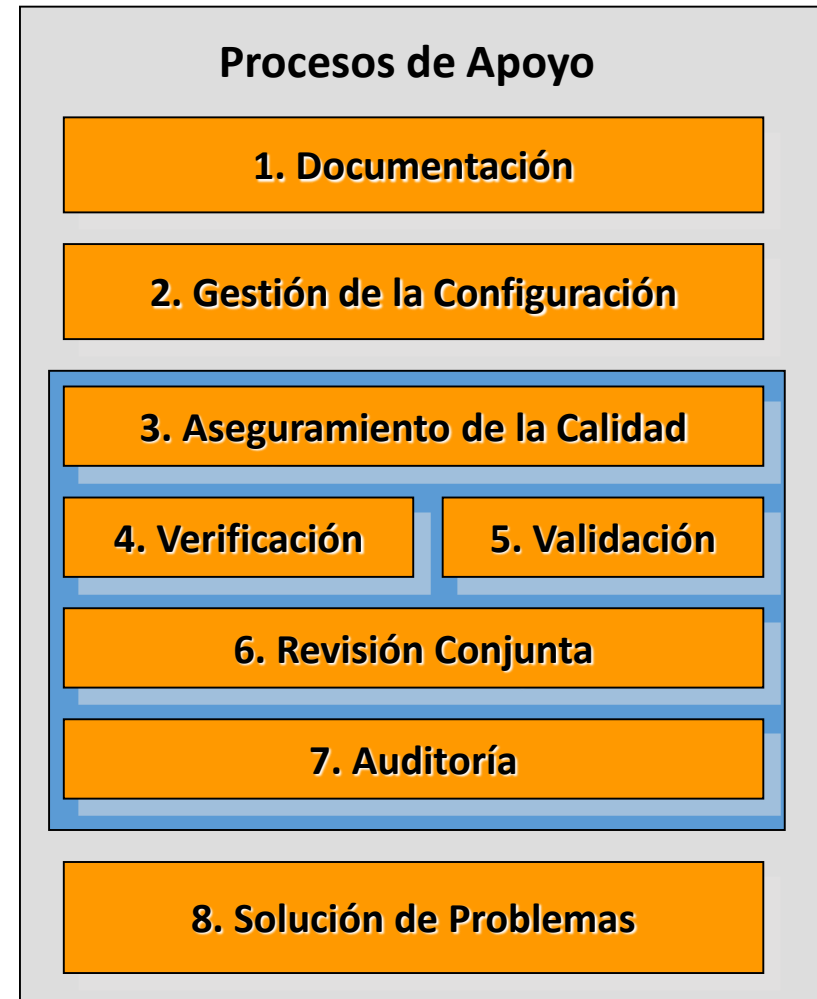
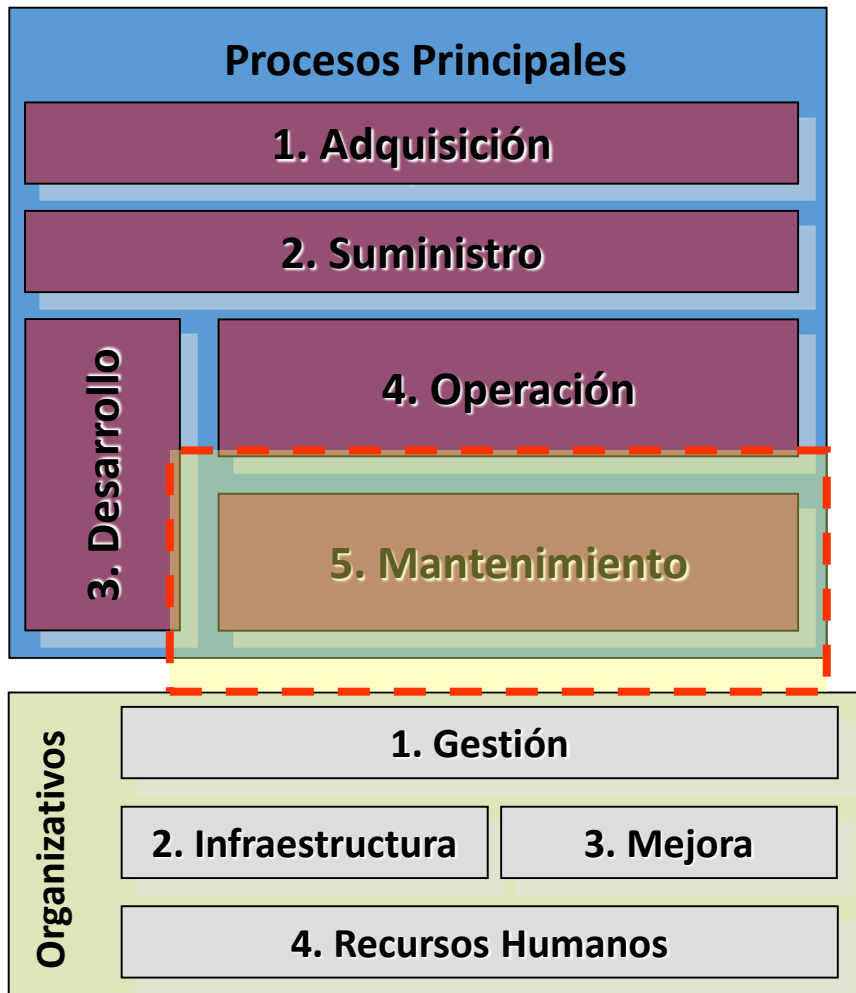
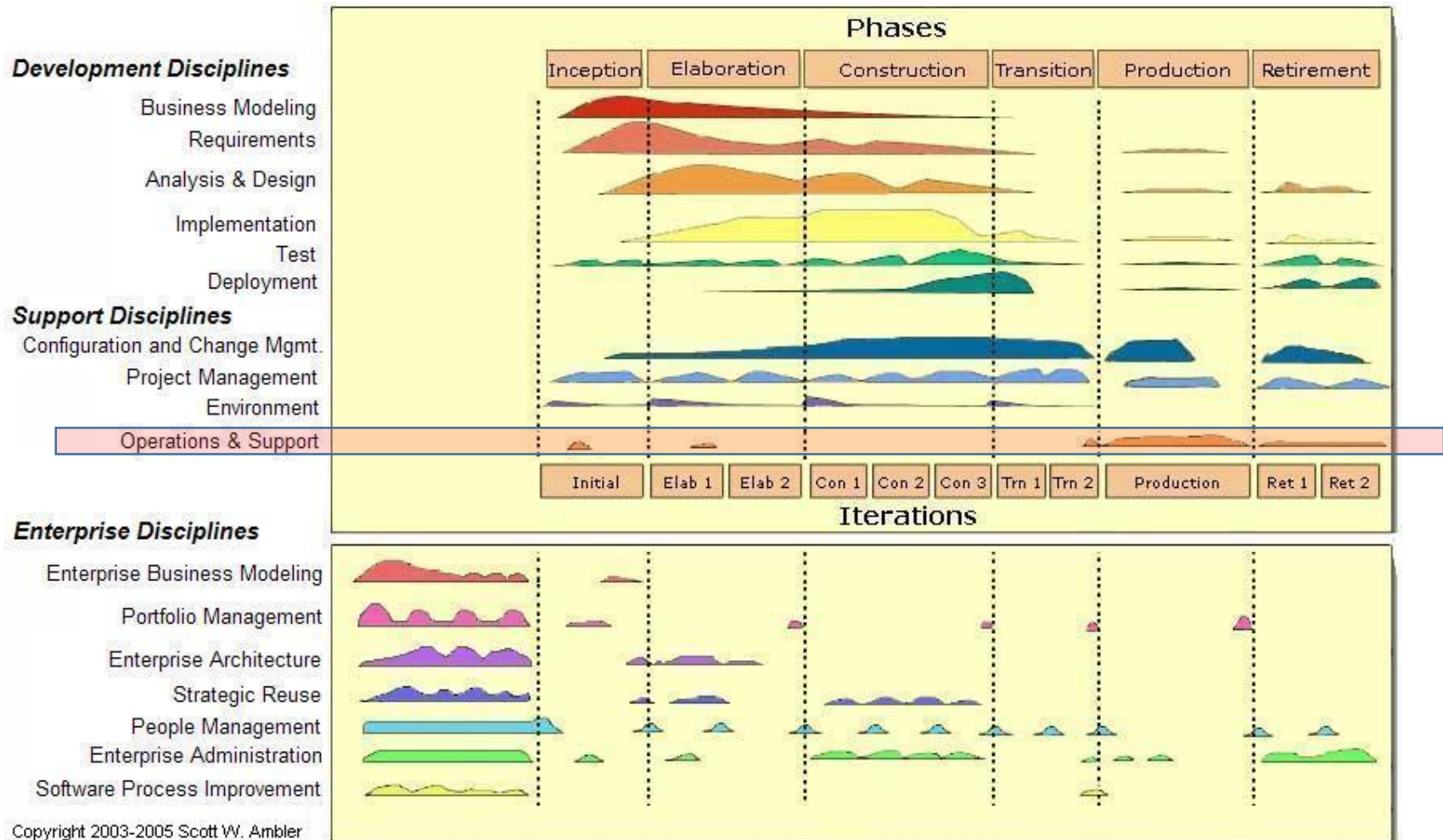


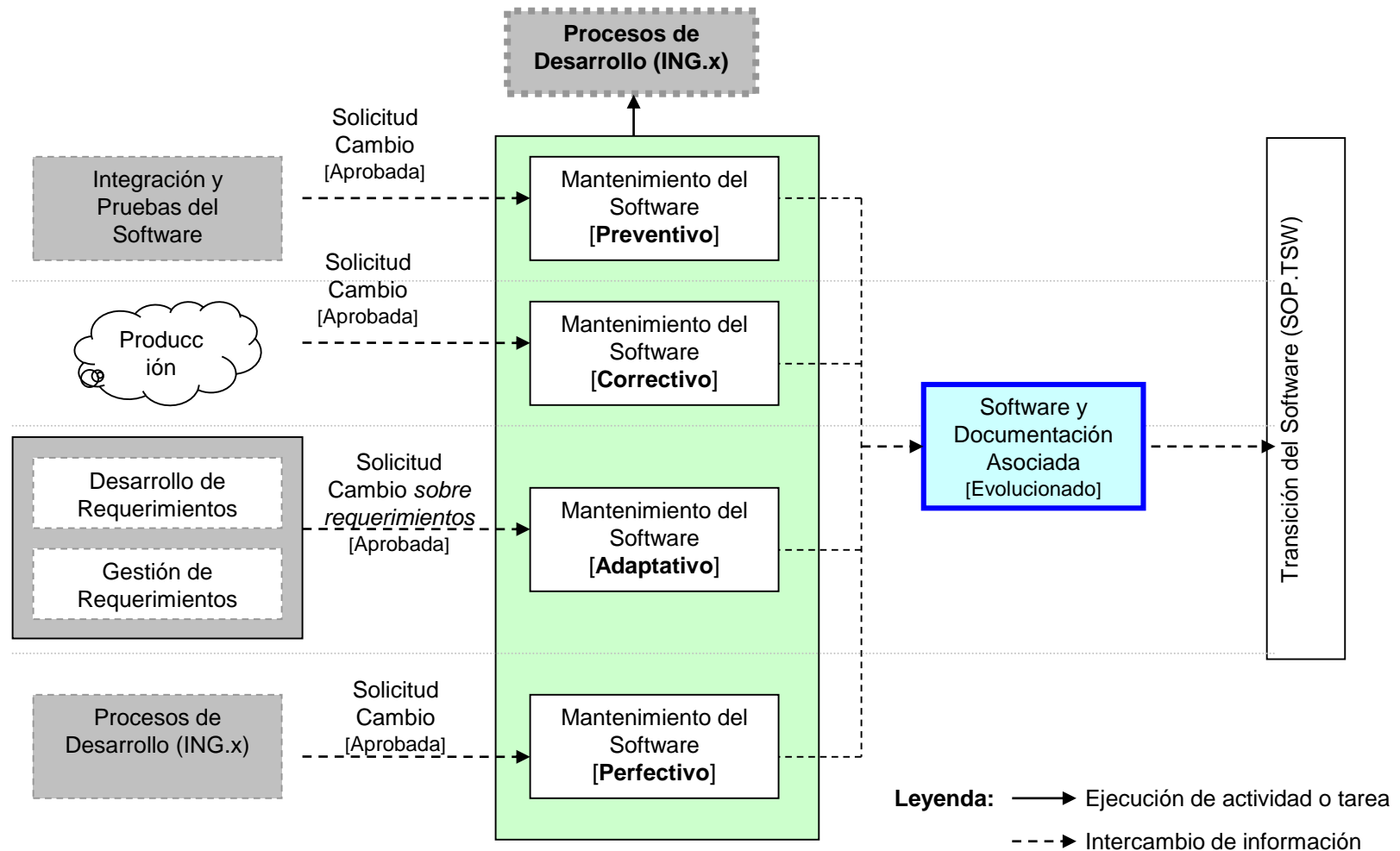
Unidad 6 – Mantenimiento de Software

Ciclo de Vida NTP-ISO/IEC 12207





Esquema del Mantenimiento de Software



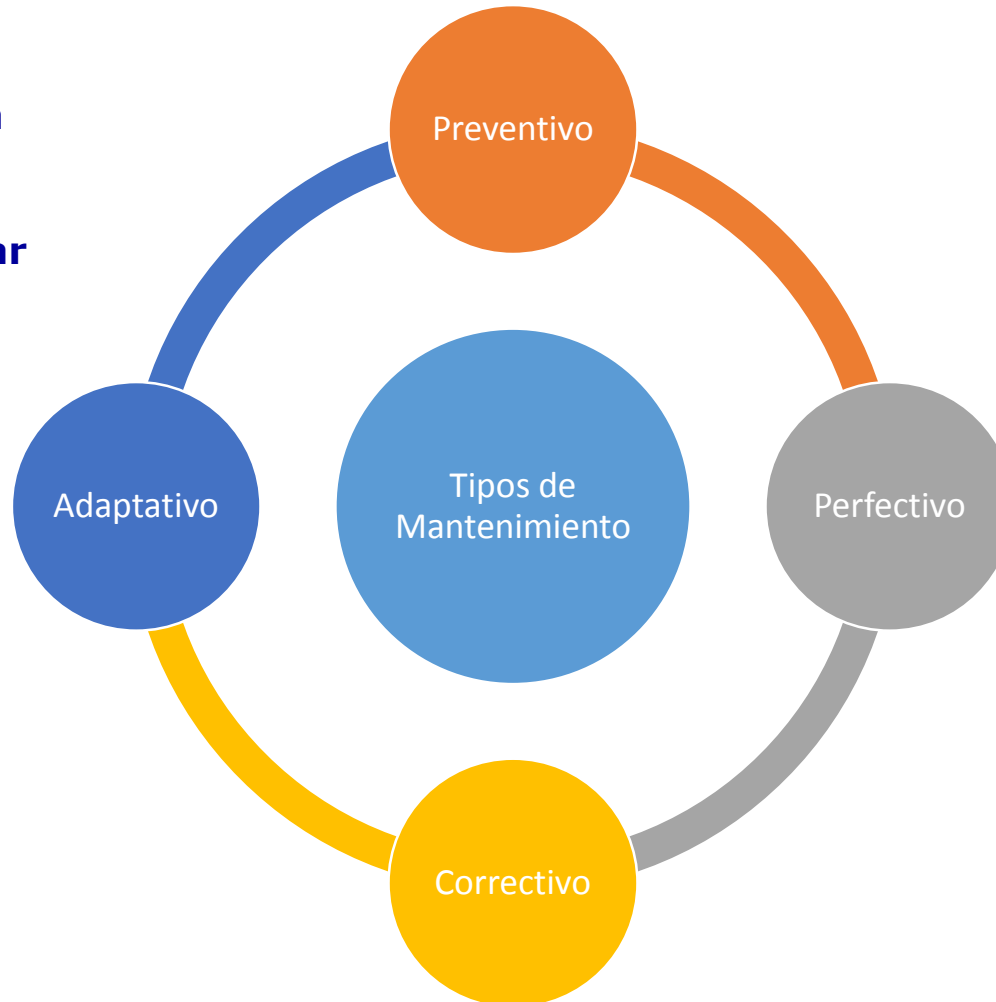
Mantenimiento de Software

- Su objetivo principal modificar el producto software existente preservando su integridad, incluyendo la migración y retirada del producto software



Tipos de Mantenimiento

la medida cualitativa de la facilidad para comprender, corregir y adaptar o mejorar el software.



La mantenibilidad es un atributo de calidad del software.

Mantenimiento Correctivo del Software

- Permite realiza la corrección de errores o fallas encontradas en el producto software en producción.
- Su propósito esencial consiste en cambiar el software para corregir aquellos defectos encontrados en el software.
- Es el más común ya que siempre es de carácter “urgente”.



Mantenimiento Adaptativo del Software

- Incluye aquellas modificaciones requeridas debidas al cambio de la configuración.
- Este mantenimiento garantiza cambiar al software para acomodarlo a aquellos cambios que se producen en su entorno de operación.
- Puede ser la adaptación del software a una nueva versión e sistema operativo o componente.



Mantenimiento Preventivo del Software

- Incluye las incorporaciones, modificaciones y eliminaciones necesarias en un producto para cubrir la expansión o cambio en las necesidades del usuario.
- Permite detectar y corregir fallas latentes en el producto software antes de que estas se manifiesten como fallas efectivas.
- Requiere un análisis proactivo, y disponibilidad de recursos y tiempo para poder llevarlo a cabo.



En su versión de 1993, el estándar IEEE 1229 incluía también en su clasificación el mantenimiento preventivo como aquel que se realiza para evitar la aparición futura de errores, o para mejorar la integridad de producto en prevención de éstos. Algunos textos lo consideran como un 4º tipo de mantenimiento, y otros lo incluyen como mantenimiento correctivo.

Mantenimiento Perfectivo del Software (Reingeniería del Software)

- Incluye las acciones llevadas a cabo para mejorar la calidad interna de los sistemas en cualquiera de sus características:
 - reestructuración del código,
 - definición más clara del sistema y
 - optimización del rendimiento y eficiencia.
- El mantenimiento perfectivo produce cambios en el software a fin de que se pueda corregir, adaptar y mejorar de una manera más fácil.
- Se busca mejorar el desempeño y la facilidad de mantenimiento del producto software.

Las 8 reglas de Lehman



1. Cambio Continuo



2. Complejidad
Creciente



3. Autorregulación



4. Conservación de la
Estabilidad
Organizacional



5. Conservación de la
familiaridad



6. Crecimiento
continuo



7. Calidad decreciente



8. Sistema de Retro-
alimentación

- Sistema E-type

- Sistemas software que solucionan un problema o implementan una aplicación para computadoras en el *mundo real*.

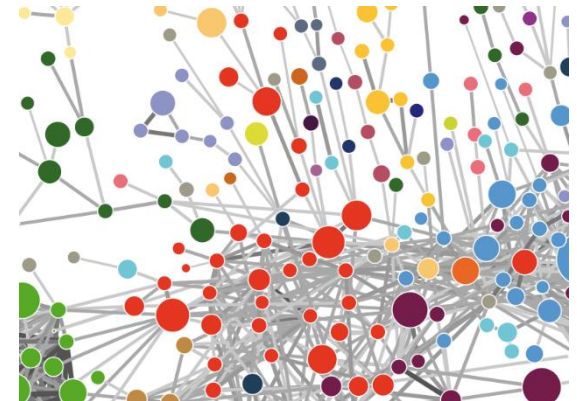
Ley 1 -Cambio Continuo

- Un programa E-type que es usado debe continuamente adaptarse sino este se transforma progresivamente en un software poco satisfactorio.
- La evolución se logra a través de un proceso de mantenimiento controlado y conducido por la retroalimentación.



Ley II-Complejidad Creciente

- En la medida que un programa evoluciona su complejidad aumenta a menos que se realice trabajo de mantenimiento o se reduzca su complejidad.
 - Al surgir necesidades de adaptación y los cambios se implementan de manera sucesiva, las interacciones y dependencias entre los elementos del sistema se incrementa de acuerdo a un patrón poco o nada estructurado y por ende conduce a una entropía creciente del sistema



Ley III-Autoregulación

- El proceso de evolución del programa es auto regulado y basado en una distribución normal de las dedidas del proceso y atributos del proceso:
 - La retroalimentación positiva y negativa que implementa los cambios chequea ty genera un balance que proporciona un ejemplo de mecanismo de retroalimentación al crecimiento y estabilización controlado.



Ley IV-Conservación de la Estabilidad Organizacional

- La efectividad promedio global sobre un sistema que evoluciona es invariante sobre el ciclo de vida del producto:
 - Conduce a la estabilización del producto que evoluciona



Ley V-Conservación de la Familiaridad

- *Durante la Vida Activa de un programa que evoluciona, el contenido de releases sucesivos es estáticamente invariable:*
 - Involucramiento de todos los participantes en el desarrollo y evolución del producto con el producto en sí mismo.



Ley VI-Crecimiento Continuo

- El contenido funcional de un programa tiene que incrementarse de manera continua para mantener la satisfacción de los usuarios en el tiempo.



Ley VII-Calidad decreciente

- Los programas E-type se perciben como productos que decrementan su calidad a menos que sean rigurosamente mantenidos y adaptados a los entornos operacionales cambiantes bajo los que operan.



Ley VIII-Sistema de Retroalimentación

- Los procesos de programación de sistemas E-Type son Multi-lazos, con retroalimentación a diferentes niveles y deben tratarse como tales para que se puedan mejorar y modificar satisfactoriamente.

