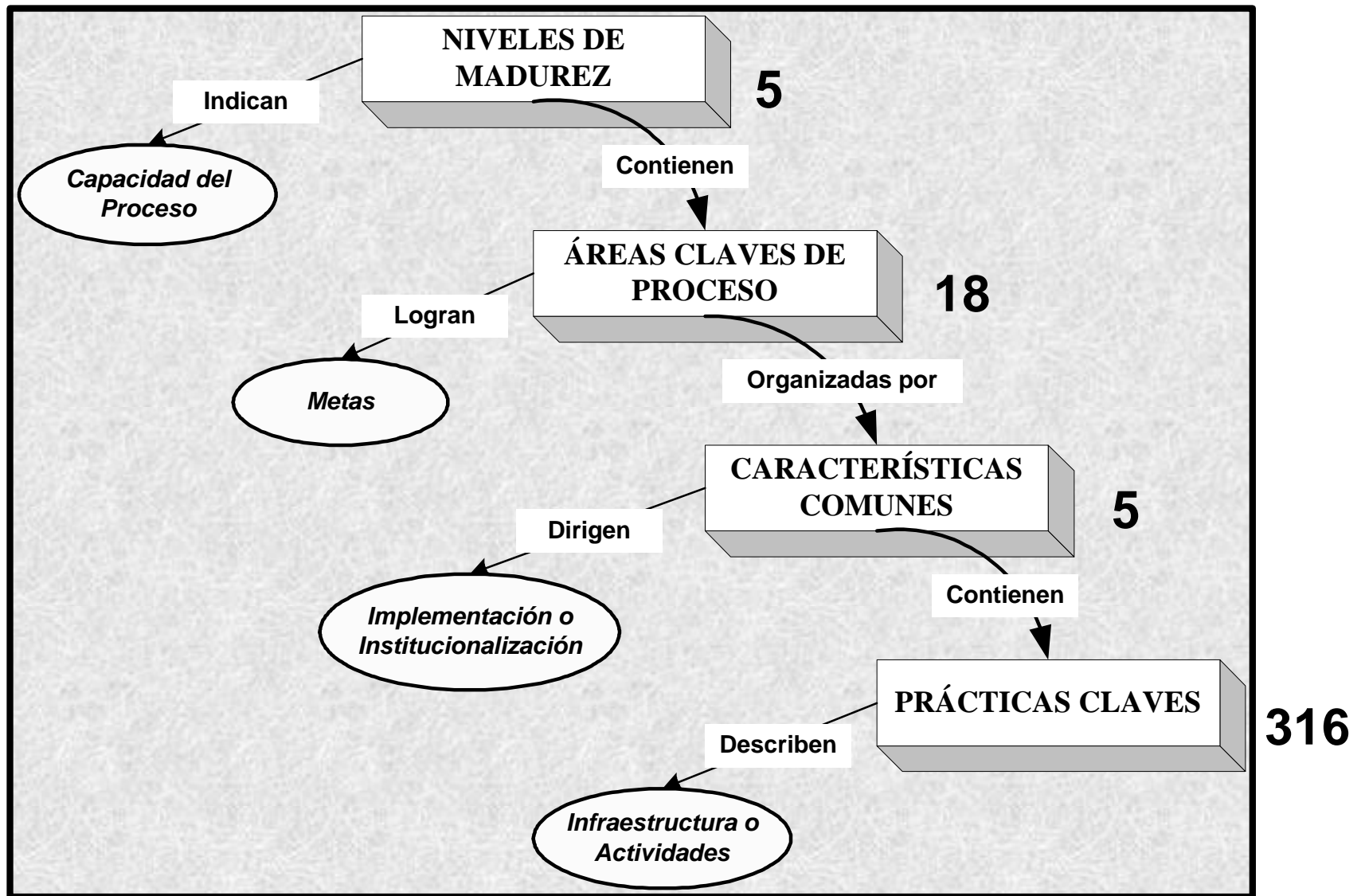
➤ **Administración de Cambios Tecnológicos.**

- ❑ Su propósito es identificar las nuevas tecnologías (es decir, herramientas, métodos, y procesos) que sean beneficiosas y transferirlas a la organización de una manera ordenada.

ACPs de Nivel 5

- **Administración del Cambio de Procesos.**
- ❑ Su propósito es mejorar los procesos de software usados en la organización.
 - ✓ Mejorar la calidad del software.
 - ✓ Aumentar la productividad.
 - ✓ Acortar el tiempo de desarrollo del producto.

Estructura del CMM



Características Comunes

➤ **Compromiso para desarrollar.**

“Describe las acciones que la organización debe realizar para asegurar que el proceso sea establecido y pueda perdurar”.

- Políticas.
- Compromiso de la alta gerencia.

➤ **Habilidad para desarrollar.**

“Describe las pre-condiciones que deben existir en el proyecto u organización para implementar el proceso de software en forma competente”.

- Recursos y financiamiento.
- Capacitación.
- Orientación.
- Prerrequisito.

Características Comunes

➤ **Actividades realizadas.**

“Describe los roles y procedimientos necesarios para implementar una área clave de proceso”.

Planes.

Procedimiento documentado.

➤ **Medición y análisis.**

“Describe la necesidad de medir el proceso y analizar las medidas”.

Características Comunes

➤ Verificando la implementación.

“Describe los pasos para asegurar que las actividades realizadas estén de acuerdo al proceso establecido”.

- Supervisión del senior manager.
- Supervisión de la administración del proyecto.
- Actividades de aseguramiento de calidad del software.

Características Comunes

➤ Área Clave del Proceso.

¿ACP Implementada?

Metas.

Compromiso a Desarrollar.

Habilidad a Desarrollar.

Medidas y Análisis.

Verificando la implementación.

Actividades realizadas.

Bases

Implementación

FIN

PREGUNTAS

¿?