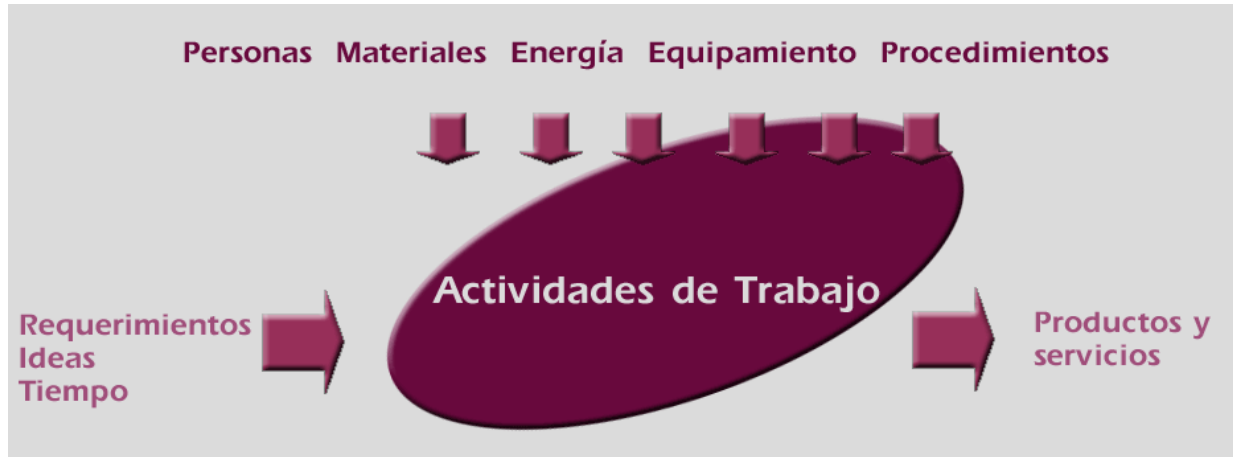


---

**Universidad Tecnológica Nacional**  
**Facultad Regional Córdoba**  
**Cátedra de Ingeniería y Calidad de Software**  
**Docentes: Judith Meles – Laura Covaro**

# COMPONENTES DE PROYECTO DE DESARROLLO DE SOFTWARE

# El proceso de Software

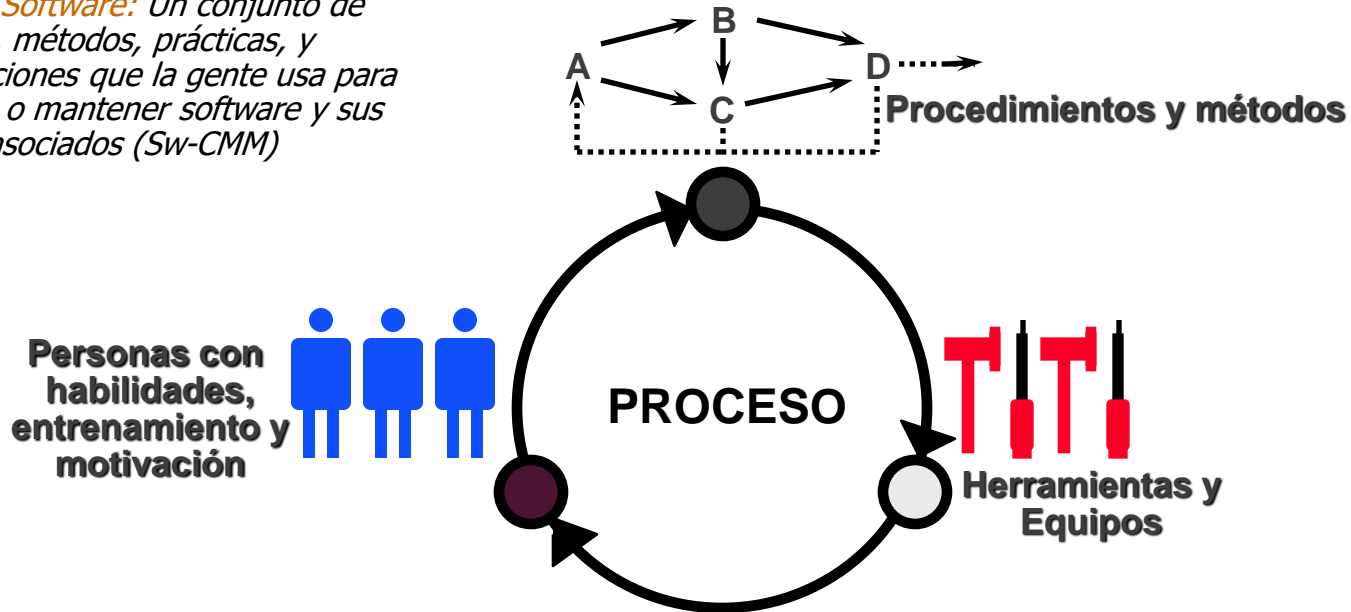


- Conjunto estructurado de actividades para desarrollar un sistema de software
- Estas actividades varían dependiendo de la organización y el tipo de sistema que debe desarrollarse.
- Debe ser explícitamente modelado si va a ser administrado.

# DEFINICIÓN DE UN PROCESO DE SOFTWARE

*Proceso:* La secuencia de pasos ejecutados para un propósito dado (IEEE)

*Proceso de Software:* Un conjunto de actividades, métodos, prácticas, y transformaciones que la gente usa para desarrollar o mantener software y sus productos asociados (Sw-CMM)



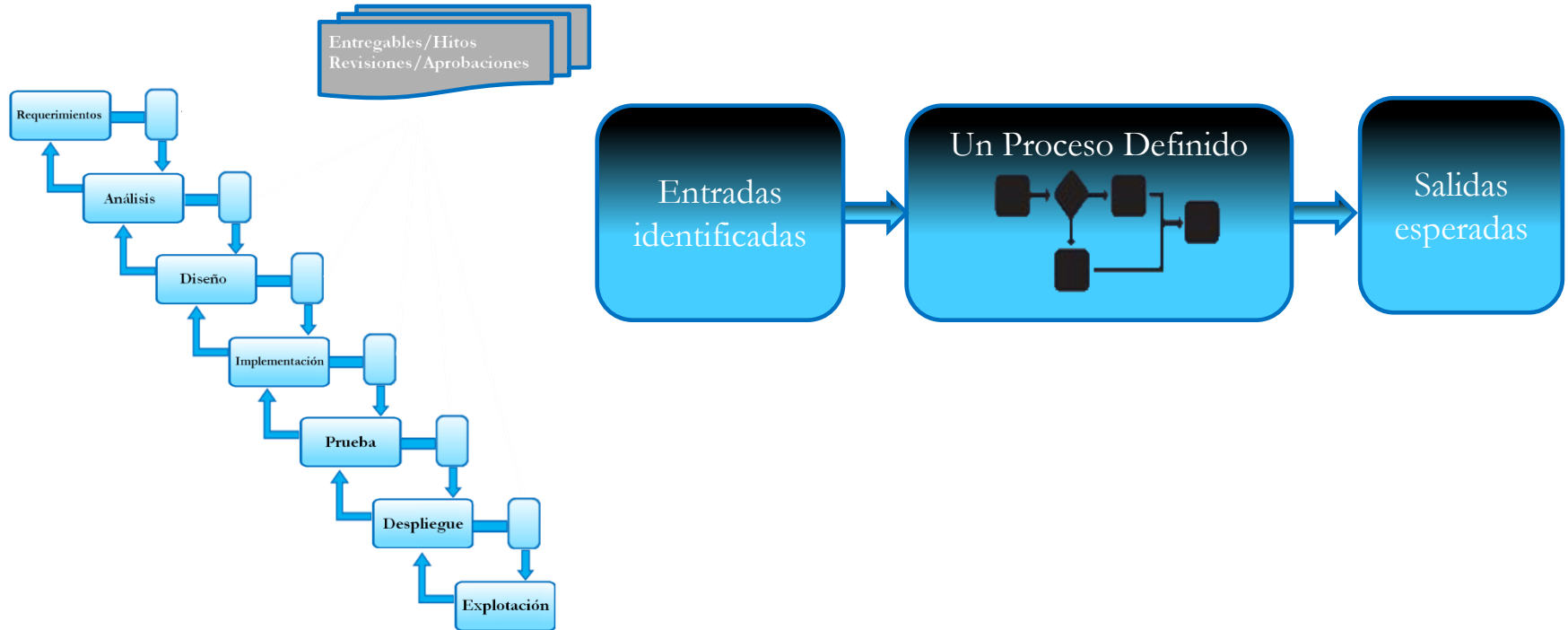
# EMPÍRICO

En la Universidad de California, en Irvine, simplemente sembraron pasto y esperaron 1 año, luego de eso se fijaron donde la gente había hecho “caminito”, entonces ahí construyeron las sendas peatonales

# DEFINIDO (INSPIRADOS EN LAS LÍNEAS DE PRODUCCIÓN)

- Asume que podemos repetir el mismo proceso una y otra vez, indefinidamente, y obtener los mismos resultados.
- La administración y control provienen de la predictibilidad del proceso definido.

# PROCESOS DEFINIDOS

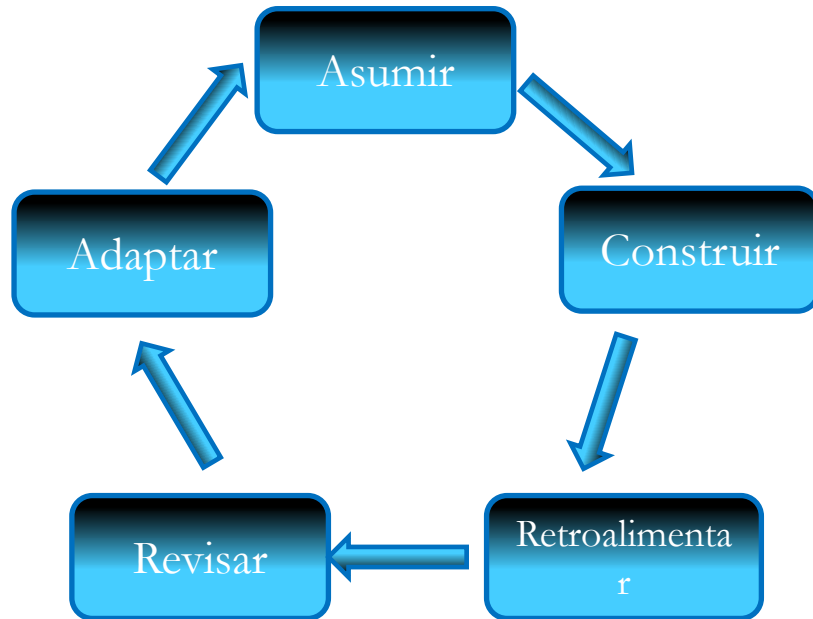


# PROCESOS EMPÍRICOS



- Asume procesos complicados con variables cambiantes. Cuando se repite el proceso, se pueden llegar a obtener resultados diferentes.
- La administración y control se realiza través de inspecciones frecuentes y adaptaciones
- Son procesos que trabajan bien con procesos creativos y complejos.

# PATRÓN DE CONOCIMIENTO EN PROCESOS EMPÍRICOS







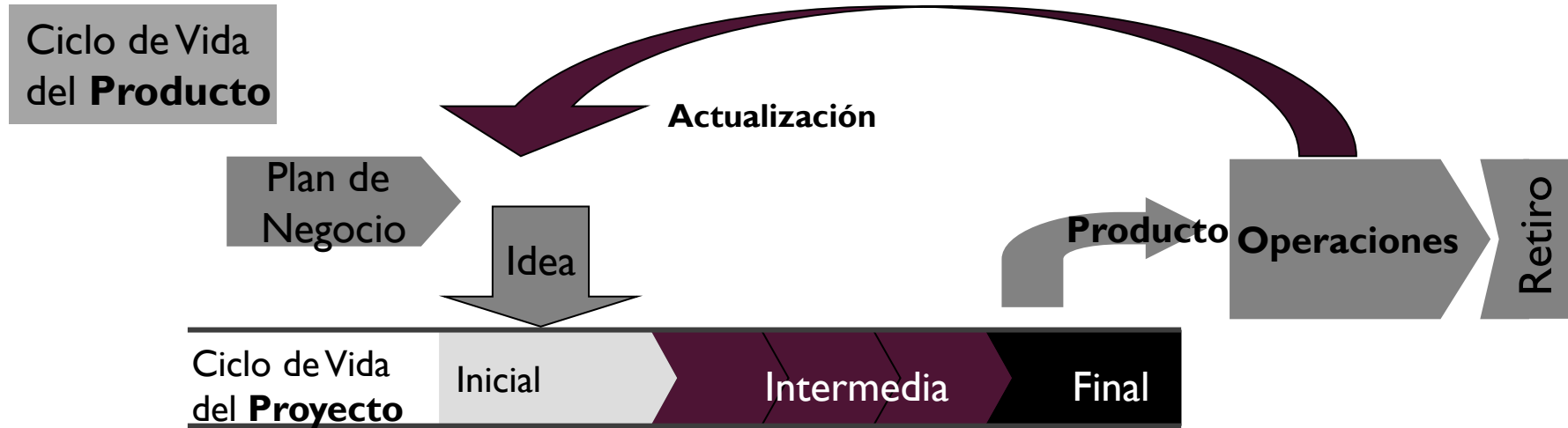
# CICLOS DE VIDA

La serie de pasos a través de los cuales el producto o proyecto progresa.

Los productos tienen su ciclo de vida.

Los proyectos también.

# RELACIÓN: CICLO DE VIDA DEL PROYECTO Y DEL PRODUCTO



## CICLOS DE VIDA DE PROYECTOS DE SOFTWARE

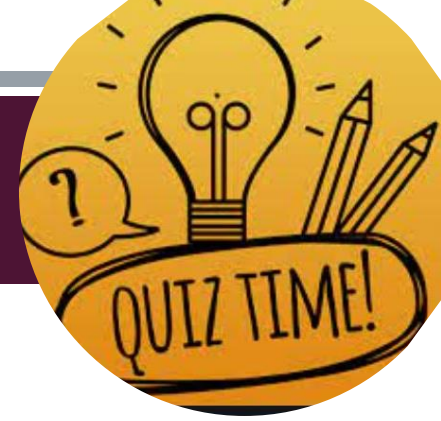
- Un ciclo de vida de un proyecto software es un representación de un proceso. Grafica una descripción del proceso desde una perspectiva particular
- Los modelos especifican
  - Las fases de proceso.
    - Ejemplo: requerimientos, especificación, diseño...
  - El orden en el cual se llevan a cabo

# CLASIFICACIÓN DE LOS CICLOS DE VIDA



- Hay tres tipos básicos de Ciclos de Vida para un proyecto de desarrollo de software
  - Secuencial
  - Iterativo
  - Recursivo

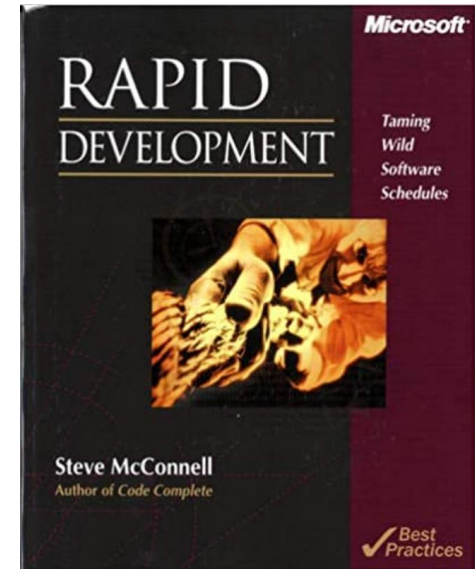
# ¿Qué ciclos de vida conocen?

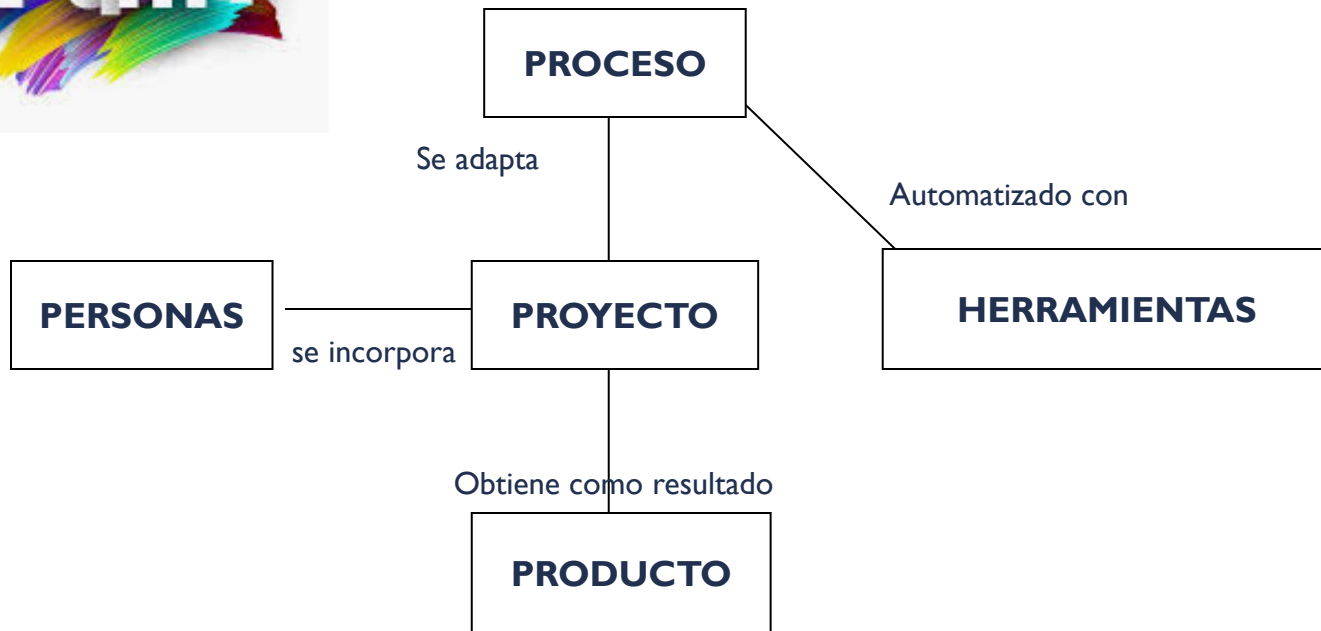


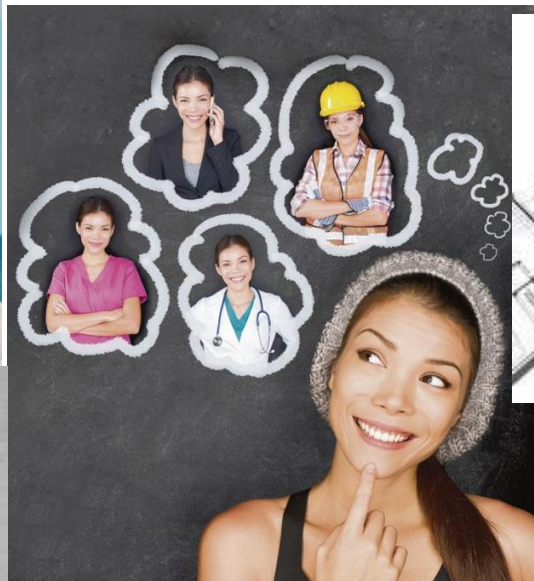
¿Qué relación hay entre procesos de desarrollo y ciclos de vida?

# ¿De ciclos de vida, hay más?

Si! Capítulo 7 de Desarrollos de proyectos informáticos (Rapid Development) de Mcconell







¿QUÉ ES UN PROYECTO?





# CARACTERÍSTICAS: ORIENTACIÓN A OBJETIVOS

- Los proyectos están dirigidos a obtener resultados y ello se refleja a través de objetivos.
- Los objetivos guían al proyecto
- Los objetivos no deben ser ambiguos
- Un objetivo claro no alcanza...debe ser también alcanzable.



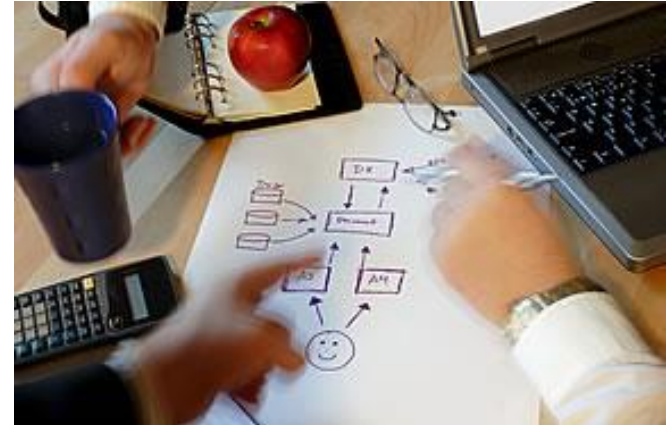
## CARACTERÍSTICAS: DURACIÓN LIMITADA

- Los proyectos son temporarios, cuando se alcanza el/los objetivo/s, el proyecto termina.
- Una línea de producción no es un proyecto.



# CARACTERÍSTICAS: TAREAS INTERRELACIONADAS BASADAS EN ESFUERZOS Y RECURSOS

- Complejidad sistémica de los problemas.



# CARACTERÍSTICAS: SON ÚNICOS

- Todos los proyectos por similares que sean tienen características que los hacen únicos.
- “No soy tan popular para ser diferente”

*Homero Simpson*



---

“PLANNING IS EVERYTHING. PLANS ARE NOTHING.”  
—FIELD MARSHAL HELMUTH GRAF VON MOLTKE



# ¿QUÉ ES LA ADMINISTRACIÓN DE PROYECTOS?

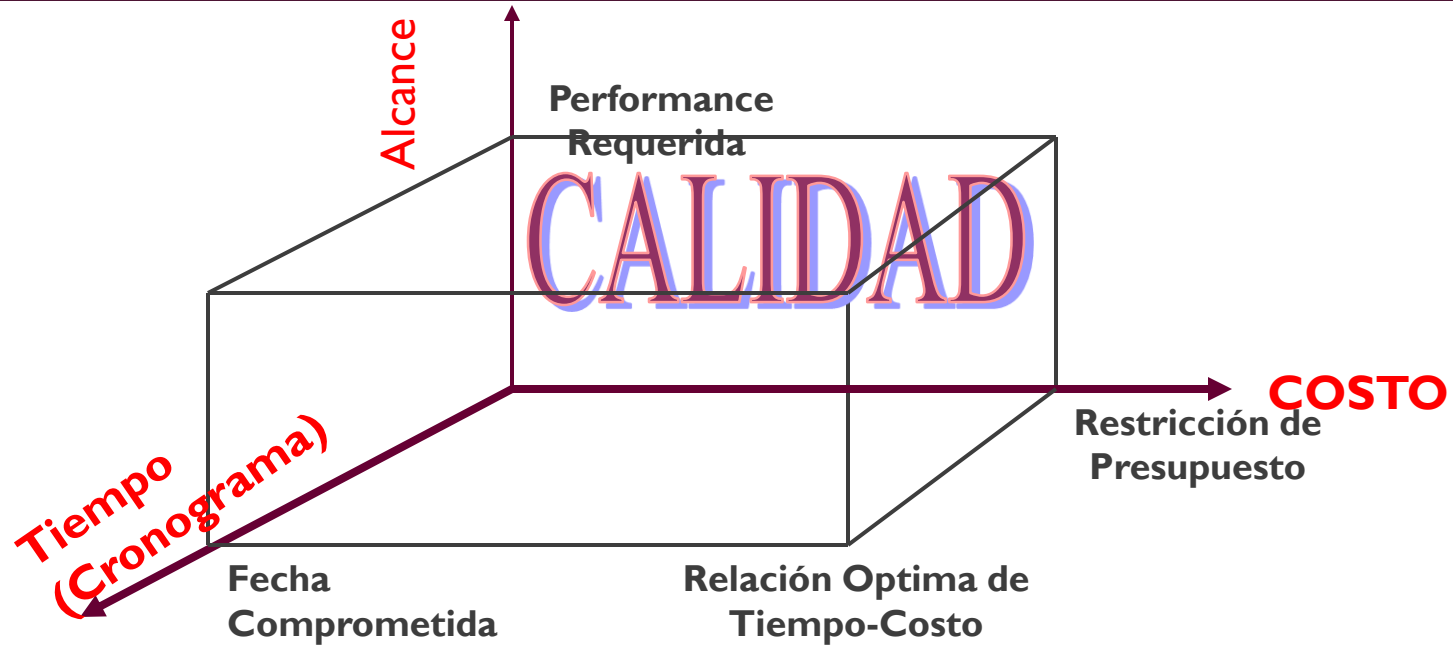
- “...tener el trabajo hecho...” en tiempo, con el presupuesto acordado y habiendo satisfecho las especificaciones o requerimientos.
- Mas académicamente... administración de proyectos es la aplicación de conocimientos, habilidades, herramientas y técnicas a las actividades del proyecto para satisfacer los requerimientos del proyecto.
- Administrar un proyecto incluye:
  - Identificar los requerimientos
  - Establecer objetivos claros y alcanzables
  - Adaptar las especificaciones, planes y el enfoque a los diferentes intereses de los involucrados (stakeholders).

## “LA RESTRICCIÓN TRIPLE” (THE TRIPLE CONSTRAIN)

- Objetivos de proyecto: que está el proyecto tratando de alcanzar?
- Tiempo: cuánto tiempo debería llevar completarlo?
- Costos: cuánto debería costar?
- El balance de estos tres factores afecta directamente la calidad del proyecto  
“proyectos de alta calidad entregan el producto requerido, el servicio o resultado, satisfaciendo los objetivos en el tiempo estipulado y con el presupuesto planificado.”

Es responsabilidad del Líder de proyecto balancear estos tres objetivos (que a menudo compiten entre ellos)

# LA RESTRICCIÓN TRIPLE



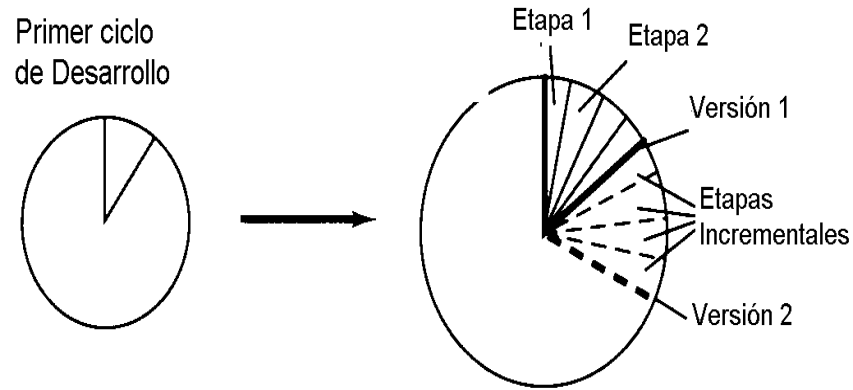


## TRIPLE RESTRICCIÓN

- Pensemos ejemplos de proyectos de desarrollo de software que tengan cambios y que variable o variables de la triple restricción se verán afectas

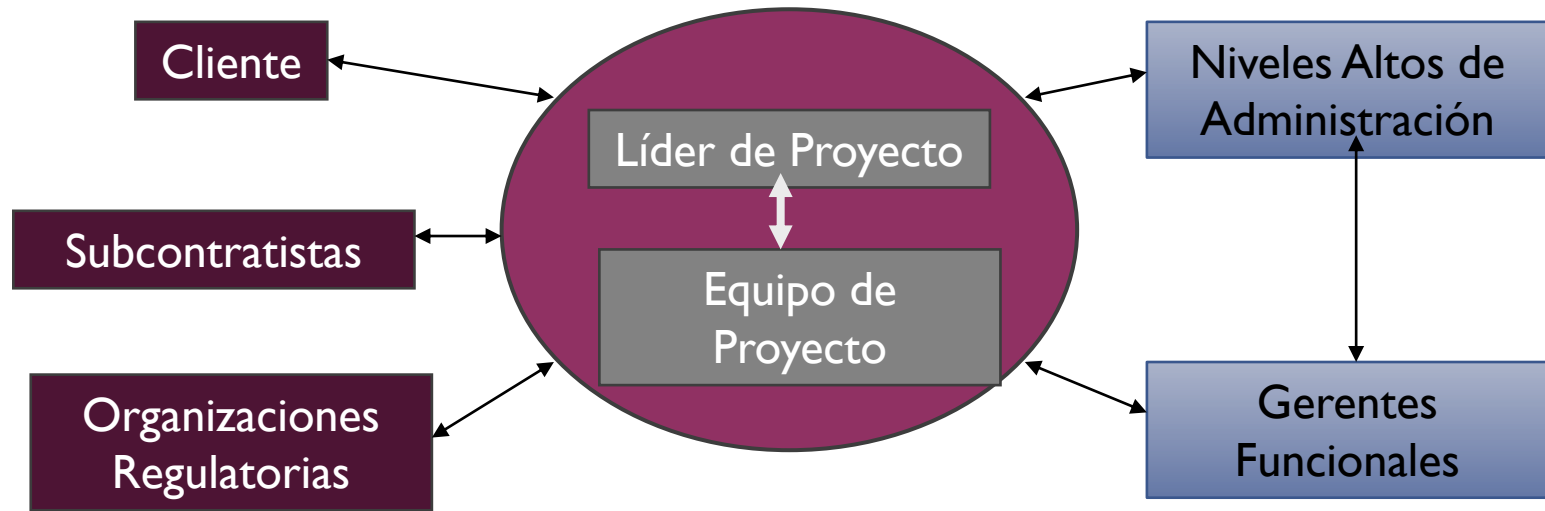


# El Desarrollo de Software



Producto Software: Cada nueva versión es desarrollada **incrementalmente** en una serie de pasos

## ROL DEL LÍDER DE PROYECTO / EQUIPO





EQUIPO DE PROYECTO

# ¿Qué es un equipo de Proyecto?

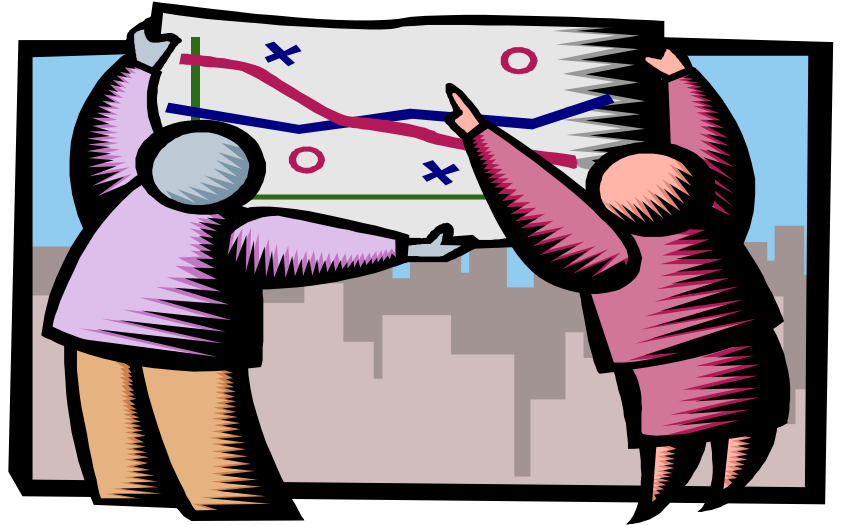
Un grupo de personas comprometidas en alcanzar un conjunto de objetivos de los cuales se sienten mutuamente responsables.

## Características de un equipo de proyecto

- Diversos conocimientos y habilidades
- Posibilidad de trabajar juntos efectivamente / desarrollar sinergia
- Usualmente es un grupo pequeño
- Tienen sentido de responsabilidad como una unidad

## ¿QUÉ ES EL PLAN DE PROYECTO?

**Un plan es a un  
proyecto lo que  
una hoja de  
ruta a un viaje**



# ¿QUÉ ES UN PLAN DE PROYECTO?

- El plan de proyecto documenta:
  - ¿Qué es lo que hacemos?
  - ¿Cuándo lo hacemos?
  - ¿Cómo lo hacemos?
  - ¿Quién lo va a hacer?

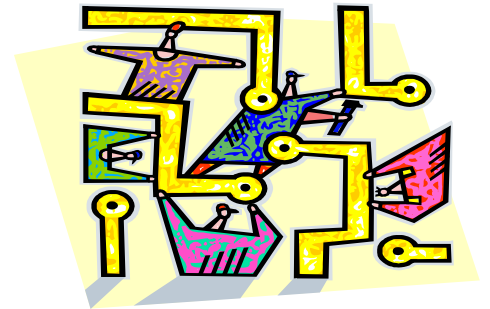


# ¿QUÉ IMPLICA LA PLANIFICACIÓN DE PROYECTOS DE SOFTWARE?

- Definición del Alcance del Proyecto
- Definición de Proceso y Ciclo de Vida
- Estimación
- Gestión de Riesgos
- Asignación de Recursos
- Programación de Proyectos
- Definición de Controles
- Definición de Métricas

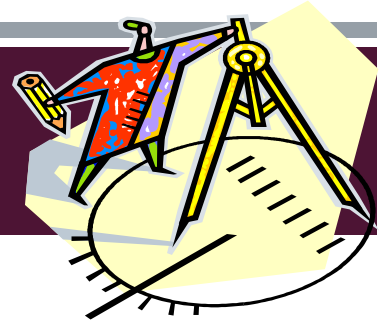
# DEFINICIÓN DEL ALCANCE

- **Alcance del Producto:**
  - Son todas las características que pueden incluirse en un producto o servicio.
- **Alcance del Proyecto:**
  - Es **todo el trabajo** y **solo el trabajo** que debe hacerse para entregar el producto o servicio con todas las características y funciones especificadas.



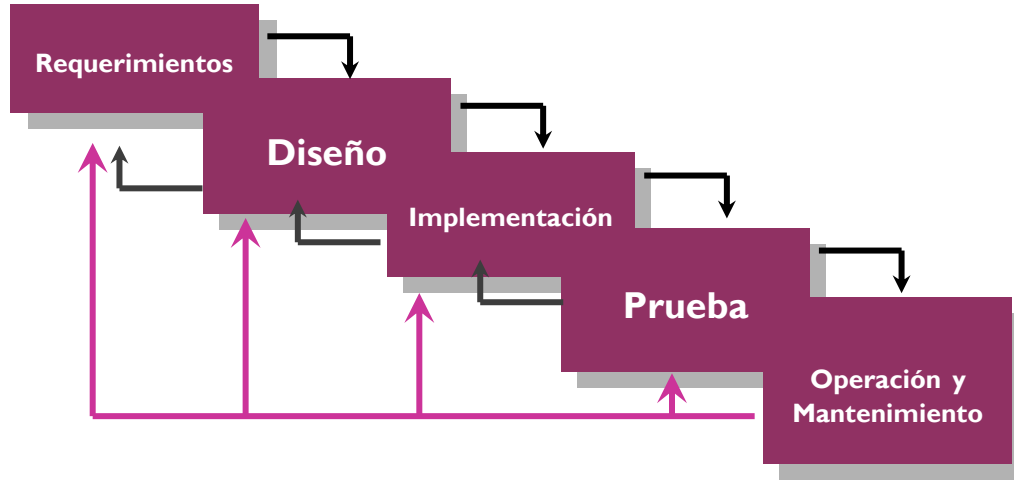


## ALCANCE: ¿CÓMO SE MIDE?



- El cumplimiento del Alcance del Proyecto:
  - Se mide contra el Plan de Proyecto (o Plan de Desarrollo de Software).
- El cumplimiento del Alcance del Producto:
  - Se mide contra la Especificación de Requerimientos.

# DEFINIR UN CICLO DE VIDA



**Workflows  
centrales**

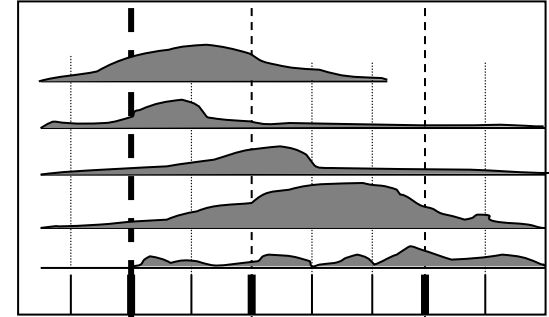
**Requerimientos**

**Análisis**

**Diseño**

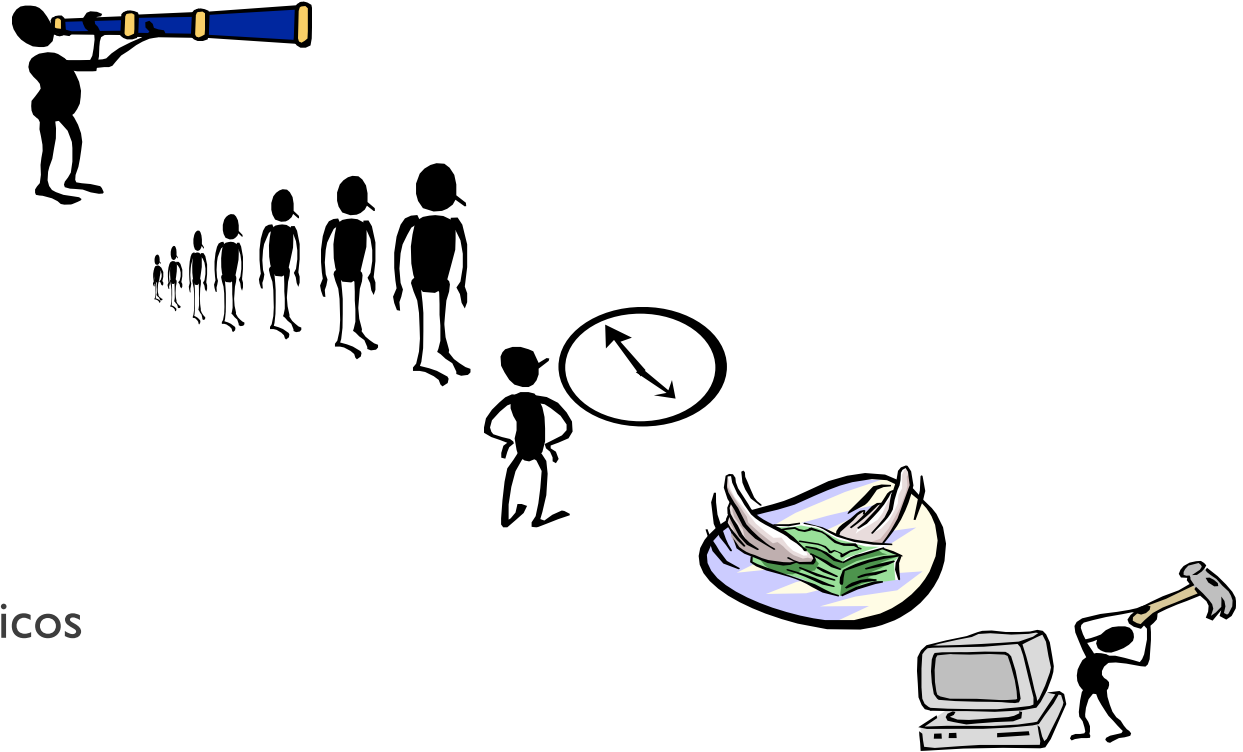
**Implementación**

**Prueba**



# ESTIMACIONES DE SOFTWARE

- Tamaño
- Esfuerzo
- Calendario
- Costo
- Recursos Críticos



# RIESGO....

- Problema esperando para suceder
- Evento que podría comprometer el éxito del proyecto



# GESTIÓN DE RIESGOS





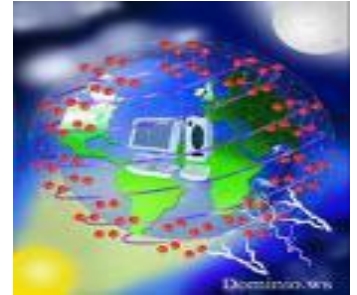
# RIESGOS

RIESGOS TÍPICOS EN  
UN PROYECTO DE  
SOFTWARE

# Métricas de software

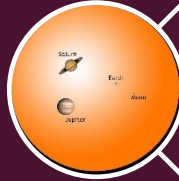
El dominio de las métricas del software se divide en:

- Métricas de proceso.
- Métricas de proyecto.
- Métricas de producto.



Las métricas del proyecto se consolidan para crear métricas de proceso que sean públicas para toda la organización del software.

# Métricas básicas para un proyecto de software



Tamaño del producto



Esfuerzo



Tiempo (Calendario)



Defectos



# EL SUEÑO DEL PIBE...

- **Desarrollador**

1. Esfuerzo
2. Esfuerzo y duración estimada y actual de una tarea.
3. % de cobertura por el unit test
4. Numero y tipo de defectos encontrados en el unit test.
5. Numero y tipo de defectos encontrados en revisión por pares.

- **Organización**

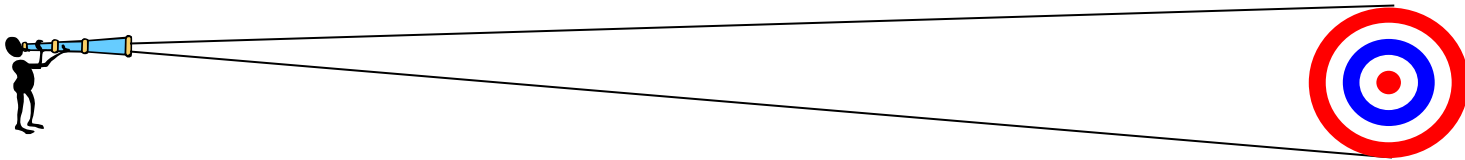
1. Tiempo Calendario
2. Performance actual y planificada de esfuerzo.
3. Performance actual y planificada de presupuesto
4. Precisión de estimaciones en Schedule y esfuerzo
5. Defectos en Release

- **Equipo de Desarrollo**

1. Tamaño del producto
2. Duración estimada y actual entre los hitos más importantes.
3. Niveles de staffing actuales y estimados.
4. Nro. de tareas planificadas y completadas.
5. Distribución del esfuerzo
6. Status de requerimientos.
7. Volatilidad de requerimientos.
8. Nro. de defectos encontrados en la integración y prueba de sistemas.
9. Nro. de defectos encontrados en peer reviews.
10. Status de distribución de defectos.
11. % de test ejecutados

# MANTÉNGALO SIMPLE...

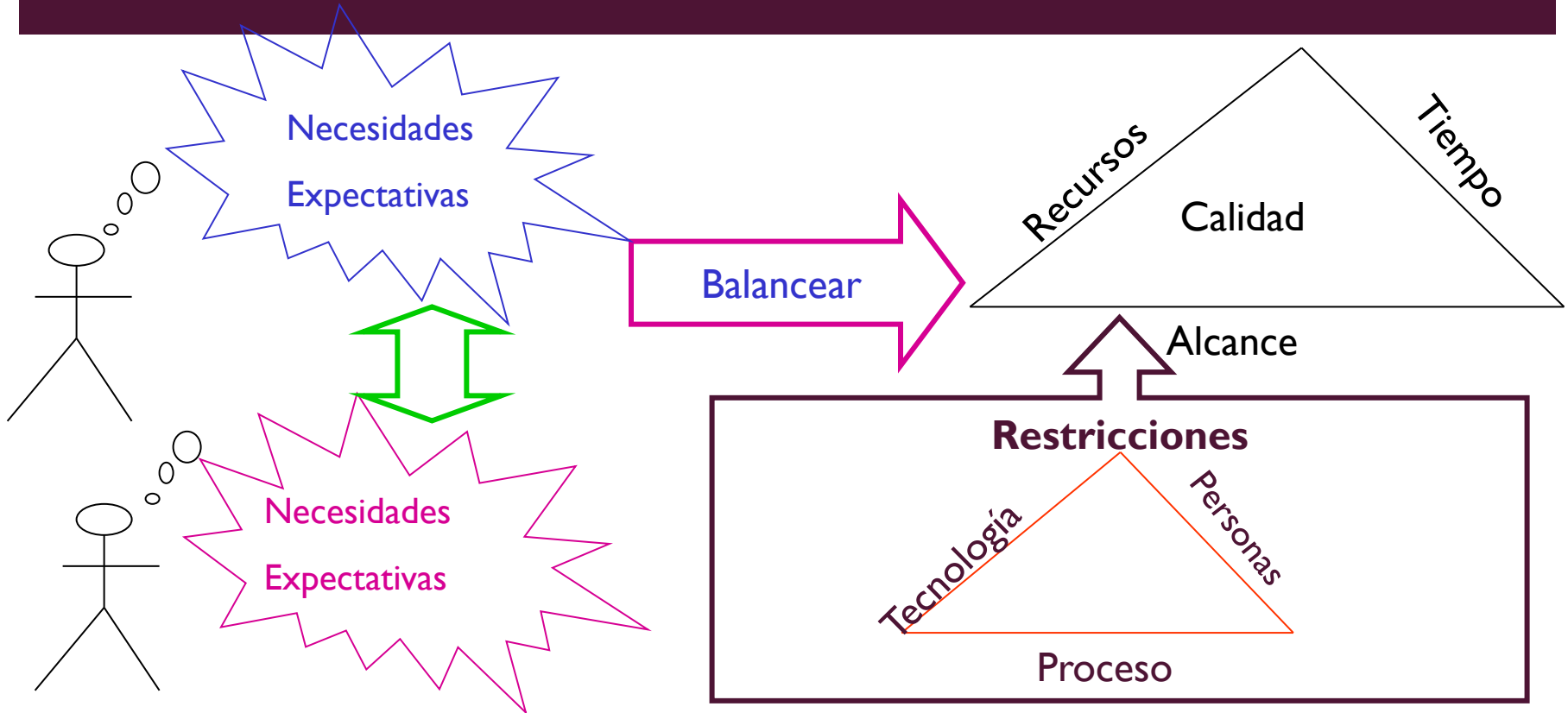
Si estás a millas de distancia de tu destino... no tiene sentido medir en milímetros.



Preguntas:

- ¿Nos da más información que la que tenemos ahora?
- ¿Es esta información de beneficio práctico?
- ¿Nos dice lo que queremos saber?

# ALGO MÁS SOBRE LAS MÉTRICAS...



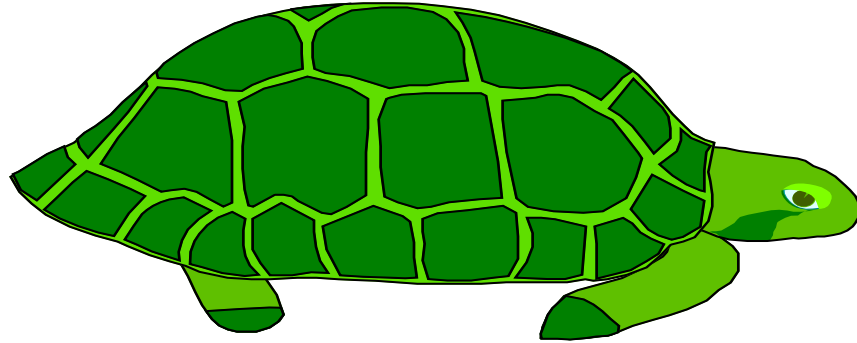
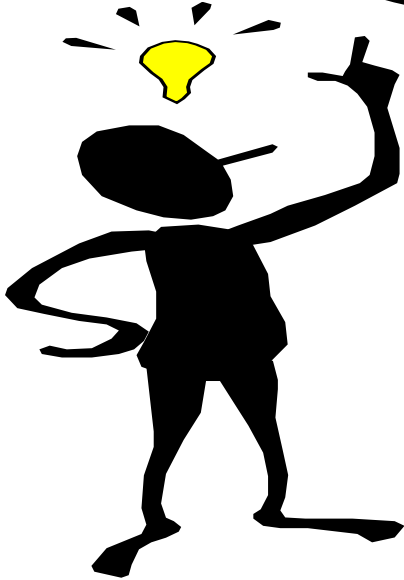


# ***MONITOREO Y CONTROL***



**Como se atrasa un proyecto**

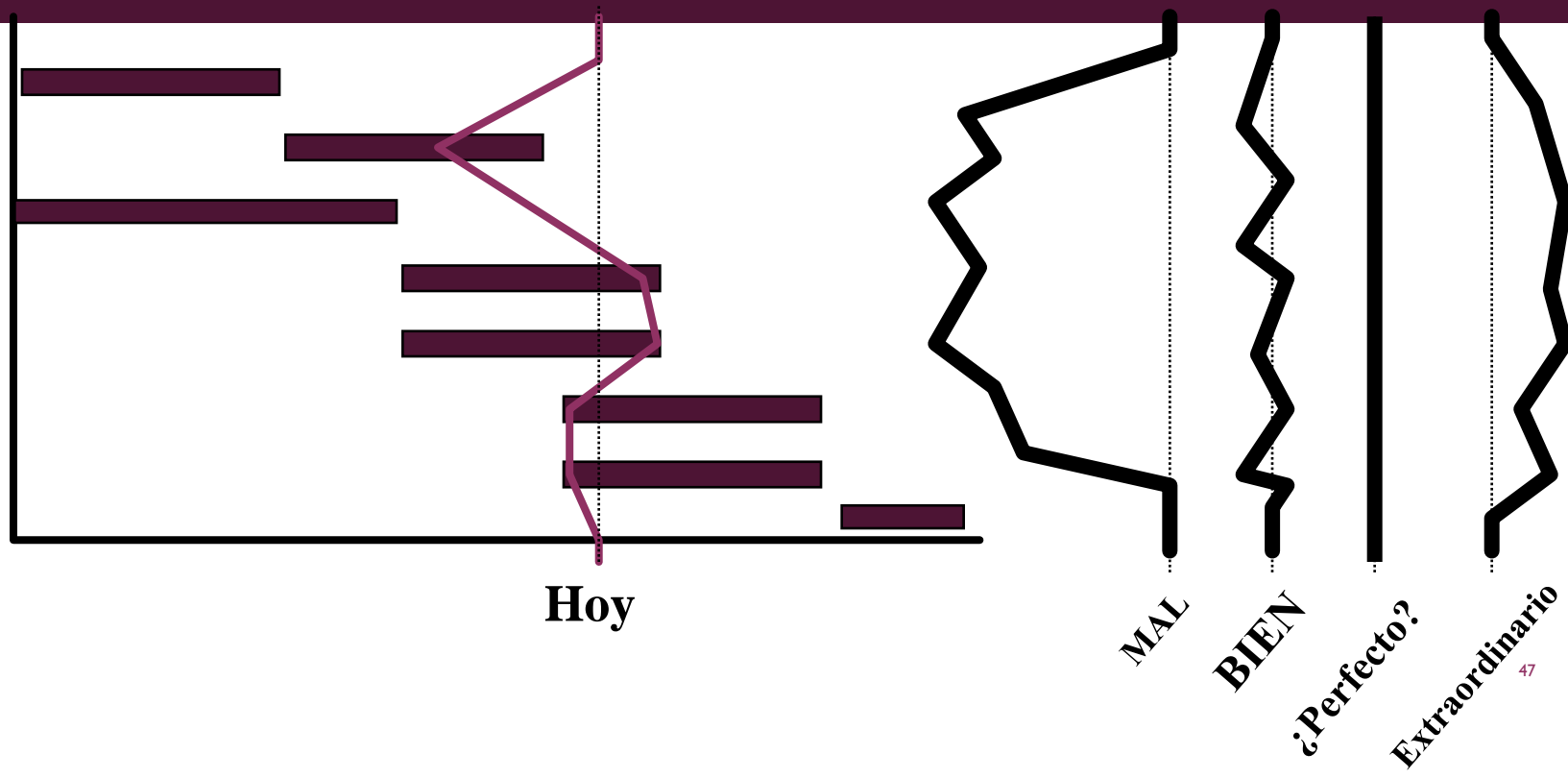
**De a un día por vez**



**Fred Brooks**

**Mythical man months**

# COMPARAR LO PLANIFICADO Y LO REAL



# TRES FACTORES TOP PARA EL ÉXITO DE UN PROYECTO

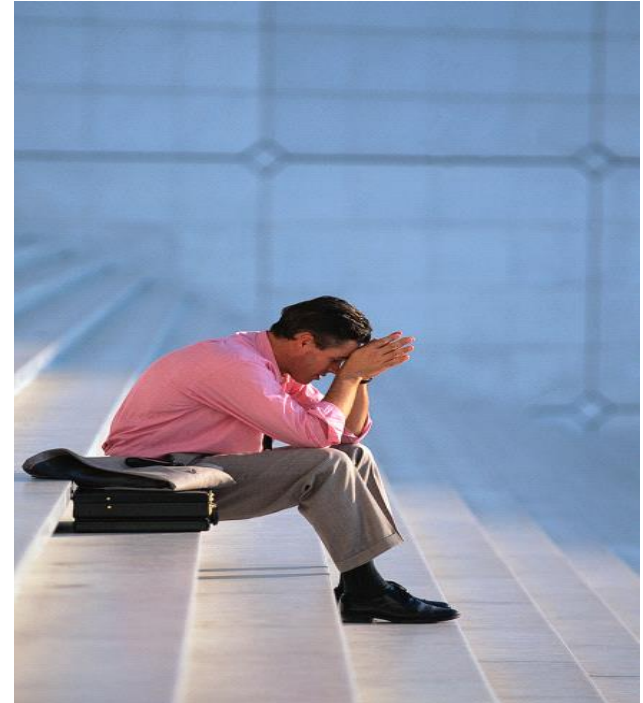
- Monitoreo & Feedback
- Tener una misión/objetivo claro
- Comunicación





# CAUSAS DE FRACASOS EN PROYECTOS

- Fallas al definir el problema
- Planificar basado en datos insuficientes
- La planificación la hizo el grupo de planificaciones
- No hay seguimiento del plan de proyecto
- Plan de proyecto pobre en detalles
- Planificación de recursos inadecuada
- Las estimaciones se basaron en “supuestos” sin consultar datos históricos
- Nadie estaba a cargo





---

¿PARA QUÉ SE UTILIZAN LAS  
ESTIMACIONES, LAS  
MÉTRICAS Y LOS RIESGOS?

¿EN QUÉ MOMENTOS DEL  
PROYECTO SE UTILIZAN?

¿CÓMO SE CORRIGEN  
DESVIACIONES EN UN  
PROYECTO?

PREGUNTAS?

