

# Metodologías Web



**ABRAHAM SÁNCHEZ LÓPEZ**  
**GRUPO MOVIS**  
**FCC-BUAP**

# Introducción

- Desde que se comenzó a plantear el desarrollo de sistemas software en la Web, la comunidad científica detectó la necesidad de proponer nuevos entornos metodológicos, técnicas y modelos que ofrecieran un marco de referencia adecuados a las nuevas características de este entorno.
- Con estas nuevas propuestas, nace una nueva línea de investigación dentro de la ingeniería de software que en los últimos años se ha denominado *Ingeniería Web*.
- Por ingeniería Web se entiende *la aplicación de sistemáticas, estructuradas y cuantificables propuestas de desarrollo, evaluación y mantenimiento de aplicaciones Web. El desarrollo de sistemas Web era al principio un proceso ad-hoc.*
- Las aplicaciones se desarrollaban sin ningún tipo de estructura y bajo ningún proceso que garantizara la calidad de los resultados.
- Cuando la ingeniería Web comenzó a nacer, surgieron muchas propuestas metodológicas y también muchos estudios comparativos y analíticos que pretendían analizar si realmente eran necesarios nuevos métodos para la Web.

# Más de la introducción

- Hoy en día, preguntarse si es necesario aplicar una metodología para desarrollar sistemas Web sería igual que preguntarse si es necesario aplicar una metodología para desarrollar un sistema de información en cualquier otro entorno.
- Desde hace años la ingeniería del software ha demostrado que la aplicación de correctos entornos metodológicos mejora la calidad de los resultados y abarata, no solo el proceso de desarrollo, sino que también mejora los costos de mantenimiento posterior del sistema.
- Algunas características de los sistemas Web como el carácter crítico del sistema de navegación o la importancia que adquiere la interfaz o la hipermedia y la personalización, requieren modelos que permitan desarrollar estos aspectos de manera adecuada.
- La necesidad del tratamiento adecuado de la navegación en el desarrollo de sistemas navegacionales ha sido una de las características que ha llevado a diversos grupos de investigación a proponer nuevos modelos y técnicas adecuadas para su tratamiento.

# Planteamiento I

- Durante este apartado se presenta una visión general de la situación actual del tratamiento de la navegación en la ingeniería Web ofreciendo una descripción breve de los procesos de desarrollo de las actuales propuestas.
- Esta visión general se centra principalmente en estudiar cómo se está tratando desde los marcos metodológicos el aspecto de la navegación en el ciclo de vida y de demostrar que este aspecto resulta crítico para todos los grupos de investigación que se mueven en el entorno de la ingeniería Web.
- Se trata así de ofrecer una visión de la inquietud general que existe dentro del mundo de la ingeniería Web por la navegación, mostrando como todas las propuestas incluyen la navegación en su ciclo de vida.
- Pero también muestra las lagunas y puntos abiertos que aún existen con respecto a la navegación.
- Por un lado, hay que comenzar diciendo que no existe una propuesta estándar aceptada por toda la comunidad investigadora. Aquí se presentan catorce metodologías, que integran las más referenciadas en la bibliografía.

# Planteamiento II

5

- Sin embargo, hay que decir que existen muchas otras que no han sido incluidas en este estudio bien porque no han sido tan difundidas, o bien porque actualmente están empezando a desarrollarse o bien porque se centran demasiado en las últimas etapas del ciclo de vida y no aportan ninguna información en el entorno de la navegación.
- También hay que resaltar que, al ser este un campo muy joven, la mayoría de las propuestas están siendo revisadas continuamente y cambian con mucha frecuencia. Se les añaden nuevos modelos, técnicas o fases en el ciclo de vida, se les modifican los ya existentes, etc.
- El estudio que se presenta aquí, se basa en la documentación encontrada, o en muchos casos cedida por los propios autores.
- Y por último, es necesario indicar que uno de los principales problemas que existen a la hora de estudiar metodologías para la Web es la multiplicidad de terminología y la falta de estándares.
- Sin embargo, puede observarse una evolución en los últimos años que están intentando unificar criterios para conseguir una propuesta unificada para la Web.

# Planteamiento III

6

- Para representar los procesos de cada propuesta o las relaciones entre los modelos que proponen, se han utilizado los diagramas de actividades y diagrama de clases de UML respectivamente:
  - HDM (Hypermedia Design Model)
  - RMM (Relationship Management Method)
  - EORM (Enhanced Object Relationship Methodology)
  - OOHDM (Object-Oriented Hypermedia Design Method)
  - WSDM (Web Site Design Method)
  - SOHDM (Scenario-based Object-oriented Hypermedia Design Methodology)
  - RNA (Relationship-Navigational Analysis)
  - HFPM (Hypermedia Flexible Process Modelling Strategy)
  - Building Web Applications with UML (Conallen)
  - WebML (Web Modelling Language)
  - UWE (UML-based Web Engineering)
  - W2000
  - Proyecto UWA (Ubiquitous Web Applications)
  - OOH (Object-Oriented Hypermedia Method)

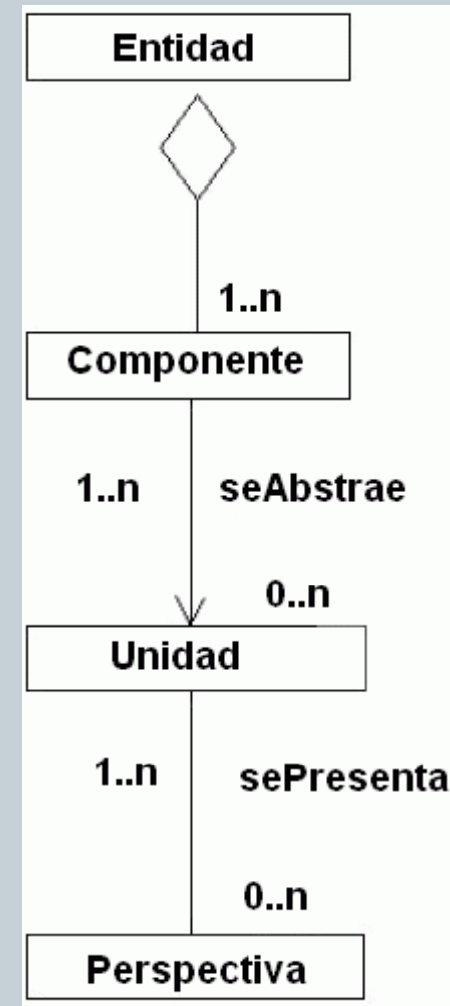
# HDM, I

- HDM [Garzotto et al., 1993] fue la primera aproximación que se hizo para dar soporte al desarrollo de sistemas de información hipermedia. No es realmente una metodología de desarrollo.
- En realidad, HDM es una extensión del modelo entidad-vínculo [Chen 1976, Codd 1992], en el que se incluyen nuevos aspectos para el modelado de sistemas hipermedia.
- HDM propone un conjunto de elementos que permiten al diseñador especificar una aplicación.
- Estos elementos son las *entidades*, los *componentes*, las *perspectivas*, las *unidades* y los *enlaces*.
- *Todos estos elementos pueden incorporarse en la semántica del* clásico modelo entidad-vínculo.
- Sin embargo, y a pesar de que los términos como las entidades hayan sido heredados de los ERD, han sido extendidos para poder representar una estructura compleja que contenga enlaces y una semántica de navegación interna.

# HDM, II

8

- En definitiva una aplicación especificada mediante un modelo HDM consiste en una estructura general compuesta por unas unidades básicas denominadas entidades. Una entidad denota un objeto físico o conceptual del universo del discurso de la aplicación.
- En HDM las entidades son agrupadas en *tipos de entidad*. Los tipos de entidad se caracterizan por un nombre, por un conjunto de perspectivas bajo las que se pueden presentar su contenido y un conjunto de *enlaces de aplicación* por los que se puede navegar.
- Una entidad es la unidad mínima autónoma de cualquier modelo HDM, pero existen otros conceptos añadidos que se verán a continuación.





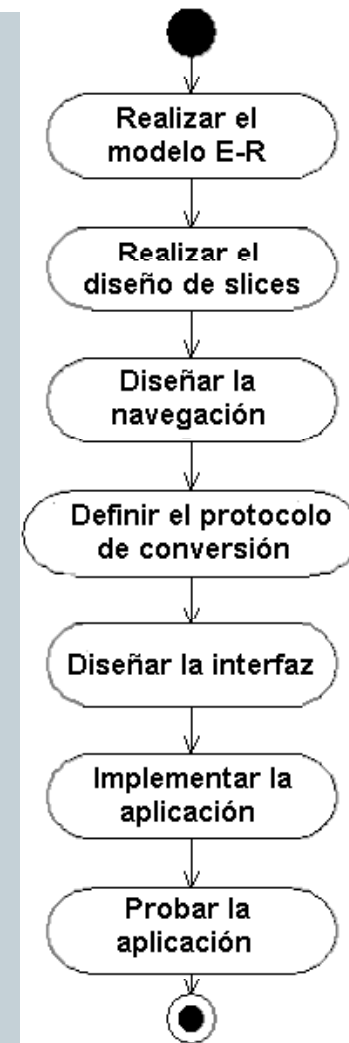
# HDM, III

- Cada entidad está compuesta por una jerarquía de *componentes que heredan las* propiedades de dicha entidad. Los componentes no tienen razón de ser sin que exista la entidad de la que dependen.
- Los componentes son, por su parte, abstracciones para diseñar un conjunto de *unidades o nodos que representan un mismo conjunto de* información de la entidad.
- Una unidad, es pues un depósito de la información contenida en una aplicación.
- Una unidad representa un fragmento del contenido de una entidad presentada bajo una *perspectiva particular. De esta forma, la perspectiva permite* representar la multiplicidad de presentaciones de un mismo contenido de información (por ejemplo, la presentación de un documento en múltiples idiomas).
- En la actualidad HDM no se usa, principalmente por dos razones. Por un lado, el paradigma estructurado ha dejado paso al paradigma de la orientación a objetos y por otro, aunque HDM propone ideas para el desarrollo de la hipermedia, no ofrece un proceso metodológico completo.
- Pero, a pesar de que haya caído en desuso, HDM ha sido el referente para otras muchas propuestas.

# RMM, I

10

- RMM [Izakowitz et al. 1995], a diferencia de su antecesora HDM, se puede considerar una metodología pues asume un ciclo de vida completo compuesto de siete fases o etapas en las que el diseñador va modelando la estructura de la aplicación y las posibilidades de navegación de la misma.
- La propuesta está basada en el modelo entidad-vínculo y en HDM. Partiendo de ellos define un nuevo modelo, el modelo RMDM, que propone un lenguaje que permite describir los objetos del dominio, sus interrelaciones y los mecanismos de navegación hipertexto de la aplicación.
- En la figura se muestra el ciclo de vida propuesto por RMM.



# RMM, II

- El proceso comienza con la *realización del modelo E-R sin entrar en detalles de navegación o presentación* y tratando el sistema hipermedia como a cualquier sistema clásico.
- Tras esto se hace el *diseño de slices*.
- *Un slice es un subconjunto de atributos de una entidad que van a ser presentados de forma agrupada al usuario, es lo que se podría llamar una vista del sistema.*
- El diseño de slices es enriquecido durante la siguiente fase de *diseño de la navegación, en la que se indica como se podrá pasar de una entidad a otra.*
- El diseño de slices y el enriquecimiento del mismo con los aspectos de la navegación generan el modelo RMDM.
- Basándose en este modelo, en la cuarta fase se *define el protocolo de conversión que va a describir el proceso a seguir para pasar del modelo RMDM a la plataforma de desarrollo concreta.*

# RMM, III

12

- Tras esto, se realiza el *diseño de la interfaz en el que se definen las pantallas tal y como se presentarán al usuario*.
- Conocida la interfaz, se pasa a la *implementación de la aplicación en el lenguaje concreto elegido*.
- Y por último, en la séptima fase se *realizan las pruebas de la aplicación*.

# EORM, I

13

- Junto con OOHDM, EORM [Lange 1995] es una de las primeras propuestas para la Web centrada en el paradigma de la orientación a objetos.
- Se basa en muchas de las ideas que se definen en HDM, pero las traslada a la orientación a objetos.
- El ciclo de vida que propone EORM es bastante clásico y se muestra en la figura.
- EORM estructura el proceso de desarrollo en tres fases: *análisis*, *diseño* y *construcción*.
- La fase de análisis realmente no se corresponde con lo que otras propuestas denominan análisis.
- Se correspondería más a un diseño de objetos, hablando en términos de OMT.



# EORM, II

14

- Consiste en hacer un modelo orientado a objetos, según las pautas y nomenclatura de OMT para representar la aplicación.
- En esta primera fase, aspectos como la navegación o la interfaz no se toman en cuenta [Lange 1994].
- Tras esto, en la fase de diseño se procede a modificar el modelo de objetos obtenido en la fase anterior añadiendo semántica suficiente a las relaciones para representar los enlaces.
- Este modelo de objetos enriquecido se denomina *EORM* y en él se van a reflejar tanto la estructura de la información (modelo abstracto hipermedial compuesto por nodos y enlaces) como las posibilidades de navegación ofrecidas por el sistema.
- Finalmente, en la fase de construcción todos estos aspectos son traducidos al lenguaje de programación concreto que se esté definiendo.

# OOHDM, I

15

- De todas las propuestas de desarrollo Web que han surgido en los últimos años, es quizás OOHDM [Rossi, 1996] la que más ha influido y más ha sido estudiada.
- Esta propuesta nace basada en HDM pero dentro del paradigma de la orientación a objetos.
- Lo más interesante de OOHDM es la gran aceptación que ha tenido el proceso de desarrollo que propone y que se muestra en la siguiente figura.
- Este proceso comienza con la *realización del modelo de clases conceptuales*. En él, mediante un *diagrama de clases*, se representa la estructura estática del sistema.
- Tras esto, se realiza un *modelo de navegación del sistema*. Este modelo ofrece una *vista del modelo conceptual* y expresa cómo se podrá navegar a través de la información representada en el modelo conceptual.
- En la tercera fase del proceso se realiza el *modelo de interfaz abstracta*.
- Este modelo ofrece una vista de cómo se va a presentar la información al usuario.
- De esta forma, el modelo de navegación es una vista del conceptual y el de interfaz abstracta es una vista del modelo navegacional.

# OOHDM, II

16

- En la última fase, se realiza la *implementación* de los tres aspectos.
- En todas estas fases, OOHDM ofrece lenguajes de modelado específicos para representar estos modelos. Algunos de ellos, como los contextos de navegación o las vistas abstractas de datos han caído en desuso en la actualidad.
- Sin embargo, otros como el hecho de representar la navegación mediante un diagrama de clases especiales denominadas *clases de navegación*, ha sido ampliamente aceptada por la comunidad investigadora.





# OOHDM, III

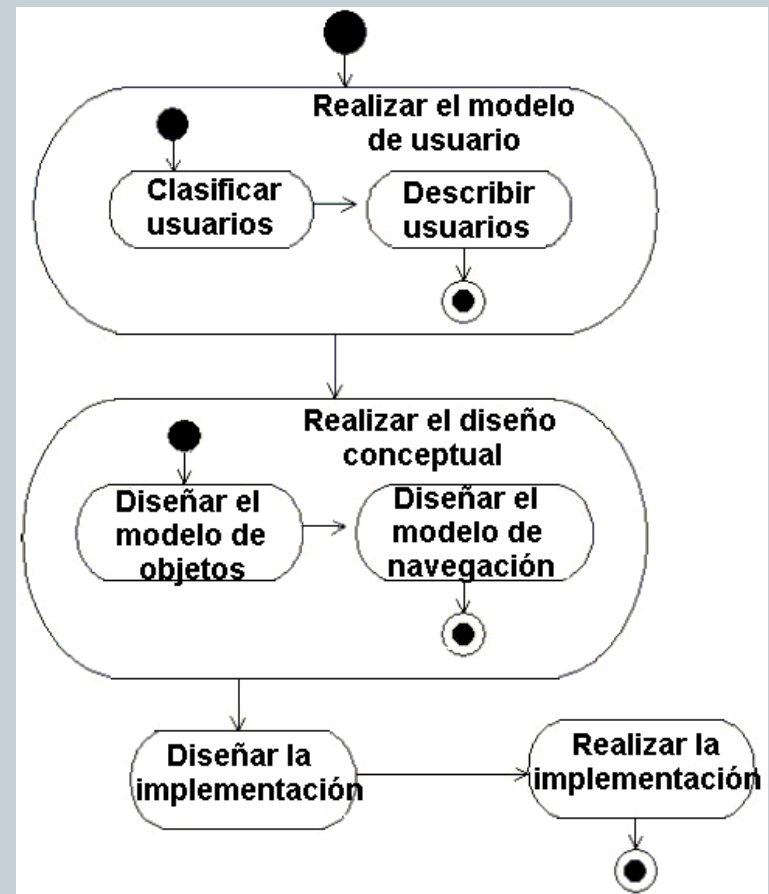
17

- Otra de las ideas que OOHDM propuso y que ha tenido mucha aceptación ha sido el hecho de separar el modelado de los aspectos de los sistemas hipermedia.
- El modelar lo conceptual, lo navegacional y la interfaz abstracta de manera independiente ha dado muy buenos resultados y ha sido asumido en muchas propuestas posteriores como se analizará más adelante.
- Por último, debemos decir que OOHDM no es una propuesta estática [Schwabe & de Almenia 1998].
- En la actualidad está siendo mejorada y enriquecida.
- Así, por ejemplo, se ha completado añadiendo una fase previa de tratamiento de requisitos, basado en una técnica denominada de UIDs [Schwabe & Rossi 1998a] [Vilain et al. 2000a][Lima & Schwabe 2003].

# WSDM, I

18

- El *Método para diseño de sitios Web (WSDM)* [De Troyer & Leune 1998] es una propuesta en la que el sistema se define en base a los grupos de usuarios.
- Su ciclo de vida se divide en cuatro fases: *modelo de usuario, diseño conceptual, diseño de la implementación e implementación.*
- A su vez, el *modelo de usuario* se divide en dos subfases: *clasificación y descripción.*
- Por su parte el *diseño conceptual* se divide en otras dos subfases: *modelado de objetos y diseño navegacional.*
- En la figura se representa su ciclo de vida.



# WSDM, II

19

- Durante la fase de realización del modelo de usuario, se estudian los diferentes roles del usuario que van a interactuar con el sistema y se clasifican según las relaciones que existan entre ellos.
- En la siguiente fase, y basados en la clasificación de usuarios realizada, se realiza el modelado conceptual del sistema.
- El modelado conceptual no tiene el mismo significado que en OOHDM.
- Durante el modelado conceptual se realizan dos tareas a la vez:
  - el modelado de objetos, que es lo que en OOHDM se llama modelo conceptual, y
  - el diseño de la navegación, que coincide con la idea del diseño navegacional de OOHDM.
- En WSDM puede existir más de un modelo de navegación, dependiendo de los roles de usuario detectados durante la primera fase.

# WSDM, III

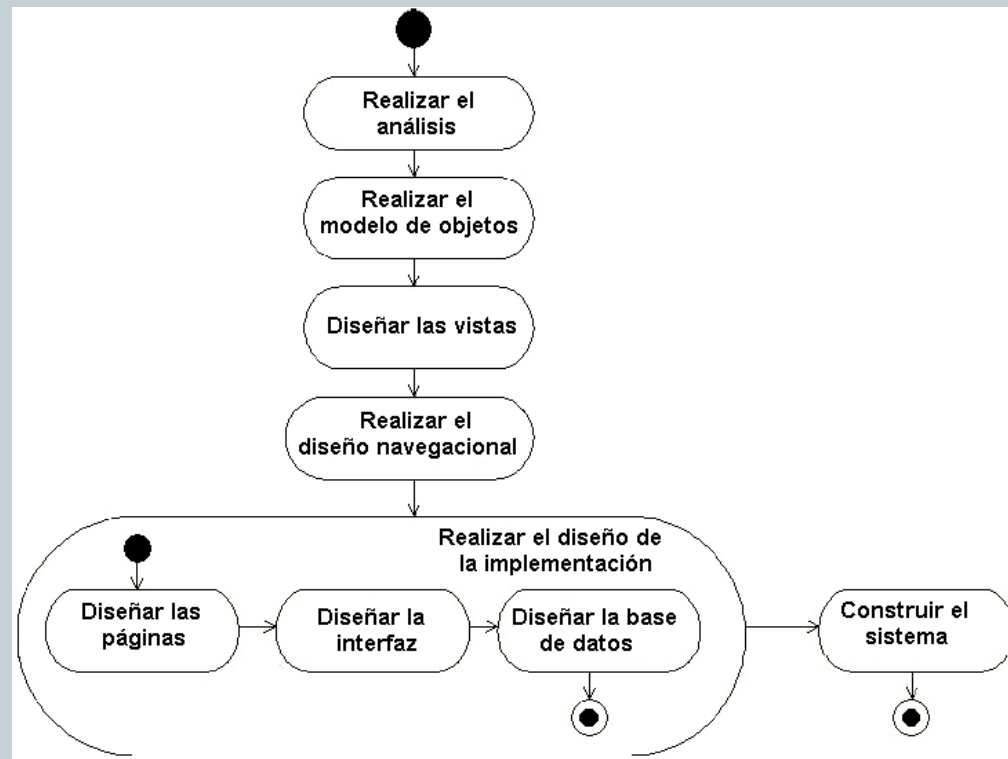
20

- Tras esto, y ya en la fase de diseño de la implementación, se modela la interfaz para cada rol de usuario.
- Y por último, en la fase de implementación se codifican todos estos aspectos en el lenguaje concreto que se haya seleccionado.
- WSDM es también una propuesta viva que está cambiando y adaptándose a nuevos requisitos.
- En la actualidad, uno de los trabajos más interesantes y novedosos que se le está aplicando es el desarrollo de una herramienta CASE que permita aplicar el ciclo de vida de desarrollo de WSDM [De Troyer et al. 2003a][De Troyer et al. 2003b].

# SOHDM, I

21

- SOHDM [Lee et al.1998][Suh & Lee 2001] es una de las primeras propuestas para la Web que más importancia da a la tarea de tratamiento de requisitos.
- Se caracteriza principalmente porque su ciclo de vida comienza con la aplicación de los escenarios como técnica de elicitación y definición de requisitos [Weidenhaupt et al. 1998].
- Su proceso de desarrollo se divide en seis fases tal y como se describe en la figura.



# SOHDM, II

- Dicho proceso comienza por la fase de *análisis donde se debe realizar un estudio de las necesidades de la aplicación, del entorno de trabajo y de los actores.*
- La finalidad principal de esta fase es conseguir los escenarios que representen las actividades que se pueden llevar a cabo en el sistema.
- Tras esto, se realiza un *modelado de objetos* en el que se desarrolla un diagrama de clases que representa la estructura conceptual del sistema.
- En la siguiente fase, la de *diseño de vistas*, los objetos son reorganizados en *unidades navegacionales que representan una vista de los objetos del sistema.*
- En la fase de *diseño navegacional* se enriquecen dichas vistas definiendo los *enlaces e hiperenlaces* que existen en el sistema.
- Tras esto, se pasa a la fase de *diseño de la implementación*, en la que se diseñan las *páginas, la interfaz y la base de datos del sistema.*
- Y por último, se afronta la *fase de construcción en la que se implementa la aplicación.*

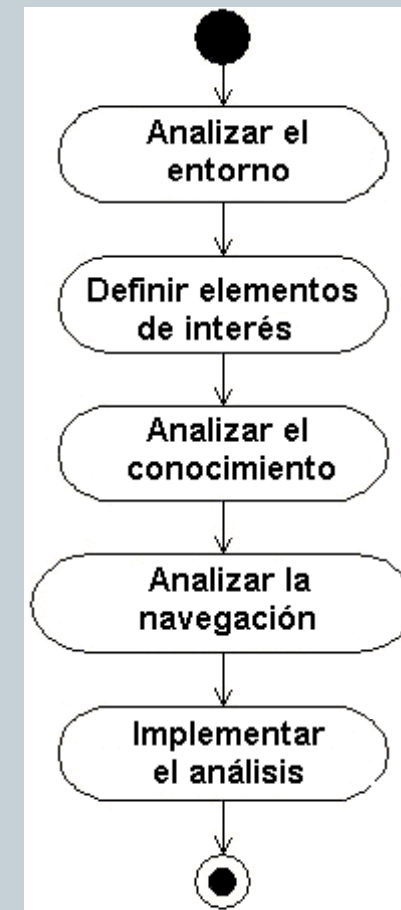
# RNA, I

- A diferencia de otras propuestas, RNA [Bieber et al. 1998] no cubre varias fases del ciclo de vida y centra todo su proceso de desarrollo en la fase de análisis del sistema.
- Sin embargo, como se puede concluir tras la presentación de su proceso de desarrollo, el concepto de análisis no coincide exactamente con el de otras propuestas y casi se podría decir que cubre tareas de la especificación de requisitos, el análisis, el diseño o incluso la implementación.
- En la figura siguiente se presenta un diagrama de su ciclo de vida.
- Éste comienza con un *análisis del entorno, en el que se estudia las características de la audiencia, clasificando y estudiando los diferentes roles del sistema.*
- Una vez estudiados el entorno, se realiza la *definición de los elementos de interés como pantallas, documentos, etc. que se van a requerir durante el proceso de desarrollo.*
- El proceso continúa con el *análisis del conocimiento, que se basa en desarrollar y plantear un esquema que represente el sistema, identificando los objetos, los procesos y las operaciones que debe ofrecer.*

# RNA, II

24

- Tras esto, se realiza un *análisis de la navegación*, durante el que se estudia cómo se podrá navegar en el sistema.
- Y por último, se realiza una *implementación del análisis* en el que se codifica el sistema definido en el lenguaje de programación seleccionado.





# HFPM, I

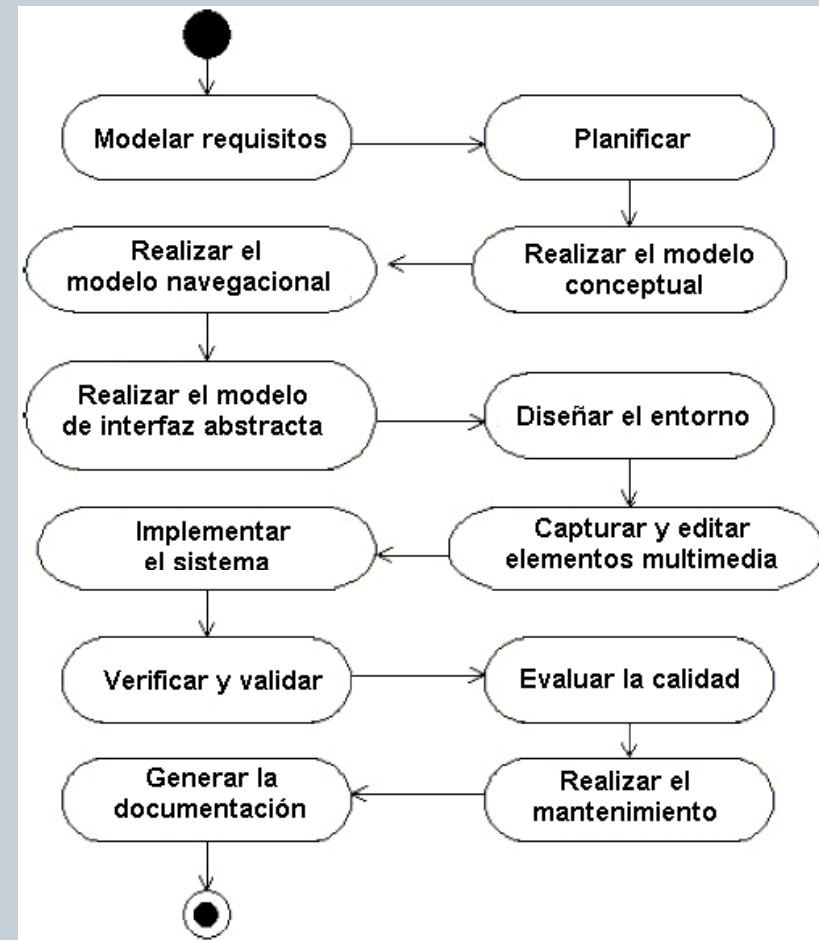
25

- HFPM [Olsina 1998] es quizás la propuesta más completa que se ha realizado en cuanto a las fases del ciclo de vida que cubre.
- No se caracteriza por proponer nuevos modelos o técnicas y asume muchas de las ideas de OOHDM.
- Sin embargo, sí cubre todo el proceso de desarrollo completo, incluso hasta la generación de los documentos a realizar.
- En la figura siguiente se presenta un esquema del mismo.
- Cada una de las fases de este proceso, se concreta a su vez en tareas y subtareas, ofreciendo así una detallada guía de desarrollo.
- El proceso comienza con un *modelado de los requisitos que se deben cubrir y con la realización de la planificación*.
- *Tras esto, se realizan las fases de modelado conceptual, modelado navegacional y modelado de interfaz abstracta, que son similares a los de OOHDM.*
- Tras ellos, se realiza el *diseño del entorno, en el que se enriquecen los modelos anteriores mediante patrones de diseño*.

# HFPM, II

26

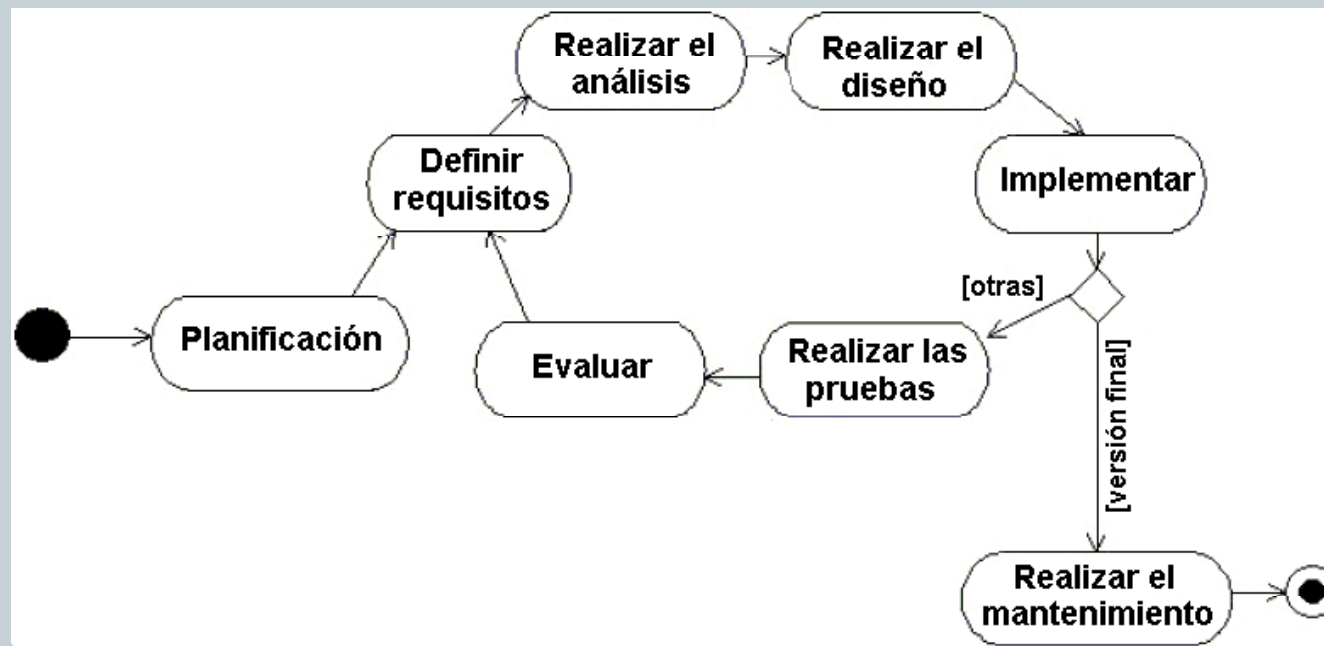
- En la siguiente fase, de *captura y edición de elementos multimedia*, se deben plantear los *múltiples medios con los que se va a trabajar*, así como los sistemas de almacenamiento que se usarán en los mismos.
- Con todo este conocimiento, se pasa a la *implementación del sistema*. Y tras ello, a una fase de *validación y verificación*.
- En este punto, sobre el sistema se aplican una serie de métricas que permiten evaluar la calidad del mismo.
- Y para terminar, se contemplan las fases de *mantenimiento y de generación de documentación como el manual de usuario o los documentos resultantes de cada fase*.



# Conallen, I

27

- La propuesta de Conallen [Conallen 1999] es una propuesta que también ha adquirido gran importancia en los últimos años.
- Se caracteriza por basarse en UML y en el proceso unificado en todo su ciclo de vida y realiza muchas ampliaciones y definiciones de estereotipos propios para la Web, usando las posibilidades de extensión de UML.



# Conallen, II

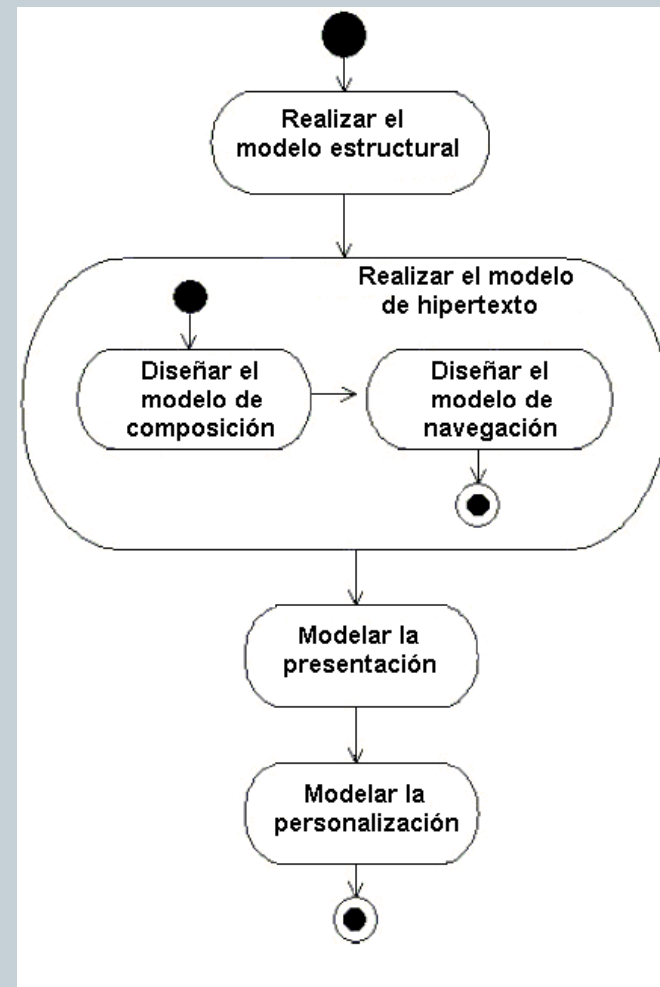
28

- Su ciclo de vida comienza con una fase previa de *planificación*.
- *Tras ella, se comienza un proceso cíclico que cubre las fases de tratamiento de requisitos, análisis, diseño, implementación, pruebas y evaluación. Una vez aceptado el sistema, se pasa a la explotación del mismo.*
- Básicamente, los modelos y técnicas que ofrece están heredados de UML, pero los enriquece y define muchos nuevos elementos de modelado específicos para la Web como la posibilidad de representar en los modelos de clase unidades java beans, frames, botones, etc.
- Estos nuevos estereotipos pueden incorporarse en la herramienta Rational Rose y así utilizarlos fácilmente durante la realización de los modelos.

# WebML, I

29

- Tal y como sus propios autores lo definen, WebML [Ceri et al. 2000][Ceri et al. 2003] es una notación para especificar complejos sitios Web en el ámbito conceptual [Ceri et al. 2002].
- Permite una descripción de los sitios Web desde distintos puntos de vista: el conceptual, el navegacional, el de presentación, etc.
- Su ciclo de vida es mostrado en la figura siguiente.
- El proceso de desarrollo comienza con el *modelado conceptual del sistema*, en el que mediante algún lenguaje de modelado como UML, WebML no exige ninguno en concreto, se representa la estructura estática del sistema.



# WebML, II

30

- Tras esto, se realiza el *modelo de hipertexto en el que se describen* uno o más hipertextos que pueden ser publicados en el sitio Web.
- Cada uno de estos hipertextos define una vista del sitio.
- La descripción de los hipertextos se realiza mediante dos modelos: el *modelo de composición, que define las páginas que componen* el sistema, y el *modelo de navegación, que describe cómo se podrá navegar a través de ellas*.
- En el siguiente paso del proceso se describe el *modelo de presentación, que define la apariencia física de las páginas*.
- Y por último, el *modelo de personalización* define como debe adaptarse el sistema a los diferentes roles de usuario.
- Uno de los aportes más interesantes de WebML es que ofrece una herramienta CASE que permite aplicar las técnicas propuestas y conseguir los resultados sistemáticamente [WebRatio 2004].

# UWE, I

31

- UWE [Koch 2001] es una propuesta metodológica basada en el Proceso Unificado [Jacobson et al. 1999] [Kruchten 1998] y UML para el desarrollo de aplicaciones Web.
- UWE cubre todo el ciclo de vida de este tipo de aplicaciones centrando además su atención en aplicaciones personalizadas o adaptativas.
- Su proceso de desarrollo se basa en tres fases principales: la fase de captura de requisitos, la fase de análisis y diseño y la fase de implementación.
- El proceso de desarrollo de UWE se caracteriza por la importancia que da a la segunda fase, la de análisis y diseño.



# UWE, II

- Todo el proceso de desarrollo de UWE se encuentra detallado y definido, así como la estructura de los modelos que se van generando.
- Sin embargo es en el análisis y diseño donde se enfoca más la propuesta.
- UWE es una propuesta muy completa que concreta mucho las tareas a realizar.
- De esta forma, y aunque no se ha concretado en la figura para no complicarla demasiado UWE propone las siguientes tareas:
  1. En la fase de requisitos: propone comenzar con la identificación de los usuarios y la elicitación de los requisitos. Trata de diferente forma las necesidades de información, las necesidades de navegación, las necesidades de adaptación y las de interfaz de usuario, así como algunos requisitos adicionales relacionados, por ejemplo, con las restricciones hardware o la seguridad. Tras esto, centra el trabajo en el estudio de los casos de uso, la generación de los glosarios y el prototipado de la interfaz de usuario.
  2. La fase de análisis y diseño es similar a la de OOHDM. Sin embargo, UWE engloba más aspectos que OOHDM. De hecho, UWE distingue entre diseño conceptual, de modelo de usuario, de navegación, de presentación, de adaptación, de la arquitectura, en el diseño detallado de las clases y en la definición de los subsistemas e interfaces.



# UWE, III

3. Por último, en la fase de implementación, UWE incluye todas las tareas que llevan a la implementación de los modelos aceptados: implementación de la arquitectura, implementación de la estructura del hiperespacio, implementación del modelo de usuario, implementación de la interfaz de usuario, implementación de los mecanismos adaptativos y las tareas referentes a la integración de todas estas implementaciones.
- Centrados en el estudio de UWE en el tratamiento de la navegación, podemos comentar que UWE propone ya en su fase de requisitos una elicitación de los requisitos de navegación.
  - UWE considera los requisitos de navegación como un tipo de requisito funcional y, aunque realmente no propone técnicas específicas para este tratamiento, principalmente se basa en los casos de uso [Jacobson 1995][Booch et al. 1999].
  - Los separa con idea de identificar mejor los aspectos que influirán en el modelo navegacional que se realiza en la fase de análisis y diseño.
  - Este modelo de navegación se construye en dos fases. En una primera etapa se desarrolla un modelo de espacio de la navegación, construido como vista del modelo conceptual y que indica cuáles son las clases y modelos visitables.

# UWE, IV

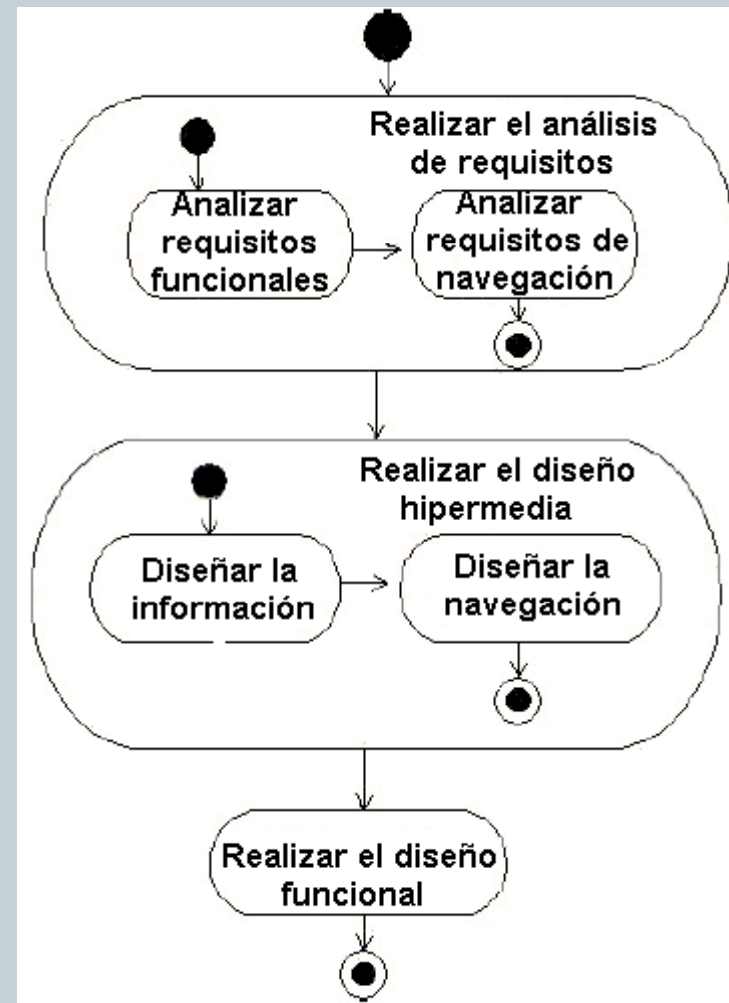
34

- Se representa mediante un modelo de clases especiales denominadas clases navegacionales, que no son más que clases de UML estereotipadas para indicar su semántica.
- Este modelo se enriquece en una segunda etapa con el modelo de la estructura de la navegación. En él ya no sólo se indica qué es visitable, sino también cómo estos objetos son visitados.
- UWE es una propuesta que en los últimos años ha conseguido gran aceptación en los foros de investigación.
- Sus modelos basados totalmente en UML están siendo muy bien valorados.
- Además, es una propuesta viva.
- Actualmente también se está trabajando en una herramienta que sea capaz de soportar su ciclo de vida denominada ArgoUWE [ArgoUWE 2004].

# W2000, I

35

- La propuesta de W2000 [Baresi et al. 2001] ha sido la evolución a la orientación a objetos de su antecesora HDM.
- Con respecto a HDM tiene dos diferencias básicas. La primera de ellas es que HDM no era realmente una propuesta metodológica sino un modelo enriquecido del diagrama Entidad-Vínculo, W2000 sí propone un ciclo de vida para el desarrollo de sistemas Web.
- La otra gran diferencia es que W2000 se centra en el paradigma de la orientación a objetos.
- Pero a pesar de estas diferencias, todas las aserciones que propuso HDM han sido aceptadas como válidas en W2000 y han sido adecuadas a la orientación a objetos.



# W2000, II

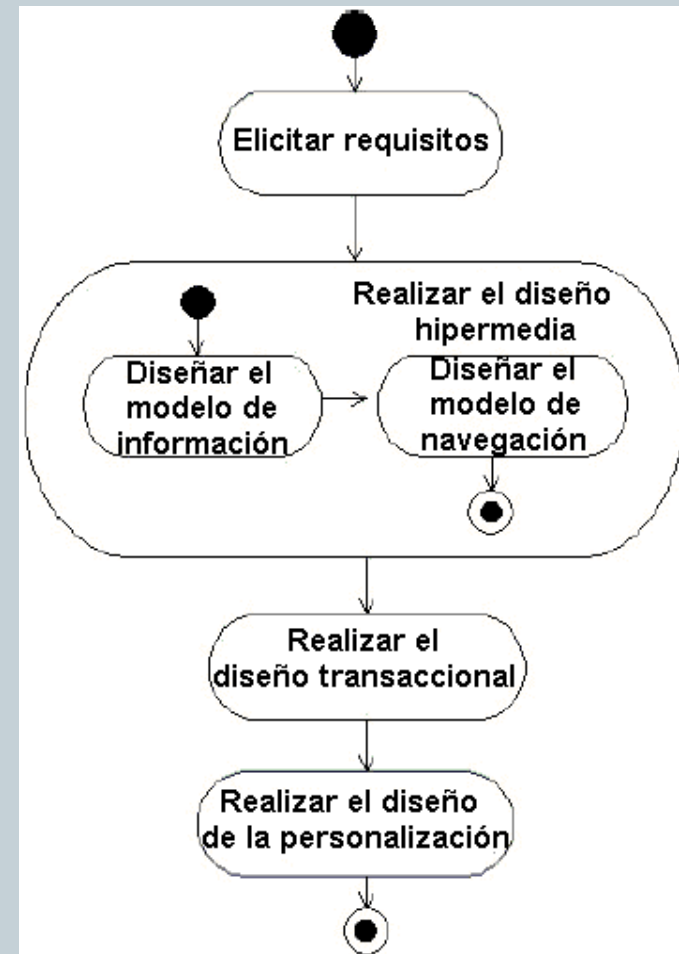
36

- El proceso comienza con una fase de *análisis de requisitos basado principalmente en los casos de uso*.
- Con el conocimiento adquirido durante la fase de requisitos, se pasa a la fase de *diseño de hipermedia*. En ella, se realizan dos modelos: el *modelo conceptual* y el *modelo navegacional*.
- Para ello, los autores han modificado y extendido algunos modelos de UML como el diagrama de clases o el diagrama de estados.
- Por último, se pasa a una fase de *diseño funcional*, en el que se ha adaptado el *diagrama de secuencias* para expresar la funcionalidad del sistema.
- Una característica importante de W2000, es que separa el aspecto de la navegación y la estructura de la información desde las primeras fases del ciclo de vida.
- Existe un análisis de requisitos funcionales y un análisis de requisitos navegacionales.
- Sin embargo, la técnica usada para ambos tipos de requisitos son los casos de uso, sin especificar realmente cómo se pueden llegar a separar, tratar, identificar y elicitar ambos tipos de requisitos de manera específica.

# Proyecto UWA, I

37

- El proyecto UWA [UWA 2001] es un proyecto en el que participan diversos grupos de investigación que desde hace años han trabajado en el mundo de la ingeniería Web.
- De hecho, muchas de las propuestas que se hacen en su ciclo de vida y de las técnicas y modelos que utilizan han sido heredadas de otras propuestas como W2000.
- El proceso de desarrollo de UWA se divide en cuatro grandes fases tal y como se muestra en la figura:
  1. Elicitación de requisitos,
  2. El diseño hipermedia,
  3. El diseño de transacciones, y
  4. El diseño de la personalización.



# Proyecto UWA, II

38

1. *La elicitación de requisitos, en la que se decide qué debe hacer la aplicación. Es una elicitación de requisitos basada en objetivos.*

El proceso comienza marcando los objetivos a alto nivel y a partir de ellos, se van concretando las necesidades que acaban definiendo los requisitos del sistema.

1. *El diseño hipermedia, que modela los datos, la navegación y presentación de los mismos y los servicios que se ofrecen al usuario. Esta fase viene heredada de W2000.*

De hecho propone tres modelos para modelar la hipermedia: el modelo de información, el modelo de navegación y el modelo de presentación.

1. *El diseño de transacciones, que modela el comportamiento de las transacciones que se producen en la interacción y cómo afectan al sistema.*

En esta fase se modelan las transacciones desde dos puntos de vista: el punto de vista estático, que estudia qué transacciones se pueden producir, y el punto de vista dinámico, que estudia las consecuencias que cada transacción puede producir en el sistema.

# Proyecto UWA, III

39

4. El *diseño de la personalización*, que analiza cómo se debe adaptar la aplicación al contexto en el que se ejecuta. Esta fase es heredada de WUML[Kappel et al. 2001a].

Esta es una propuesta de metodología para la Web que no ha sido incluida en este estudio pues está centrada exclusivamente en las últimas fases del ciclo de vida. Su propuesta se basa en el uso de los componentes como técnica de implementación, por lo que el diseño de la personalización en el proyecto UWA se hace mediante componentes.

# OOH, I

40

- OOH [Cachero 2003] nace con la orientación orientada a objetos, con la idea de capturar la semántica necesaria para el modelado eficiente de interfaces de usuario y su implantación en Web.
- La idea de OOH es ofrecer un marco metodológico que permita modelar de manera sistemática interfaces adaptativas y adecuadas al conjunto de usuarios del sistema.
- Su ciclo de vida, como se muestra en la figura, comienza con un análisis de requisitos que da paso a la fase de ingeniería.
- En esta fase de ingeniería se realiza el análisis y el diseño del dominio y de la navegación, así como el diseño de la presentación.





# OOH, II

41

- Tras esto se pasa a una fase de construcción y adaptación en la que se obtiene, en base a un conjunto de plantillas, el sistema final.
- La última fase incluye la evaluación de la interfaz por parte del cliente.
- Todas estas fases se centran mucho en los roles de usuario.
- OOH se caracteriza por diseñar interfaces en las que se implementa la navegación del sistema adaptado a cada rol de usuario a las necesidades que cada uno de ellos plantea.
- Otra de las características más importantes de OOH es que lleva asociada una herramienta que soporta su ciclo de vida.
- Toda la información de VisualWADE está disponible en [VisualWADE 2004].

# Otras metodologías I

- Obviamente existen otras propuestas, que no se han incluido tan detalladamente, porque o bien son muy recientes, o porque resulta difícil encontrar bibliografía reciente de alguna de ellas.
- **OO-Method y OOWS:** la propuesta OO-Method [Pastor et al. 1997], desarrollada inicialmente para proyectos de ingeniería software en general, ha evolucionado en los últimos años dando origen a nuevas versiones en las que se han incluido aspectos más concretos para la Web. De todas ellas, una de las más interesantes, es OOWS [Fons et al. 2003]. OOWS es una propuesta para el tratamiento de requisitos y análisis de la navegación en la Web. Sin embargo es aún una propuesta bastante joven.
- **Proyecto Araneus:** el trabajo realizado Paolo Atzeni y Paolo Merialdo de la Universidad de Roma y por Giasalvatore Mecca de la universidad de la Basilicata [Mecca et al. 1999]. denominado proyecto Araneus se basa en un modelo denominado Araneus. Éste es un modelo de datos específico para describir esquemas de aplicaciones con hipertextos. Está orientado al mundo de las bases de datos y al modelo E/R.

# Otras metodologías II

- **The MacWeb Hypermedia Design Method:** el entorno de desarrollo hipermedia MacWeb, propuesto en 1995 [Nanard & Nanard 1995] no se puede considerar una propuesta metodológica propiamente dicha. En el entorno MacWeb, una aplicación hipermedia se desarrolla en base a la interfaz. Los autores de este entorno puntualizan en el hecho de que lo más importante de una aplicación Web es la comunicación con el usuario. Por ello, centran todo el proceso de desarrollo en la interfaz.
- **OO/Pattern Approach:** esta aproximación [Thomson et al. 1998] es bastante similar a HFPM pues ambas proponen el uso de patrones y de la orientación a objetos para el diseño navegacional y la interfaz. Sin embargo, esta propuesta, a diferencia de HFPM, no cubre el ciclo completo de desarrollo. Resulta, por el contrario, interesante pues es la primera propuesta que asume la técnica de los conocidos casos de uso para realizar la fase de análisis de la aplicación.
- **OSM:** no es en sí una propuesta metodológica, es un modelo orientado a objetos que pretende ser lo suficientemente sólido como para dar soporte a todas las fases del ciclo de vida de un proyecto de desarrollo de software en la Web (especificación, análisis, diseño, implementación y evolución) [Liddle et al.2001a].

# Otras metodologías III

44

- Propone representar los sistemas Web mediante tres modelos: el modelo de objetos y relaciones entre ellos, el modelo de comportamiento, el modelo de presentación [Liddle et al. 2001b]. Es una de las propuestas más originales pues no está cercana a ninguna de las anteriores presentadas.
- **Design-driven Requirements Elicitation:** esta propuesta centrada en requisitos es una parte del proceso propuesto por Lowe y Eklund [Lowe & Eklund 2002] [Eklund & Lowe 2001] para desarrollar aplicaciones Web. Consiste en un proceso de capturar, definir y validar requisitos durante el proceso de diseño. El proceso se basa en el prototipado para evaluar con el cliente las posibilidades. Uno de los aspectos más interesantes de esta propuesta es que ha sido realizada en base a resultados empíricos en proyectos reales.
- **WUML:** es también una propuesta muy cercana a las últimas fases del ciclo de vida [Kappel et al. 2001b]. De hecho parte del mismo ha sido aceptada en metodologías más generales como la propuesta del proyecto UWA. Se basa también en el uso de UML como lenguaje gráfico de modelado.

# Otras metodologías IV

45

- **WebComposition:** es una propuesta metodológica bastante peculiar pues se basa en el uso de componentes en su ciclo de vida [Gellersen et al. 1997]. Su ciclo de vida, totalmente centrado en una fase de diseño muy cercano a la implementación comienza con el desarrollo de los componentes que luego se dividen y prototipan hasta llegar a implementarlos.