



INICIO
GRABACIÓN



SANJOSÉ
FUNDACIÓN DE EDUCACIÓN SUPERIOR



Arquitectura De TI

Wilson Cárdenas Cr.
Fundación de educación superior San Jose
Semana 7

A photograph of a software development team in a modern office. A man and a woman are leaning over a desk, looking at a laptop. The man is pointing at the screen. In the background, another person is visible, also working. The image is overlaid with a large pink circle and a smaller grey circle, both containing the text 'DESARROLLO DE SOFTWARE'.

DESARROLLO DE SOFTWARE

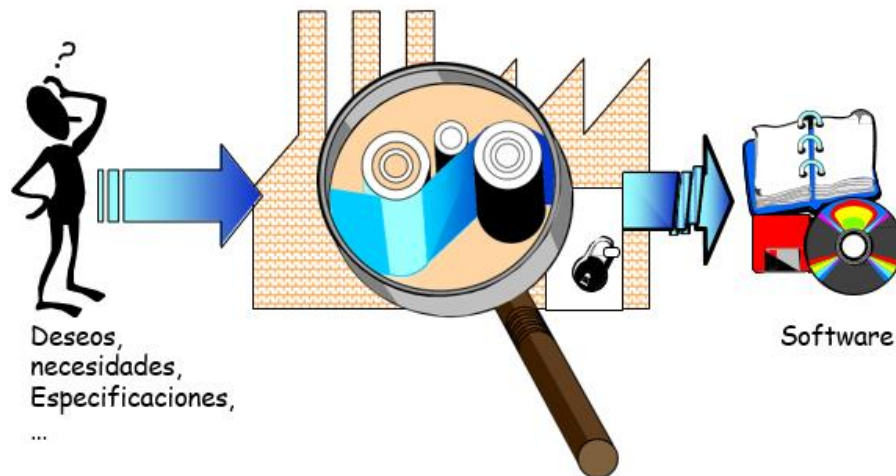


Introducción al proceso de desarrollo



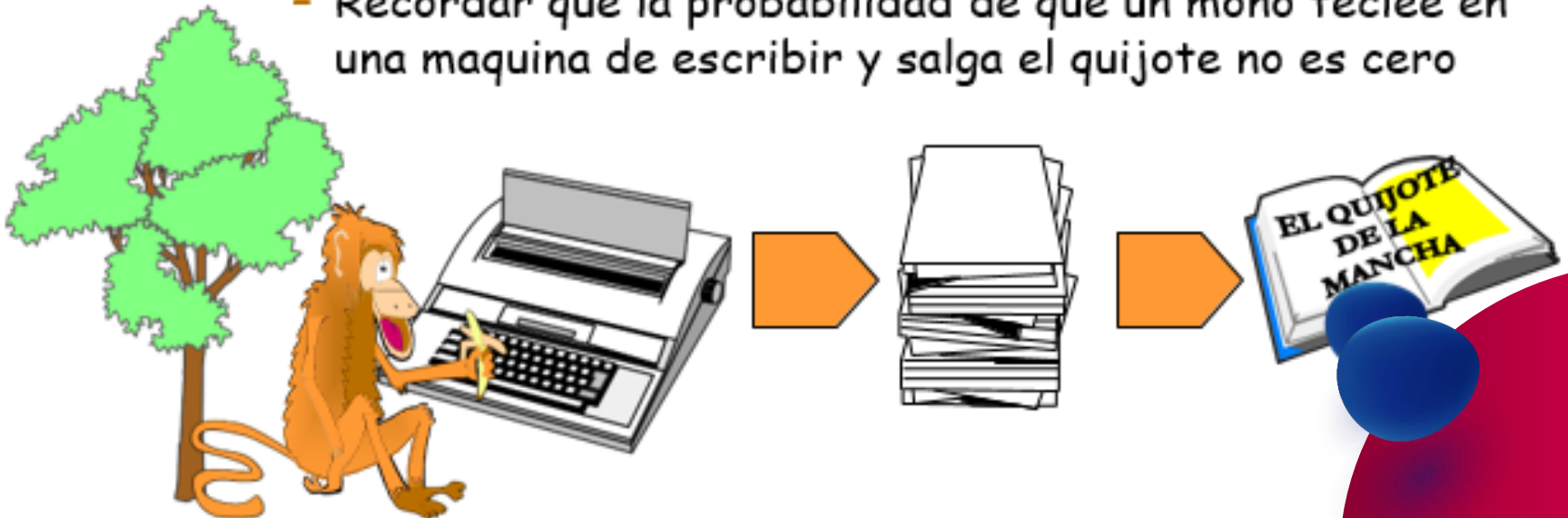
En general, cuando las personas abordan el desarrollo de cualquier proyecto evolucionan desde ideas abstractas hacia concreciones realizables.

*¿Que queremos decir con
proceso de desarrollo?*



Es extraño que alguien diga:

- "Me puse a programar y me salió una contabilidad"
- Algunas aplicaciones da la impresión de que...
 - Recordar que la probabilidad de que un mono teclee en una maquina de escribir y salga el quijote no es cero



Normalmente el planteamiento es:

- Vamos a desarrollar una contabilidad,
 - Hacer la especificación.
- en Visual BASIC y Access,
 - Realizar el diseño.
- y se codifica la aplicación.



El grado de precisión con que se lleve a cabo cada paso influirá en el resultado



- ¿Me podrías indicar, por favor, hacia dónde tengo que ir?
- Eso depende de adónde quieras llegar, contesto el gato.
- A mí no me importa demasiado adónde...- empezó a explicar Alicia.
- En ese caso, da igual cualquier dirección - interrumpió el Gato.
- ...Siempre que llegue a alguna parte - terminó Alicia a modo de explicación.
- Basta que empieces a andar - le aseguro el Gato-, dando un paso tras otro.
 - Alicia en el país de las maravillas

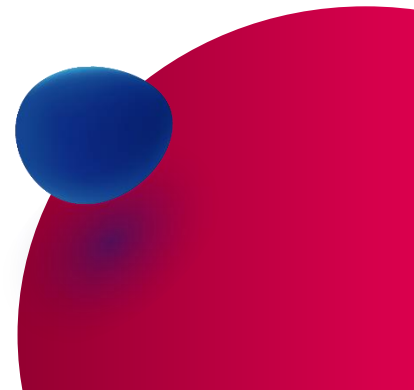




Ingeniería e Ingeniería del Software




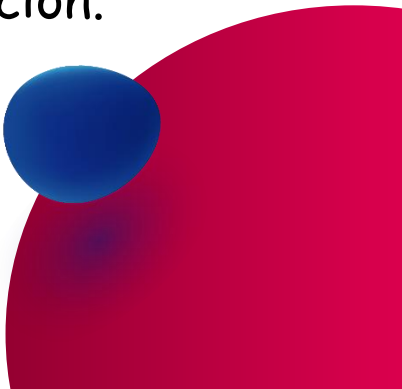
- ¿Donde nos encontramos?
- ¿Que hace un Ingeniero?
- Definiciones de Ingeniería del software
- Técnicas básicas utilizadas históricamente





¿Donde nos encontramos?





- Con Problemas, tendremos que disponer de:
 - Técnicas de desarrollo que **minimicen la complejidad** de un sistema software.
 - Métodos y conceptos que permitan al productor y al cliente **explorar la naturaleza del sistema** software lo antes posible.
 - Técnicas que **minimicen los efectos** devastadores de las **modificaciones** durante el desarrollo y la explotación.
- 
- 



¿Que hace un Ingeniero?



- Construir artefactos dentro de unas limitaciones de coste, utilizando el conocimiento y las teorías de la ciencia sobre la cual se basa el artefacto.
 - Si se recurre a la Real Academia Española, un ingeniero es una persona con titulación universitaria superior que la capacita para ejercer la ingeniería en alguna de sus ramas; pero en realidad los ingenieros son algo más que lo que dice un título oficial. Son los grandes inventores; aquellos que diseñan, construyen y mantienen sistemas y estructuras complejos, desde edificios hasta software.
- 
- 



Definición de la ingeniería del software (NATO '68)

- La ingeniería del software es el establecimiento y uso de principios de ingeniería razonables con el objetivo de obtener software económicamente, que sea de confianza y trabaje eficientemente en las máquinas reales.

Definición (R.E. Fairley 1985)

- Hemos definido la Ingeniería del Software como la disciplina tecnológica concerniente a la producción y mantenimiento sistemáticos de productos software que son desarrollados y modificados en el tiempo y con los costes estimados...
- Además, la Ingeniería del software tiene que ver con cuestiones de gestión que caen fuera del dominio de la programación tradicional.

Características de la ingeniería del software (Van Vliet 1993)



Construcción de programas grandes

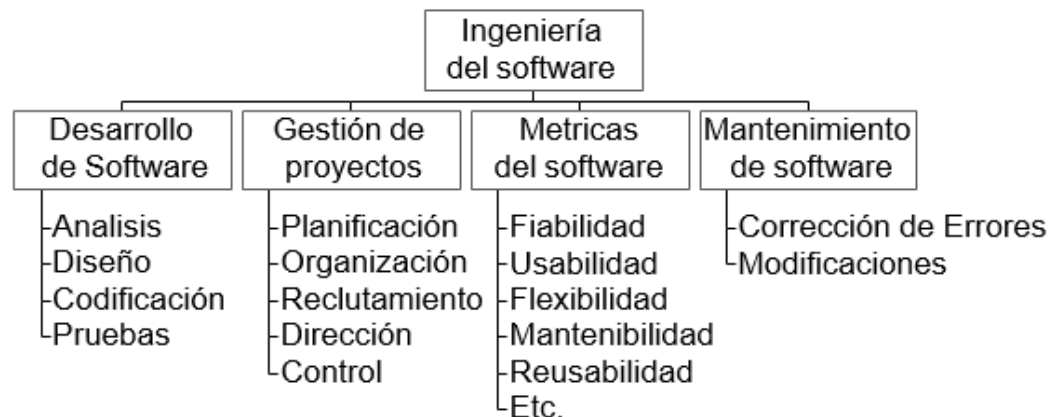
Controlar la complejidad

Cooperación entre las personas implicadas

Evolución del software

Eficiencia en el desarrollo

Soporte real a los usuarios

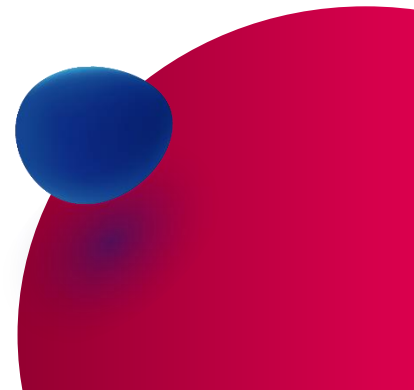




Técnicas básicas usadas en las ingenierías



- Históricamente se han utilizado técnicas como:
 - El modelado
 - División del Producto
 - División del Proceso
- En principio se deberían utilizar estas técnicas, también en informática.

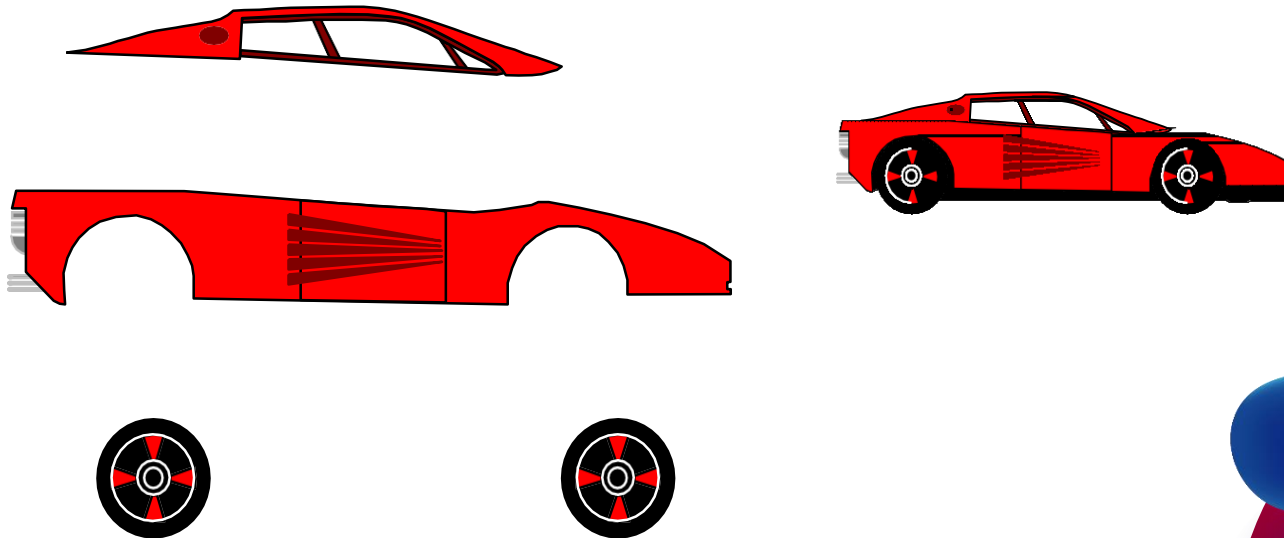


El modelado.



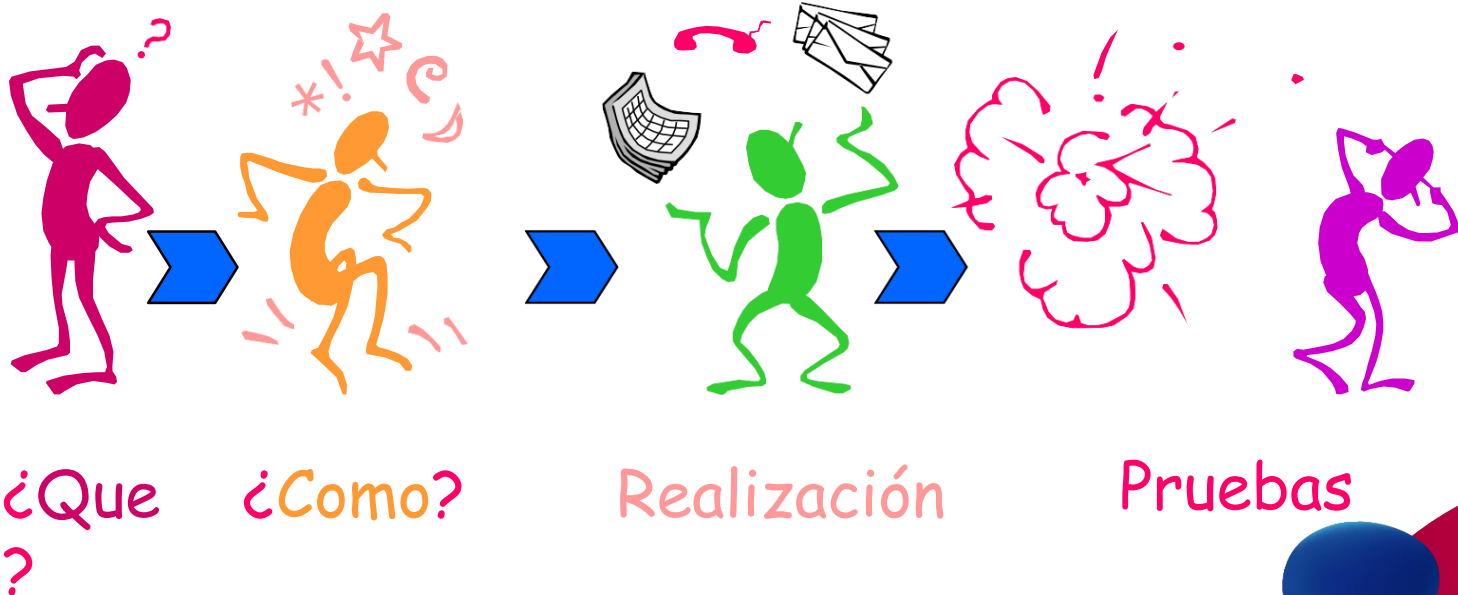
Simplificación del objeto en el mundo real, pero que es suficientemente realista como para dar una idea de lo que ocurrirá en la realidad y usarse como base del desarrollo.

División del producto: Se fracciona el producto de modo que cada fragmento lo puede realizar un miembro del grupo de desarrollo.

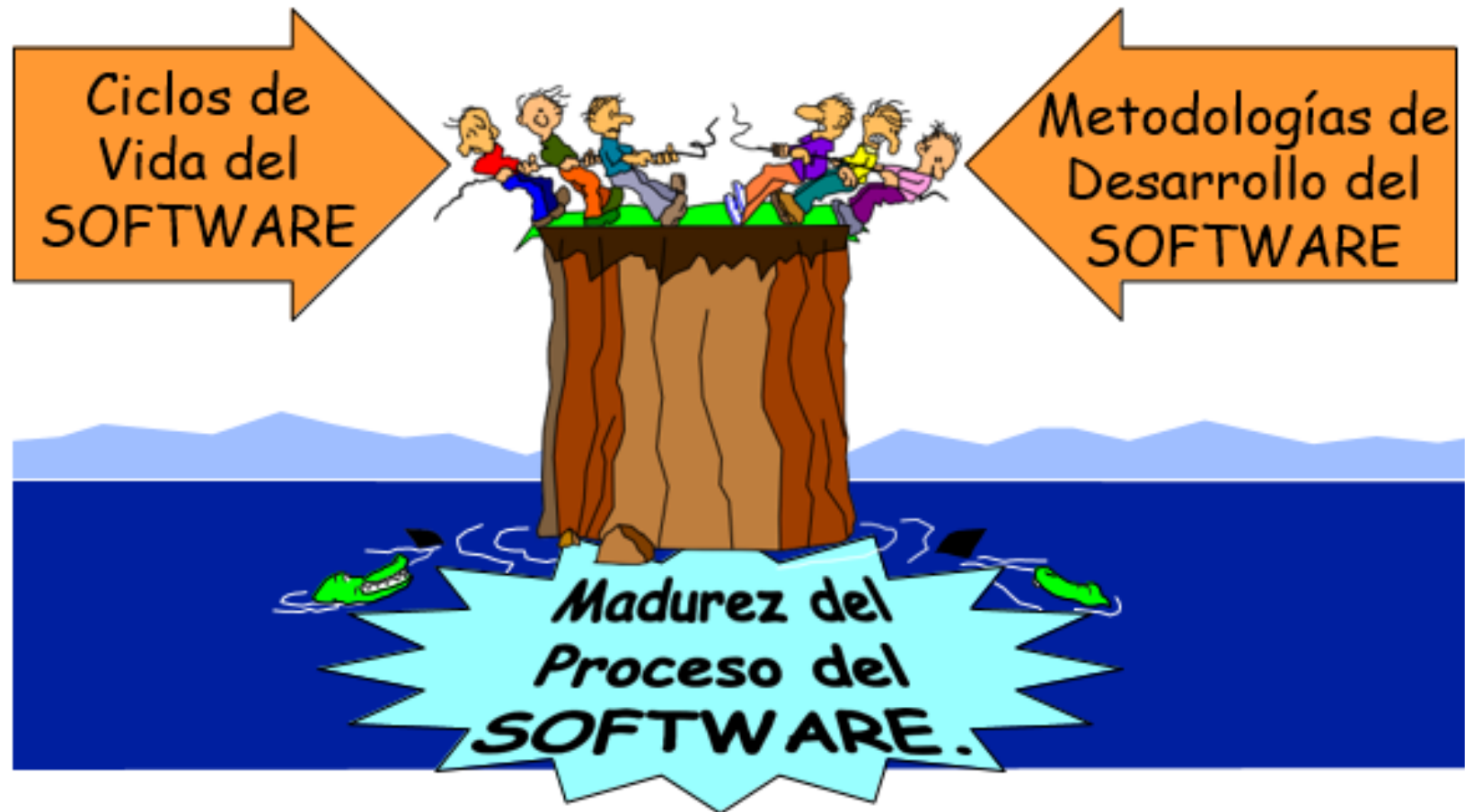


División del Proceso.

- Implica dividir el desarrollo del artefacto por fases. Normalmente se habla de especificación, diseño y fabricación.



En el desarrollo de software nos encontramos con la siguiente situación







Ciclo de Vida del Software



- Consiste en determinar:
 - las fases productivas de un proyecto,
 - los objetivos de cada fase productiva, y
 - los productos obtenidos en cada una de estas fases así como sus características.

Ciclos de Vida del Software

- Se han propuesto muchos ciclos de vida para el desarrollo del software, pero estos son los más representativos:
 - Poner la cola al burro.
 - Ciclo de vida clásico o en cascada.
 - Construcción rápida de Prototipos Desechables
 - Incremental
 - Evolución de prototipos
 - Reutilización de Software
 - Síntesis automática de software
 - En espiral.
- 
- 

Ciclo de vida clásico o en cascada.

- La Versión Ideal (Perfecta)
- El Modelo en V
- El Helado de Cucurucho
- El Modelo Real
- Propuesta de Yourdon



La Versión Ideal





El Modelo en V

Identificación
de Necesidades

Explotación

Especificación
Esencial

Validación

Especificación
Física

Empaquetado

Diseño

Integración

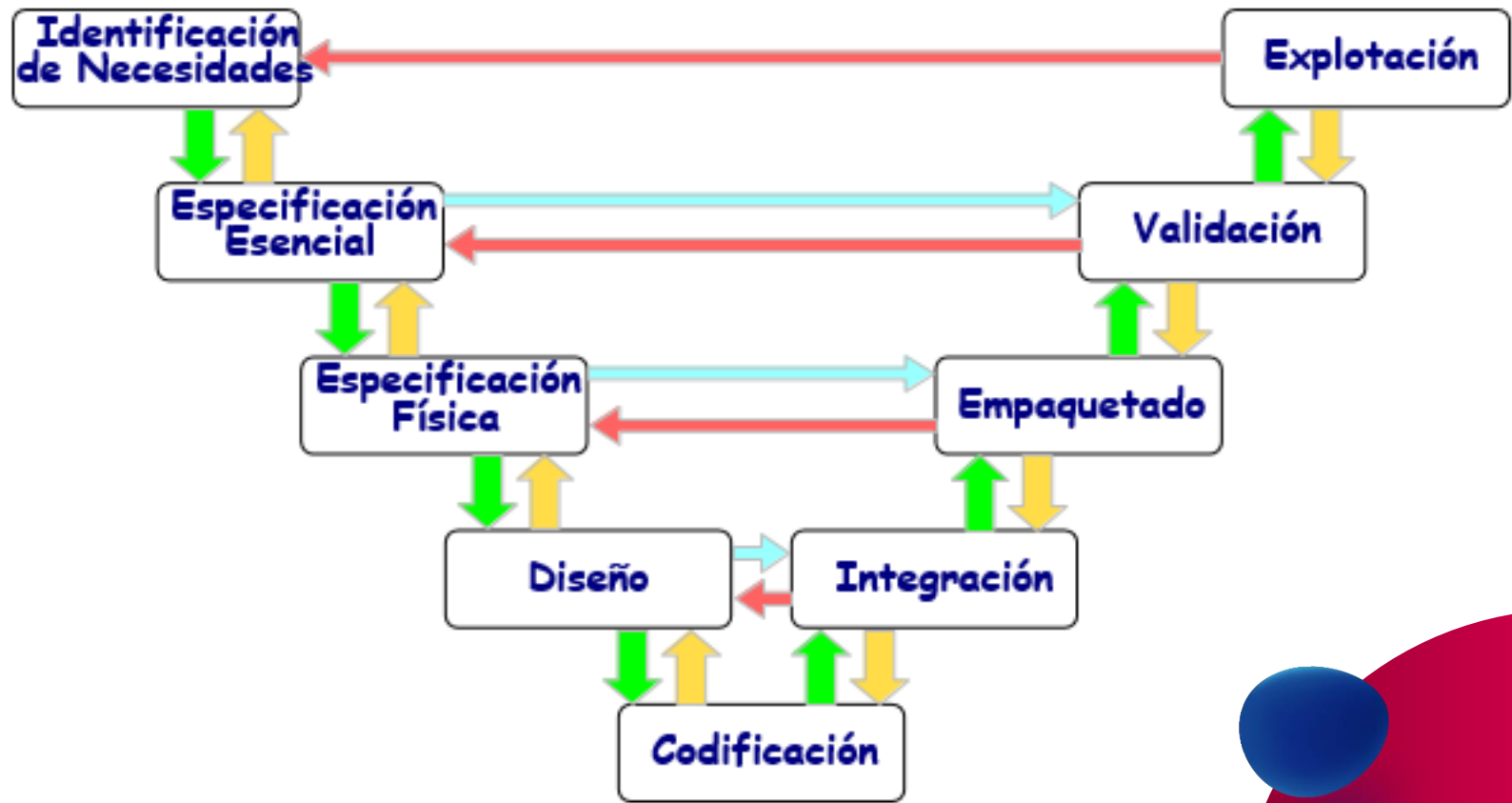
Codificación



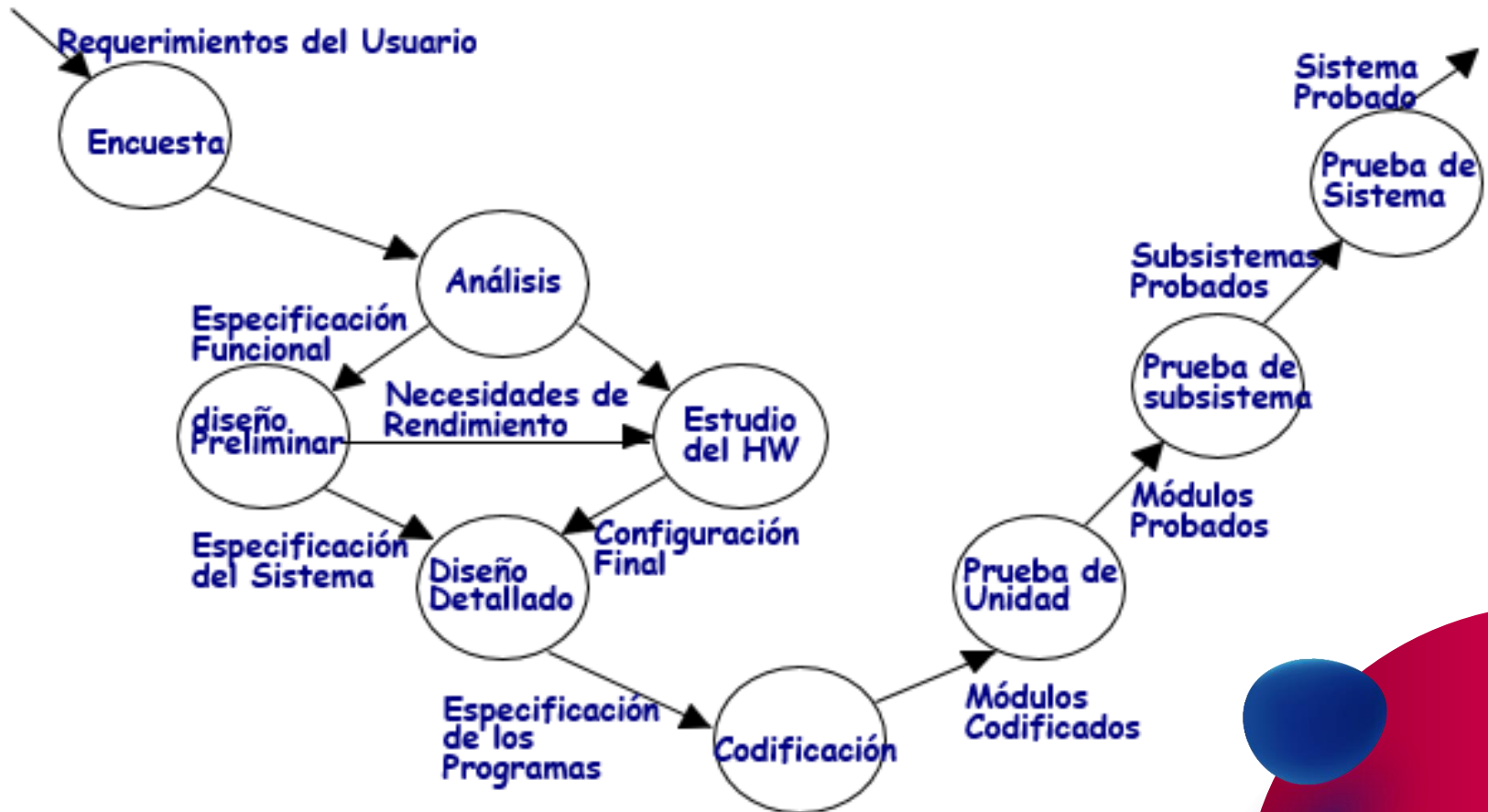
El Helado de Cucurucho



El Modelo Real



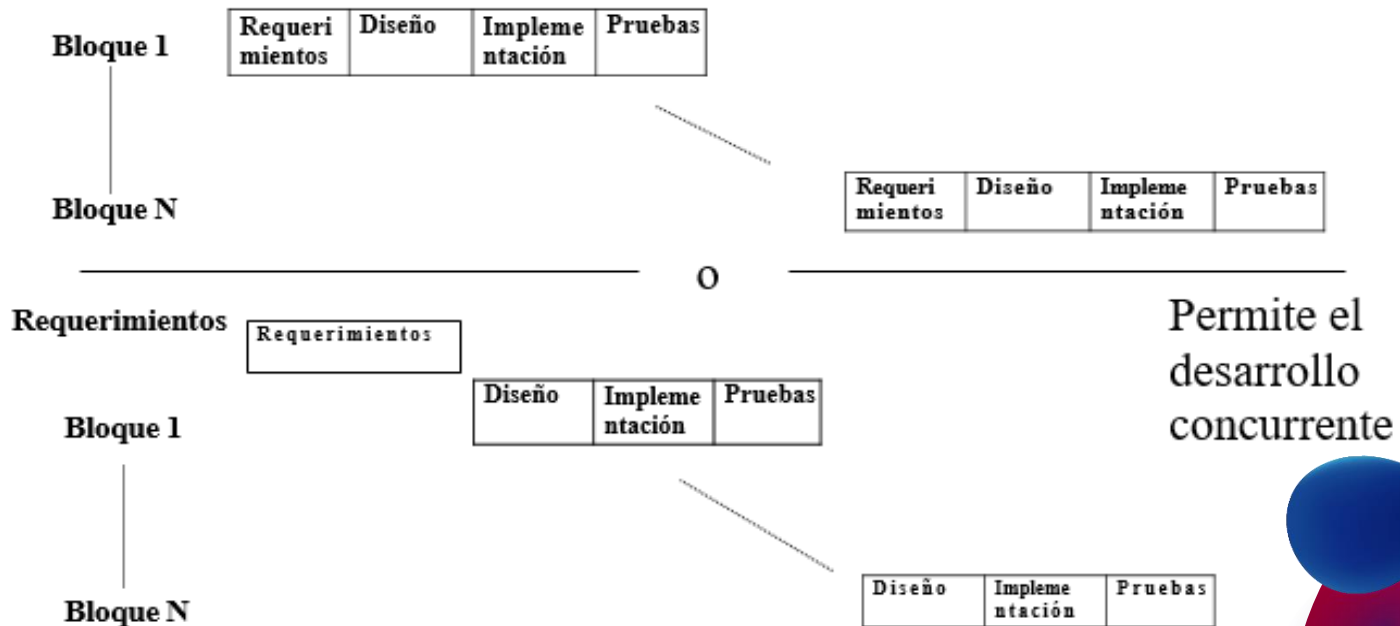
Propuesta de Yourdon



Existen dos clases de prototipos



- De INTERFACE.
- Usualmente un modelo de papel o sobre PC en el que se muestran pantallas y listados.
- De COMPORTAMIENTO:
- En anchura. Ofrece todos los menús del sistema y simula débilmente los procesos.
- En profundidad. Cubre funciones que presentan ambigüedades al cliente o a los informáticos.
- Completo pero de baja calidad y rendimiento.



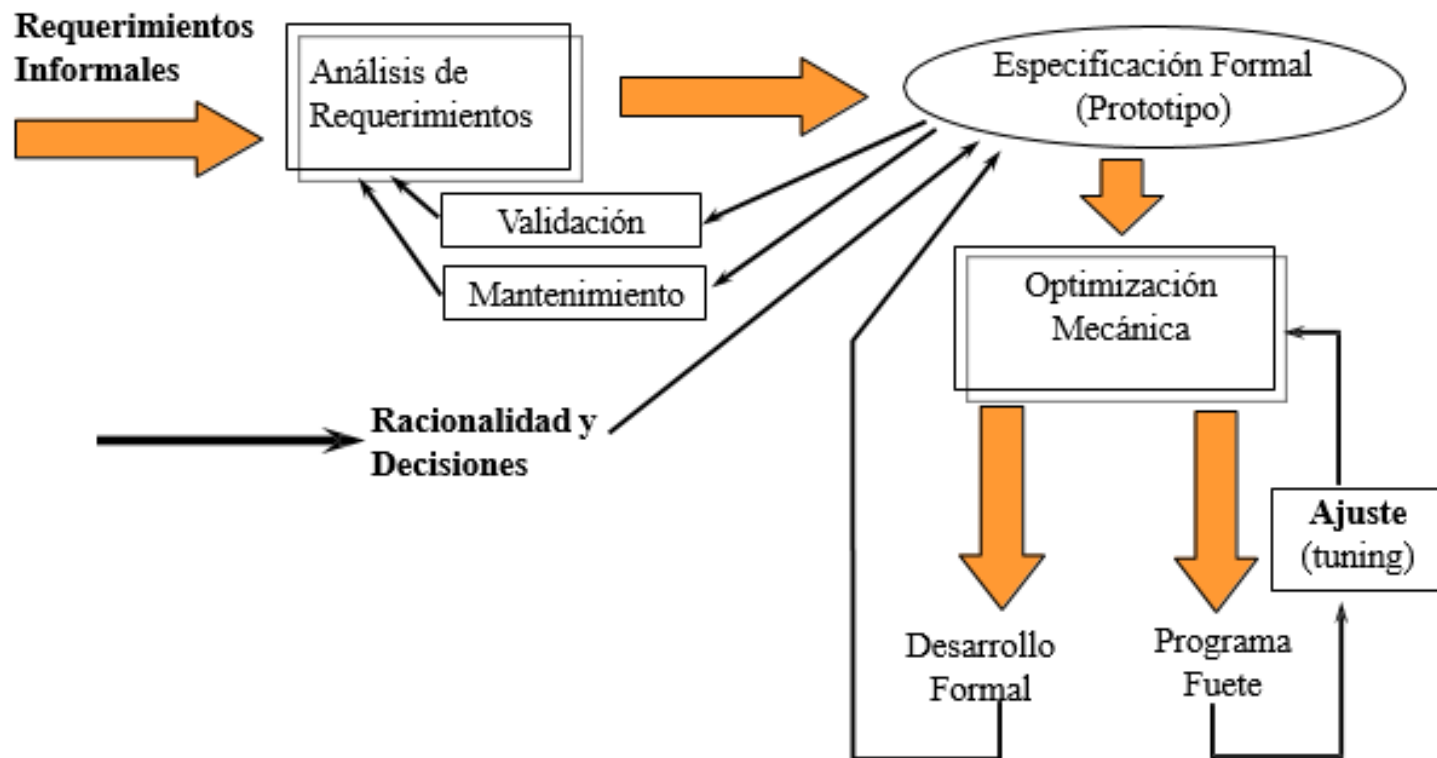
Evolución de Prototipos



Reutilización de Software



- Tiene como objetivos:
- reducir el coste del software.
- Producir sistemas de mayor calidad.
- Se basa en reutilizar Diseños, programas, módulos y datos.
- Es compatible con el prototipo evolutivo.



En espiral.

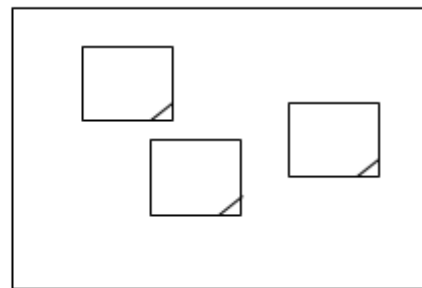




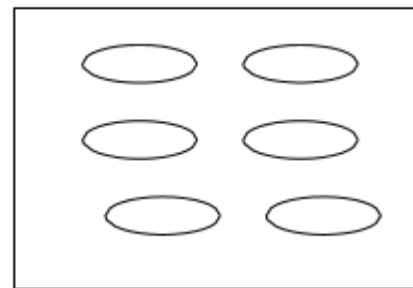
- Métodos informales; Joaquín Lucio-Villegas los clasifica como Quick & Dirty (USA) Match stick box (Europa) Balones p'alante y maricón el último
- Métodos Semiformales: Métodos Estructurados SA/SD (structured analysis & structured design) Métrica Métodos Orientados a Objetos OMT UML
- Métodos Formales

Métodos Estructurados

- Métodos Orientados a la Estructura de los Datos
- Métodos de flujo de datos



PROCESOS

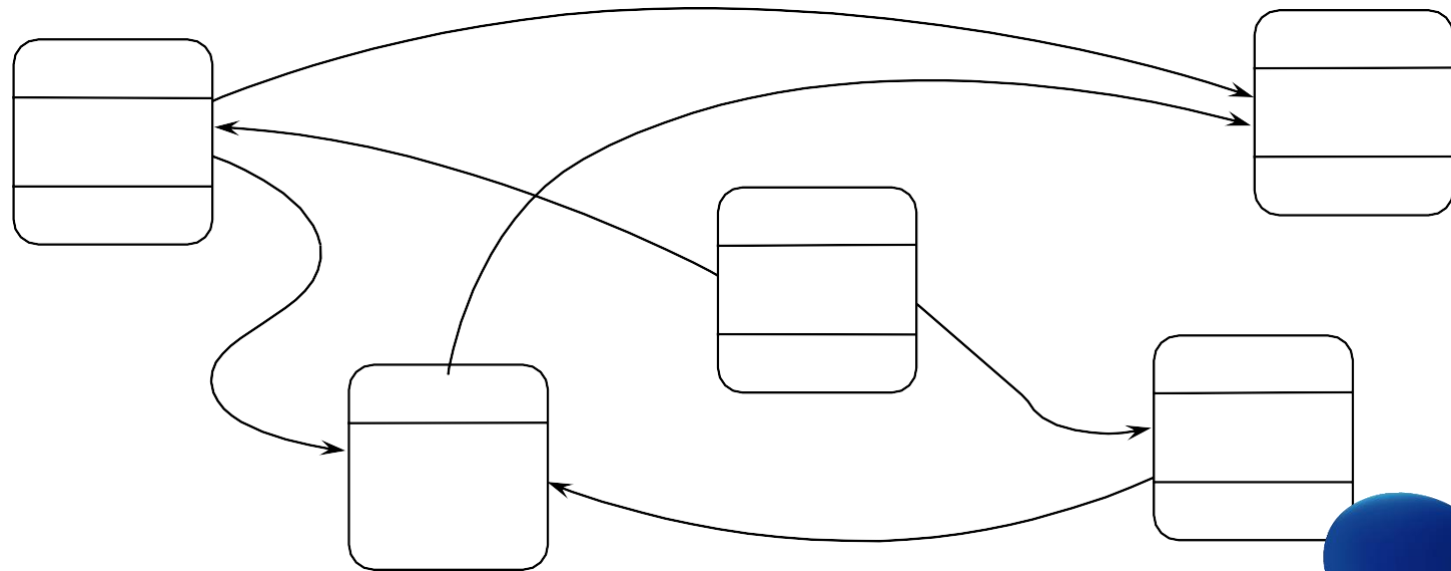


DATOS

Métodos Orientados a Objetos



- Los métodos orientados a objeto describen e implementan los sistemas de información desde un punto de vista ontológico.



Métodos Formales



- Los métodos formales permiten al ingeniero de software especificar, desarrollar y verificar un sistema informático mediante la aplicación de una notación matemática rigurosa.
- Utilizando un lenguaje de especificación formal, un método formal proporciona los medios de especificar un sistema de forma que se aseguren, de forma sistemática, la consistencia, la completitud y la corrección.
- Se suelen basar en notaciones matemáticas similares a las del álgebra de conjuntos y la lógica

Madurez del proceso en la organización de desarrollo


- La misma industria, diferentes niveles de madurez.




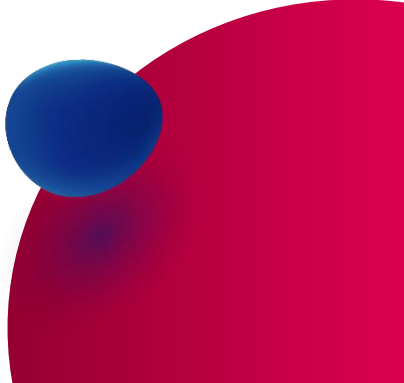


Madurez del proceso en la organización de desarrollo

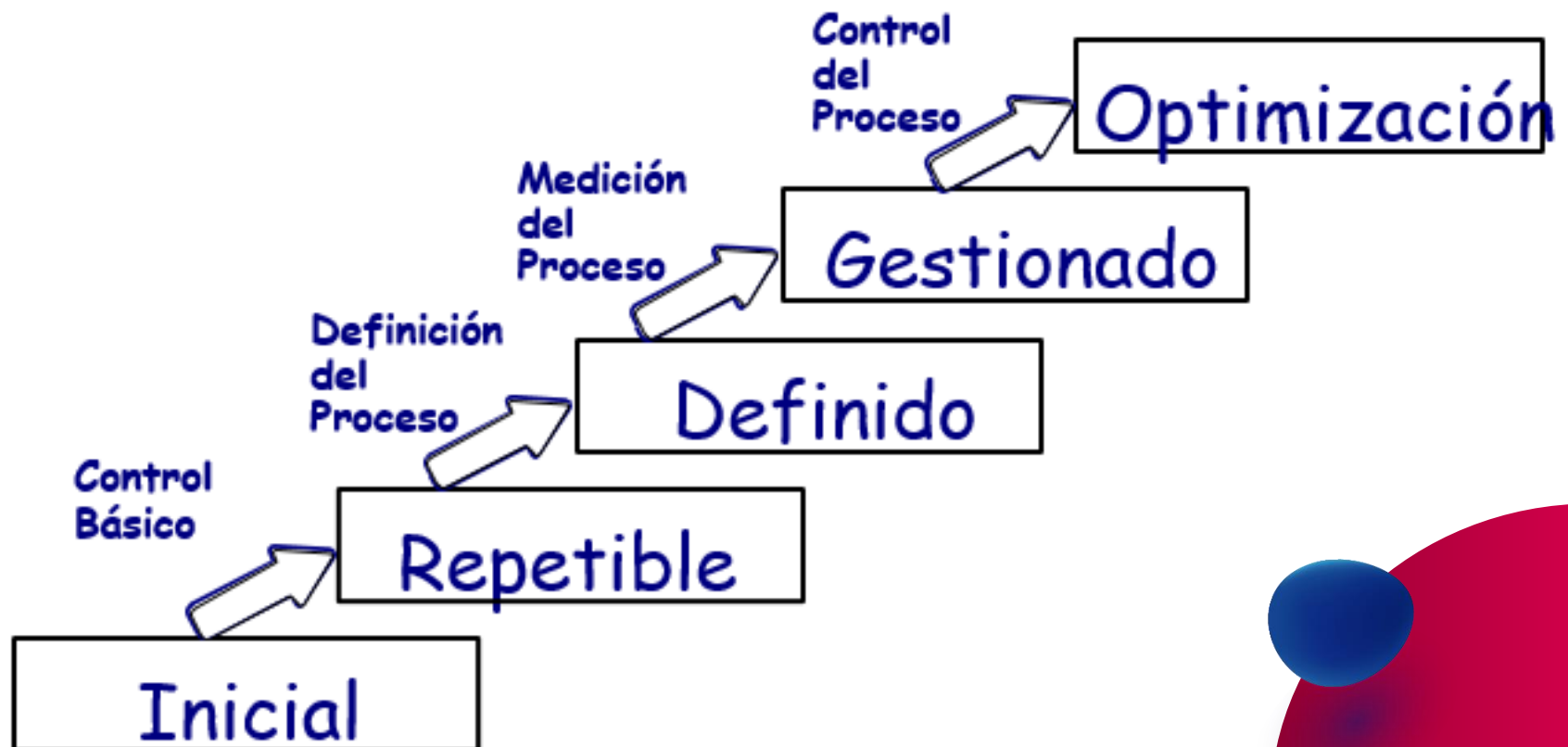


- 
- Hay factores que no quedan claramente reflejados en el ciclo de vida ni en las técnicas de desarrollo.
 - Los factores no estudiados son:
 - El cumplimiento de los plazos de entrega.
 - La calidad (número de errores en el Software).
 - El coste del proyecto.

CMM (Capability Maturity Model)

- Proporciona una Guía sobre como
 - ✓ controlar los procesos:
 - de desarrollo del software.
 - de mantenimiento.
 - ✓ Hacer evolucionar hacia una cultura de:
 - Ingeniería del software.
 - Gestión eficiente.
- 
- 

Evolución de las organizaciones según el CMM



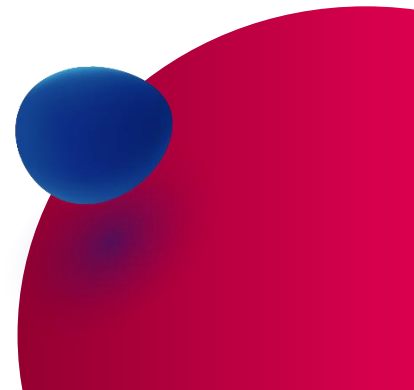


Nivel Inicial

- Según las circunstancias utilizamos un proceso distinto. (algunos caóticos)
- A medida,
- Poco formalizado,
- Uso de herramientas informales.
- Pocos procesos definidos.
- El éxito depende del esfuerzo individual.

Nivel de Repetición

- Se tiene procesos estables de desarrollo, con control estadístico.
- Uso de datos históricos
- Establecimiento de procesos de gestión de proyecto, para hacer seguimiento de:
 - Coste.
 - Planificación.
 - Funcionalidad.





Nivel de Definición


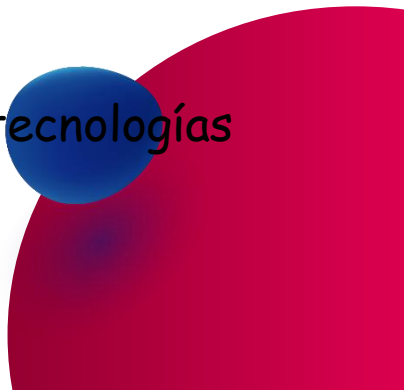
- Proceso de desarrollo perfectamente definido y estandarizado.
- Integrado en la organización.
- Bien documentado.
- Todos los proyectos utilizan una versión documentada y aprobada de proceso.



Nivel de Repetición

- Se tiene procesos estables de desarrollo, con control estadístico.
- Uso de datos históricos
- Establecimiento de procesos de gestión de proyecto, para hacer seguimiento de:
 - Coste.
 - Planificación.
 - Funcionalidad.

Nivel de Optimización

- A través de mediciones del proceso utilizando ideas y tecnologías innovadoras obtenemos:
 - Mejoras en calidad y cantidad.
- 
- 

Conclusiones



- El desarrollo del software y la programación es uno de los pilares fundamentales de la informática y al cual se dedican muchas horas de esfuerzos en empresas, colegios, academias y universidades.
- Conforme a la tecnología va avanzando, van apareciendo nuevas soluciones, nuevas formas de programación, nuevos lenguajes y un sin fin de herramientas que intentan realizar el trabajo del desarrollador un poco mas fácil.
- La programación orientadas a objetos o los compiladores basados en maquinas virtuales (en muchos casos, multiplataforma), también a sus puestos unas renovación en la manera de programar.
- Microsoft como empresa desarrolladora de software, es consciente de lo importante que es hacer buenos desarrollos y lo complicado que es; por eso, intenta aportar las mejores soluciones al mercado. En la actualidad la sociedad se encuentra en una época de transición, que se encamina hacia un nuevo estilo de programación basada en estándares y para ello Microsoft propone la plataforma .NET.



**FIN DE
GRABACIÓN**