

# Desarrollo en espiral

---

El **desarrollo en espiral** es un modelo de ciclo de vida del software definido por primera vez por Barry Boehm en 1988, utilizado generalmente en la Ingeniería de software. Las actividades de este modelo se conforman en una espiral, en la que cada bucle o iteración representa un conjunto de actividades. Las actividades no están fijadas a priori, sino que las siguientes se eligen en función del análisis de riesgo, comenzando por el bucle interior.

## Introducción

La Ingeniería de software, se vale y establece a partir de una serie de modelos que establecen y muestran las distintas etapas y estados por lo que pasa un producto software, desde su concepción inicial, pasando por su desarrollo, puesta en marcha y posterior mantenimiento, hasta la retirada del producto. A estos modelos se les denomina «modelos de ciclo de vida del software». El primer modelo concebido fue el de Royce, más comunmente conocido como *desarrollo en cascada* o *desarrollo lineal secuencial*. Este modelo establece que las diversas actividades que se van realizando al desarrollar un producto software se suceden de forma lineal.

Boehm, autor de diversos artículos de ingeniería del software; modelos de estimación de esfuerzo y tiempo que se consume en hacer productos software; y Modelos de Ciclo de Vida; ideó y promulgó un modelo desde un enfoque distinto al tradicional en Cascada: El Modelo Evolutivo Espiral. Su Modelo de Ciclo de Vida en Espiral tiene en cuenta fuertemente el **riesgo** que aparece a la hora de desarrollar software. Para ello, se comienza mirando las posibles alternativas de desarrollo, se opta por la de riesgo más asumible y se hace un ciclo de la espiral. Si el cliente quiere seguir haciendo mejoras en el software, se vuelve a evaluar las distintas nuevas alternativas y riesgos y se realiza otra vuelta de la espiral, así hasta que llegue un momento en el que el producto software desarrollado sea aceptado y no necesite seguir mejorándose con otro nuevo ciclo.

Este modelo fue propuesto por Boehm en 1988. Básicamente consiste en una serie de ciclos que se repiten en forma de espiral, comenzando desde el centro. Se suele interpretar como que dentro de cada ciclo de la espiral se sigue un Modelo Cascada, pero no necesariamente debe ser así. El Espiral puede verse como un modelo evolutivo que conjuga la naturaleza iterativa del modelo MCP con los aspectos controlados y sistemáticos del Modelo Cascada, con el agregado de gestión de riesgos.

### En cada vuelta o iteración hay que tener en cuenta

**Los Objetivos:** Que necesidad debe cubrir el producto.

**Alternativas:** Las diferentes formas de conseguir los objetivos de forma exitosa, desde diferentes puntos de vista como pueden ser:

1. Características: experiencia del personal, requisitos a cumplir, etc.
2. Formas de gestión del sistema.
3. Riesgo asumido con cada alternativa.

**Desarrollar y Verificar:** Programar y probar el software.

---

**Si el resultado no es el adecuado o se necesita implementar mejoras o funcionalidades**

Se planificarán los siguientes pasos y se comienza un nuevo ciclo de la espiral. La espiral tiene una forma de caracola y se dice que mantiene dos dimensiones, la radial y la angular:

1. **Angular:** Indica el avance del proyecto del software dentro de un ciclo.
2. **Radial:** Indica el aumento del coste del proyecto, ya que con cada nueva iteración se pasa más tiempo desarrollando.

Este sistema es muy utilizado en proyectos grandes y complejos como puede ser, por ejemplo, la creación de un Sistema Operativo.

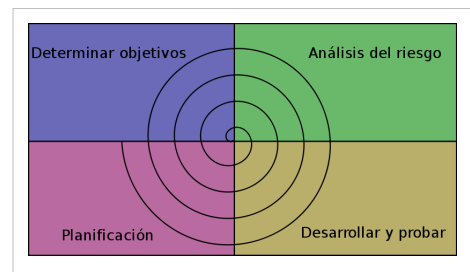
Al ser un modelo de Ciclo de Vida orientado a la gestión de riesgo se dice que uno de los aspectos fundamentales de su éxito radica en que el equipo que lo aplique tenga la necesaria experiencia y habilidad para detectar y catalogar correctamente los riesgos.

**Tareas**

Para cada ciclo habrá cuatro actividades:

**Determinar o fijar objetivos**

- Fijar también los productos definidos a obtener: requerimientos, especificación, manual de usuario.
- Fijar las restricciones.
- Identificación de riesgos del proyecto y estrategias alternativas para evitarlos.



- Hay una cosa que solo se hace una vez: planificación inicial o previa.

**Desarrollar, verificar y validar (probar)**

- Tareas de la actividad propia y de prueba.
- Análisis de alternativas e identificación resolución de riesgos.
- Dependiendo del resultado de la evaluación de los riesgos, se elige un modelo para el desarrollo, el que puede ser cualquiera de los otros existentes, como formal, evolutivo, cascada, etc. Así si por ejemplo si los riesgos en la interfaz de usuario son dominantes, un modelo de desarrollo apropiado podría ser la construcción de prototipos evolutivos. Si lo riesgos de protección son la principal consideración, un desarrollo basado en transformaciones formales podría ser el más apropiado.

## Planificar

- Revisamos todo lo hecho, evaluándolo, y con ello decidimos si continuamos con las fases siguientes y planificamos la próxima actividad.

## Mecanismos de control

- La dimensión radial mide el coste.
- La dimensión angular mide el grado de avance del proyecto.

## Variaciones del Modelo En Espiral

- Modelo en Espiral Típico de seis regiones.
- Modelo en espiral WIN WIN.

## Ventajas

El análisis del riesgo se hace de forma explícita y clara. Une los mejores elementos de los restantes modelos.

- Reduce riesgos del proyecto
- Incorpora objetivos de calidad
- Integra el desarrollo con el mantenimiento, etc.

Además es posible tener en cuenta mejoras y nuevos requerimientos sin romper con la metodología, ya que este ciclo de vida no es rígido ni estático.

## Desventajas

- Genera mucho tiempo en el desarrollo del sistema
- Modelo costoso
- Requiere experiencia en la identificación de riesgos

## Inconvenientes

Planificar un proyecto con esta metodología es a menudo imposible, debido a la incertidumbre en el número de iteraciones que serán necesarias. En este contexto la evaluación de riesgos es de la mayor importancia y, para grandes proyectos, dicha evaluación requiere la intervención de profesionales de gran experiencia.

El IEEE clasifica al desarrollo en espiral como modelo no operativo en sus clasificaciones de MCV.<sup>[cita requerida]</sup>

## Véase también

- Ingeniería de Software
  - Desarrollo de Software
  - Modelo en Cascada o Secuencial
  - Modelo Iterativo Incremental
  - Modelo por Prototipos
  - Modelo de Desarrollo Rápido
-

## Enlaces externos

- *A Spiral Model of Software Development and Enhancement* <sup>[1]</sup>. artículo original de Barry Boehm en el que proponía el modelo de desarrollo en espiral (en inglés)

## Referencias

[1] <http://www.cs.umd.edu/class/spring2003/cmsc838p/Process/spiral.pdf>

---

# Fuentes y contribuyentes del artículo

**Desarrollo en espiral** *Fuente:* <http://es.wikipedia.org/w/index.php?oldid=44409196> *Contribuyentes:* Adelpine, Airunp, Balderai, Bucho, Caucas, Cinabrium, Cumanacr, Diegusjaimes, Dodo, EcLiB, Ereguero, Galle, GermanX, Gospelepszog, GuiXu, Ignacio Icke, Infovorov, Isha, Javierxar, Jorge c2010, Jose figueredo, Knilovc, Kordas, Kved, LarA, Martingala, Matdrones, Mxsgopr, Muro de Aguas, Murphy era un optimista, Netito777, Oscar ., PACO, Pezezin, PoLuX124, Rafa sanz, Retama, Rikaaii, RoyFocker, Sanbec, Savh, SergioN, Shooke, Siabef, Tamorlan, Taragui, Vanbasten 23, Wilfredor, Xsm34, Yamaneko, 143 ediciones anónimas

# Fuentes de imagen, Licencias y contribuyentes

**Archivo:ModeloEspiral.svg** *Fuente:* <http://es.wikipedia.org/w/index.php?title=Archivo:ModeloEspiral.svg> *Licencia:* Public Domain *Contribuyentes:* Ignacio Icke, Ma-Lik, Mdd

## Licencia

---

Creative Commons Attribution-Share Alike 3.0 Unported  
<http://creativecommons.org/licenses/by-sa/3.0/>