

Jornadas Chilenas de Computación 2001



Universidad de Magallanes

Nov. 5-9, 2001, Punta Arenas, CHILE



Sociedad Chilena
de Ciencia de la
Computación



Instituto de Ingenieros
Electricistas y
Electrónicos



Centro LatinoAmericano
de Estudios en
Informatica



Sociedad de Técnicos
en Informática



Comisión Nacional de
Investigación Científica
y Tecnológica, CHILE



Facultad de Ingeniería
Universidad de
Magallanes



[Portada](#)

[Descripción](#)

[IX Encuentro Chileno de Computación](#)

[V Workshop de Sistemas Distribuidos y Paralelismo](#)

[III Congreso de Educación Superior en Computación](#)

[II Workshop de Inteligencia Artificial](#)

[I Workshop de Ingeniería de Software](#)

[Organizadores](#)

[Galería de Fotos](#)

Jornadas Chilenas de Computación 2001

Nov. 5-9, 2001, Punta Arenas, CHILE

IX Encuentro Chileno de Computación 2001

Bases de Datos y Sistemas de Información

1. "Algoritmos de inserción segura para esquemas conceptuales de Bases de Datos"
Gonzalo Rojas D. (grojas@inf.udec.cl), Chile.
2. "Representación de información imprecisa en modelo conceptual ERR difuso"
Angélica Urrutia (aurrutia@spock.ucm.cl), José Galindo (ppgg@lcc.uma.es), Chile.
3. "Restricciones dinámicas en bases de datos"
Mauricio Zambrano (mauricio@inf.udec.cl), Chile.
4. "Comparación del álgebra relacional en bases de datos relacionales clásicas y difusas"
Angélica Urrutia Sepúlveda (aurrutia@spock.ucm.cl), Carolina Rodríguez Deik (carodeik@yahoo.com), Chile.
5. "Construcción de aplicaciones de bases de datos utilizando al arquitectura MVC y Java",
Luis A. Lobos F. (llobos@ucn.cl), Paulina A. Gaete G., Chile.
6. "Clasificación y taxonomía de minería de datos"
Ricardo Baeza-Yates (rbaeza@dcc.uchile.cl), Luis F. Bastias (lbastias@dcc.uchile.cl), Josep L. Larriba-Pey (larri@ac.upc.es), Chile.
7. "Sistema de informacoes geograficas hipermedia aberto"
Celso Roberto Perez (crp@cin.ufpe.br), Ana Carolina Salgado (acs@cin.ufpe.br), Brasil.
8. "El darse-cuenta cooperativo necesita una ontología"
Edmundo P. Leiva-Lobos (epleiva@diinf.usach.cl), Bruno Jerardino Wiesenborn (bjerardi@usach.cl), Chile.
9. "Componentes de percepcao para o ambiente PROSOFT cooperativo"
Isabel Dillmann Lima Reis (isabeldn@terra.com.br), Carla Alessandri Lima Reis (clima@inf.ufrgs.br), Rodrigo Quites Reis (quites@inf.ufrgs.br), Daltro José Nunes (daltro@inf.ufrgs.br), Brasil.

Ingeniería de software y orientación al objeto

1. "Estimación del esfuerzo a partir del modelo de datos"
Gastón Jiménez (gaston.gimenez@hospitalitaliano.com.ar), Germán Gioia (g_gioia@yahoo.com), Jorge Perez Herrera (turcoph@yahoo.com), Rodolfo Bertone (pbertone@lidi.info.unlp.edu.ar), Hugo D. Ramón (hramon@lidi.info.unlp.edu.ar), Argentina.

2. "Feature analysis for quality-based architectural design methods"
Losavio, F. (flosavio@isys.ciens.ucv.ve), Chirinos, L. (lchirino@isys.ciens.ucv.ve),
Venezuela.
3. "Propuesta de método de transformación de prototipo a bases de diagramas UML"
Silvana Roncagliolo (silvana@ucv.cl), Jorge Bozo (jbozo@ucv.cl), Beatriz Marin
(beatriz.marin.c@mail.ucv.cl), Chile.
4. "Herramienta para metamodelización de procesos software"
Francisco Ruiz (fruiz@inf-cr.uclm.es), Luis Márquez (lmarquez@proyectos.inf-
cr.uclm.es), Félix Garcia (fgarcia@proyectos.inf-cr.uclm.es), España.
5. "Análisis formal del proceso organización de un modelo de proceso multinivel."
Silvia Acuña (silvac@unse.edu.ar), Mabel del Valle Sosa (litasosa@unse.edu.ar),
Argentina.

"Pequeños lenguajes: Una técnica de desarrollo de software"
Horst H. von Brand (vonbrand@inf.utfsm.cl), Chile
6. "Specification of attribute-based architectural styles"
F. Losavio (flosavio@isys.ciens.ucv.ve), N. Levy (Nicole.Levy@prism.uvsq.fr),
Venezuela.
7. "O RPG como tecnica na construcao de software educacional"
Cassiano Dal Pizzol (11030@lci.upf.tche.br), Alexandre Lazaretti Zanatta
(zanatta@upf.tche.br), Brasil.
8. "Implementación de un modelo de ejecución en entornos orientados a objetos para los
disparos de OO-Method"
Eva Campos (ecmonge@dsic.upv.es), Manoli Albert (malbert@dsic.upv.es),
Vicente Pelechano (pele@dsic.upv.es), Oscar Pastor (opastor@dsic.upv.es), España.

Sistemas Distribuidos

1. "Integracao de bancos de dados utilizando agentes moveis"
Daniela Barreiro Claro (danclaro@inf.ufsc.br), Ana Ruth V. da Silva
(anaruth@inf.ufsc.br), Fabio Nascimento (binho@inf.ufsc.br), Joao Bosco M. Sobral
(bosco@inf.ufsc.br), Brasil.
2. "Un estudio actualizado del tamaño, distribución y uso de archivos en un ambiente
académico"
Claudio Inglesi (cinglesi@ing.puc.cl), Alvaro E. Campos (acampos@ing.puc.cl),
Chile.
3. "Visualización del balance de carga en un sistema distribuido"
Sergio Martig (smartig@cs.uns.edu.ar), Marcelo Vitturini (mmv@cs.uns.edu.ar),
Carlos Alvez (cea@cs.uns.edu.ar), Silvia Castro (smc@cs.uns.edu.ar), Javier Echaiz
(je@cs.uns.edu.ar), Jorge Ardenghi (jra@cs.uns.edu.ar), Argentina.
4. "Un algoritmo de scheduling para grafos arbitrarios sobre multiprocesadores con
memoria compartida",
Irene Zuccar (izuccar@yahoo.com), Mauricio Solar (msolar@diinf.usach.cl), Victor

Parada (vparada@diinf.usach.cl), Chile.

5. "Técnicas de detección de intrusos en sistemas computacionales: Perspectiva biológica"
Pedro Pinacho D. (ppinacho@udec.cl), Ricardo Contreras (rcontrer@inf.udec.cl), Chile.
6. "Algoritmos Genéticos Paralelos usando PVM"
Horst H. von Brand (vonbrand@inf.utfsm.cl), Eduardo Larrea, Chile.
7. "Un servicio de nombres para objetos en la Internet"
Carolina Bonacic Castro (bonacic@ona.fi.umag.cl), Carlos Arias Méndez (carias@ona.fi.umag.cl), Chile.
8. "ANE (Active Network Environment): An introduction"
Flavio Arnaldo Braga da Silva (flavio@din.uem.br), Joao Bosco Mangueira Sobral (jbosco@inf.ufsc.br), Emilio Eiji Yamane (eeyamane@din.uem.br), Rodrigo Campiolo (rcampiol@din.uem.br), Brasil.
9. "Definicao e validacao de um modelo para suporte de arquivos replicados"
Cristiano Bartolini (26877@lci.upf.tche.br), Emerson Rogrio de Oliveira Junior (emerson@upf.tche.br), Brasil.
10. "Exploración de puertos TCP vía IP"
Alfredo Naime (anaime@leon.ciens.ucv.ve), Marco De Vivo (mdevivo@reacciun.ve), Gabriela De Vivo (gdevivo@reacciun.ve), Germinal Isern (isern@ccs.neu.edu), Venezuela.
11. "Diseño de un modelo de costos para bases de datos en memoria principal"
José Luis Martí (jmarti@inf.utfsm.cl), Horst H. Brand (vonbrand@inf.utfsm.cl), Chile.

Web

1. "Newtenberg: Un modelo e implementación de un sistema de publicaciones digitales en la web"
Carlos Castillo Ocaranza (ccastill@dcc.uchile.cl), Chile.
2. "Evaluación heurística de un sitio de Internet"
F. Javier Diaz (jdiaz@ada.info.unlp.edu.ar), Ivana Harari (iharari@ada.info.unlp.edu.ar), Argentina.
3. "La versatilidad de PHP como herramienta de desarrollo en la Web"
Jorge Mansilla, Alejandro Ojeda, José Canumán (jcanuman@ona.fi.umag.cl), Chile.
4. "Experiencia con una aplicación web multimedial y de realidad virtual."
Daniel Cunsille (daniel@labsd.inf.utfsm.cl), Claudio Delgado (delgado@labsd.inf.utfsm.cl), Juan P. Menichetti (juanpablo@labsd.inf.utfsm.cl), Jorge Mujica (jorge@labsd.inf.utfsm.cl), Chile.
5. "Negociacao de preco em comercio electronico: Uma investigacao experimental"
Cristina Duarte Murta (cristina@inf.ufpr.br), Beatriz Terezinha Borsoi (beatriz@pb.cefetpr.br), Brasil.

Interfaz humano-computador y computación gráfica

1. "Descomposición de poligonos concavos"
Pedro Rodríguez M. (prodrigu@rayen.face.ubiobio.cl), Nancy Hitschfeld K. (nancy@dcc.uchile.cl), Chile.
2. "¿Gráfica o texto: cuál es mejor para modelar problemas lineales mixtos?",
Liubov Dombrovskaia (liuba@inf.utfsm.cl), Dietmar Janetzko (dietmar@cognition.iig.uni-freiburg.de), Chile.
3. "A new multilevel thresholding method"
Paulo Roberto Martins (paulorm@pr.gov.br), Jacques Facon (facon@ppgia.pucpr.br), Brazil.
4. "MEDITE: uma abordagem baseada em modelos para a concepcao de interfaces ergonómicas"
Claudia Verónica Serey Guerrero (cvserey@dsc.ufpb.br), Bernardo Lula Jr. (lula@dsc.ufpb.br), Brasil.

Lenguajes y herramientas de software

1. "Um analisador sintactico generico para linguagens livres de contexto"
Yandre Maldonado e Gomes da Costa (yandre@din.uem.br), Brasil.
2. "Lenguaje visual para la clasificación e indexación colaborativa de material multimedial"
Oriél Andrés Herrera Gamboa (oherrera@uct.cl), David Fuller Padilla (dfuller@ing.puc.cl), Chile.
3. "Elementos de un Lenguaje para el tratamiento de poblaciones computacionales basado en cuantificación existencial"
Esteban Osses (eosessa@inf.udec.cl), Andreas Polymris (apolymer@inf.udec.cl), Chile.
4. "De especificaciones declarativas a programas eficientes: Un caso de estudio"
Carlos Daniel Luna (cluna@fing.edu.uy), Uruguay.
5. "Portando el compilador lcc a linux"
Marcos Ramírez (marcos@inf.utfsm.cl), Horst von Brand (vonbrand@inf.utfsm.cl), Chile.

Redes Neuronales

1. "Implementacao analogica de um conversor A/D baseado nos neuronios de Hopfield"
José Homero Feitosa Cavalcanti (homero@ct.ufpb.br), Brasil.
2. "Robust neural network modeling for time series prediction"
Héctor Allende (hallende@inf.utfsm.cl), Claudio Moraga (moraga@cs.uni-dortmund.de), Chile.
3. "Redes neuronales y árboles de decisión para extraer conocimiento de la permanencia"

de pacientes en unidades médico-quirúrgicas"

Max Chacón (mchacon@diinf.usach.cl), Nayibe Nuñez (nnunez@diinf.usach.cl), Soledad Bastias (sbastias@optimisa.cl), Chile.

4. "Uso de redes neuronais LVQ para reconhecimento de solucoes logicas no ensino-aprendizagem de algoritmos"
Eliane Moreira S de Souza (elianems@newsite.br), Fernando Ostuni GAUTHIER (gauthier@eps.ufc.br), Otvio Roberto Martins de SOUZA (otaviors@newsite.com.br), Brasil.
5. "Uma proposta de uso de tecnologia adaptativa para la simulacao de redes neuronais em um dispositivo computacional"
Ricardo Luis de Azevedo da Rocha (Ricardo-Rocha@acm.org), Brasil.

Aplicaciones

1. "Generación automática de la guía académica de la Universidad Católica del Norte"
Loreto Talgie (ltelgie@ucn.cl), Chile.
2. "Pressure measurement with an instrumental pen for signature verification with application of segmentation, spectral analysis and neural network"
Tham Heng Keit (hltham@tm.net.my), R. Palaniappan, R. Raveendran, Fumiaki Takeda (takeda@info.kochi-tech.ac.jp), Malasia.
3. "Reduzindo as ilhas de dados em subestacoes"
Gilberto Grandi (Grandi@inf.univali.br), Fernando Ostuni Gauthier (Gauthier@eps.ufsc.br), Brasil.
4. "Un estudio experimental en la dinámica de poblaciones"
Mirta Padín, Claudio Delrieux (claudio@acm.org), Julián Domínguez, Héctor Soto, Argentina.
5. "Evaluación transiente de la distribución del tiempo operacional acumulado en sistemas reparables"
Reinaldo A. Vallejos C. (reinaldo@elo.utfsm.cl), José M. Martínez V. (josemm@dcsc.utfsm.cl), Chile.
6. "Métodos Bayesianos para la restauración de imágenes digitales de anillos de una troza de árbol"
Aliana Scheihing (escheihi@inf.uach.cl), Chile.
7. "Fuzzy Clustering: Criterion to class deputation"
Celso Alberto Rojas Pukall (albertop@politec.une.edu.py), Paraguay.

Informática médica

1. "Informática Médica: Resideño social y técnico de procesos en la aplicación de la medicina"
Graciela Elisa Barchini (grael@unse.edu.ar), Paola Daniela Budán, Argentina.
2. "La transformada de Wavelet continua aplicada a la representación de señales electroencefalográficas en el plano tiempo-frecuencia"
T. Mardesic (mmardesic@inf.utfsm.cl), L. Salinas (lsalinasinf.utfsm.cl), F. Schifferli

(schifferli@entelchile.net), Chile.

3. "WRAD -- Uma ferramenta para auxiliar no ensino da radiologia baseada em Web"
Robinson Pizzio (rpizzio@puhrs.campus2.br), Marcelo Muniz da Silva
(muniz@puhrs.campus2.br), José Euclides Rodrigues Beltran
(beltran@puhrs.campus2.br), Brasil.
4. "FIC X WTC: how much could we shrink a medical image?"
Robinson Pizzio (rpizzio@puhrs.campus2.br), Paulo R. G. Franco
(pfranco@ee.puhrs.br), Brazil.

Herramienta para Metamodelización de Procesos Software

Luis MÁRQUEZ, Félix GARCÍA, Francisco RUIZ, Mario PIATTINI, Macario POLO
Grupo Alarcos
Escuela Superior de Informática, Universidad de Castilla-La Mancha
Ronda de Calatrava, s/n. 13071, Ciudad Real (España)
lmarquez@proyectos.inf-cr.uclm.es; {fgarcia|fruiiz}@inf-cr.uclm.es

Palabras clave

Metamodelos de Procesos Software, MOF, XMI.

Resumen

La gestión de procesos software es una actividad compleja. Debido al gran número de diferentes aspectos que se deben considerar es útil establecer una arquitectura conceptual. El uso de arquitecturas que incluyan metadatos nos permite manejar la complejidad de procesos software.

En este artículo presentamos una arquitectura conceptual de 4 niveles para representar y gestionar metamodelos de procesos software basados en el estándar MOF (Meta Object Facility) propuesto por el OMG (Object Management Group) (OMG, 2000). En particular, aplicaremos estos conceptos al Proceso de Mantenimiento del Software (PMS).

Como soporte presentamos MANTIS-Metamod, una herramienta para el modelado de procesos software basada en los conceptos comentados previamente. MANTIS-Metamod es un componente de MANTIS, un entorno integral para la gestión del PMS. Aunque MANTIS está dedicado al PMS, la herramienta permite la meta-modelización de cualquier proceso software.

1- Introducción

Hasta hace poco la construcción de software se realizaba como una tarea artesanal, donde el éxito de los proyectos dependía principalmente del "arte" de los desarrolladores y no en que fueran desarrollados con un enfoque ingenieril.

Para resolver esta situación surge la ingeniería del software, cuyo principal objetivo es la obtención de productos software de calidad. Un elemento fundamental que influye en la calidad de los productos es la forma de desarrollarlos y mantenerlos, es decir los procesos involucrados tanto en el desarrollo como en el mantenimiento del software.

Según la norma ISO 12207 un proceso es un conjunto de actividades interrelacionadas que transforman entradas en salidas (ISO/IEC, 1995). Un proceso define quién está haciendo qué, cuándo, y cómo alcanzar un determinado objetivo.

Los procesos software son inherentemente complejos. Involucran a mucha gente, cada uno con responsabilidades y habilidades individuales, y producen o modifican un amplio rango de elementos (Becker et al., 1999).

El estándar ISO 12207 "establece un marco de referencia común para los procesos del ciclo de vida software, con una terminología bien definida, que puede ser referenciada por la industria software" (ISO/IEC 1995). En este marco se definen los procesos, actividades (que forman cada proceso) y tareas (que constituyen cada actividad) presentes en la adquisición, suministro, desarrollo, operación y mantenimiento del software

En la figura 1 se presenta la norma ISO 12207, que agrupa las actividades que pueden ser realizadas durante el ciclo de vida del software en cinco procesos principales, ocho procesos de soporte, y cuatro procesos organizativos, así como el proceso de adaptación.

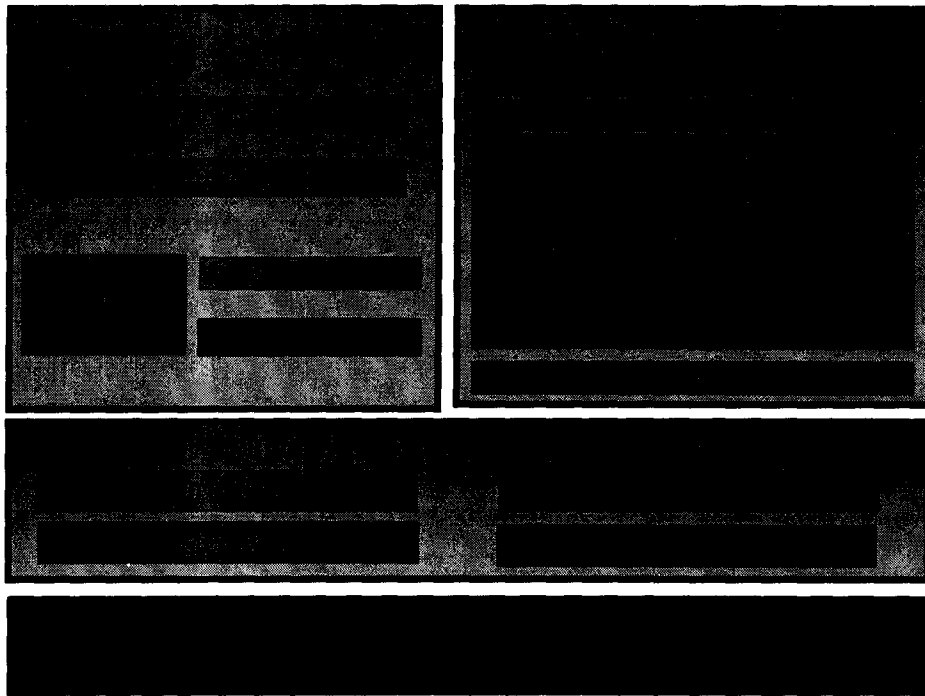


Figura 1. Procesos del ciclo de vida software según la norma ISO 12207

Sin lugar a dudas, y a vistas de la experiencia, la comunidad científica internacional ha reconocido la importancia, especialmente económica, que tiene el Proceso de Mantenimiento del Software. Diversos estudios han demostrado que la etapa de mantenimiento supone la mayor parte de los costes totales de un producto software a lo largo de su ciclo de vida (entre el 60 y el 80 por ciento) (Pigoski, 1996) y que sus características especiales, que lo diferencian significativamente del desarrollo, hacen muy útil disponer de métodos, técnicas y herramientas específicos (Piattini et al., 1998).

Por las razones citadas, últimamente se han llevado a cabo esfuerzos para abordar el problema del mantenimiento del software teniendo en cuenta sus características especiales. Debido a que la gestión del PMS es un proceso complejo, es necesario adoptar las medidas oportunas para conseguir simplificarlo y manejarlo de forma organizada.

2- Metamodelado de Procesos Software

Ante esta situación nace MANTIS (Ruiz et al., 2001), un proyecto de Investigación y Desarrollo que pretende la definición y construcción de un entorno para la gestión integral del PMS. En MANTIS se define la forma en que se organiza, gestiona, mide, soporta y mejora el PMS.

Para la gestión del PMS, MANTIS integra, entre otros, a las personas (con ciertas habilidades y que desempeñan ciertos roles en el proyecto), las técnicas (metodologías) empleadas por las personas, las herramientas (que ayudan a cumplir los estándares) y las actividades (en las que participan los equipos y que ayudan a alcanzar hitos significativos).

Para manejar mejor la complejidad del PMS parece conveniente establecer una arquitectura conceptual usando un principio importante de la ingeniería de software moderna: la separación de un sistema en capas de encapsulación que pueden ser especificadas, diseñadas y construidas independientemente.

En MANTIS se definen 4 niveles conceptuales que están basados en el estándar MOF (Meta Object Facility) para metamodelización con orientación a objetos (OMG, 2000), propuesto por el Object Management Group. En la tabla 1 se muestran estos cuatro niveles de la arquitectura MOF y su adaptación a MANTIS.

Nivel	MOF	MANTIS
M3	Modelo MOF (Meta-metamodelo)	Modelo MOF
M2	Meta-modelo	Metamodelo PMS
M1	Modelo	Metodología MANTEMA* y otras técnicas (Modelo concreto del PMS)
M0	Datos	Instancias de PMS (proyectos concretos de mantenimiento del software)

Tabla 1. Niveles Conceptuales en MOF y MANTIS.

En el nivel M0 están los ejemplares de proyectos reales y concretos de mantenimiento del software con restricciones de tiempo, costes, etc. Los datos manejados en este nivel son instancias de los conceptos definidos en el nivel superior M1. El modelo concreto que utilizamos en el nivel M1 está basado en la metodología MANTEMA y en un conjunto de técnicas adaptadas a las particularidades del mantenimiento: estimación de esfuerzo, estimación de riesgos, auditoría del proceso (Ruiz et al., 2000), etc. El nivel M2 se corresponde con el metamodelo del PMS (algunos autores no distinguen claramente en la nomenclatura entre modelo y metamodelo llamando modelo tanto al correspondiente al nivel M2 como al del nivel M1). Por ejemplo, el concepto genérico de Activity utilizado en M2, se instancia en las actividades "Análisis de Informe de Problemas" o "Implementación de la Modificación" en M1, y éstas a su vez, en instancias del nivel M0 como "Análisis de informe del problema nº 36 del proyecto PATON".

En el último nivel conceptual de MANTIS, el M3, el metamodelo de PMS es representado en un modelo-MOF. Un modelo-MOF está formado básicamente por dos tipos de objetos: clases-MOF y asociaciones-MOF (estos son los principales objetos para nosotros, aunque también existen otros: paquetes para fines de reutilización, tipos de datos, ...). Por tanto, todos los conceptos representados en el nivel M2 se consideran ahora ejemplares de clases-MOF o de asociaciones-MOF. Por ejemplo, "Actividad", "Actor" o "Artefacto" serán ejemplares de clase-MOF; y "Actividad usa Recursos" o "Artefacto es_entrada_de Actividad" son ejemplares de asociación-MOF. Un modelo-MOF se puede representar, entre otros, mediante diagramas UML o mediante el lenguaje MODL (*Meta Object Definition Language*), pero para su manejo automático y su portabilidad entre herramientas en un (EIS), que es lo que interesa en MANTIS, es mucho mejor representarlo utilizando algún estándar de intercambio de metadatos. Por esta razón, en MANTIS se utiliza XMI (*XML Metadata Interchange*) (OMG, 1999), basado en XML (*eXtensible Markup Language*), para almacenar los metamodelos.

MOF es un modelo para especificar, construir, gestionar, intercambiar e integrar metadatos en sistemas software, permitiendo una flexible integración de sistemas. MOF describe un lenguaje de modelado abstracto alineado con el núcleo del UML de OMG.

En la siguiente figura, se representa mediante un diagrama UML un resumen de la jerarquía de clases que componen el modelo MOF.

* MANTEMA es una metodología para el mantenimiento de software que integra todas las actividades relacionadas con este proceso (Polo et al., 1999)

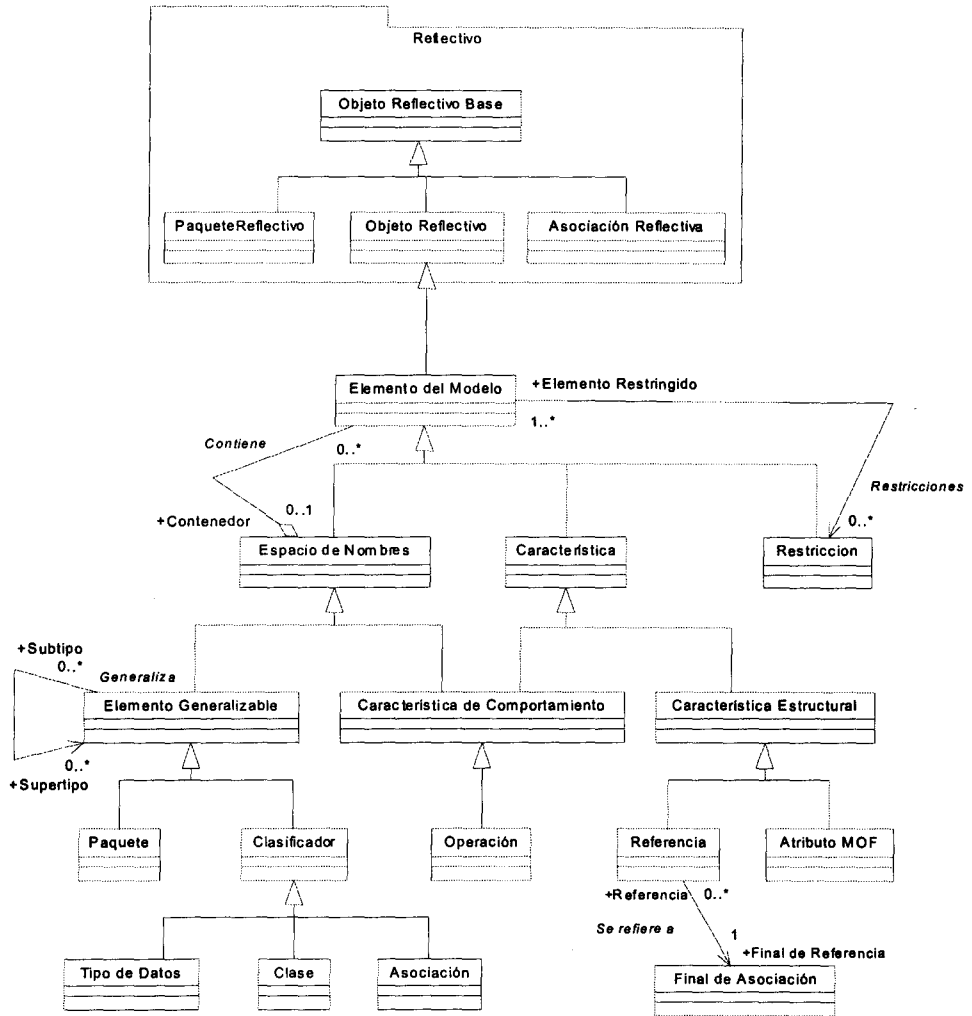


Figura 1. Jerarquía Clases MOF.

3- MANTIS-Metamod

Debido a la complejidad inherente al PMS, parece necesario contar con una herramienta automática que nos ayude en la gestión del mismo.

En este contexto surge MANTIS-Metamod, una herramienta que proporciona soporte automático para el modelado del PMS basada en el estándar MOF. Esta herramienta permite trabajar con los cuatro niveles conceptuales que describe dicho estándar.

Puesto que MOF permite la modelización de cualquier universo del discurso, MANTIS-Metamod va a permitir representar metamodelos de procesos software en general, aunque en nuestro trabajo (enmarcado dentro del proyecto MANTIS) se aplica para el modelado del PMS.

El objetivo que se persigue con esta herramienta es facilitar la gestión integral del Proceso de Mantenimiento del Software (PMS) mediante la definición de metamodelos y modelos necesarios en dicha gestión, basándonos en una terminología común y en el mecanismo de abstracción que proporciona la arquitectura de niveles de MOF.

La herramienta está basada en una arquitectura de 3 capas (3- Tier Architecture) (Larman, 1998). Por lo tanto, la arquitectura de la herramienta está formada por 3 capas verticales, con el fin de reducir la complejidad y proporcionar un cierto grado de encapsulación. Las tres capas verticales son:

- **Presentación o Interfaz:** ventanas, reportes, etcétera.
- **Lógica de aplicaciones o Dominio:** tareas y reglas que rigen el proceso.
- **Almacenamiento:** mecanismo de almacenamiento persistente.

La capa de interfaz es la encargada de obtener información del usuario, enviar la información del usuario para su procesamiento, recibir los resultados del procesamiento y presentación de los resultados al usuario. La capa de dominio tiene como funciones recibir la entrada del nivel de presentación, interactuar con los servicios de almacenamiento para ejecutar las operaciones, y enviar el resultado procesado al nivel de presentación. Finalmente, la capa de almacenamiento es la encargada del almacenamiento, recuperación y mantenimiento de los datos.

Para la entrada datos del usuario, la aplicación está compuesta por administradores de modelos y metamodelos como componentes principales, y por sistemas de ventanas que permiten la descripción de forma visual de las distintas proyecciones de dichos modelos y metamodelos.

Por ejemplo, los principales elementos atendiendo a la proyección entre los niveles M3-M2 son un administrador de metamodelos y un sistema de ventanas que permite la descripción de forma visual de las clases que forman el núcleo del modelo MOF (Paquete, Clase, Tipo de Datos, Atributo, Operación, Referencia, Asociación, Final de Asociación y Restricción).

El administrador de metamodelos, al igual que el modelo MOF, está estructurado en forma de árbol: un paquete contiene clases y asociaciones, una clase contiene atributos y operaciones, una asociación contiene restricciones etc.

En la siguiente figura puede verse el administrador de metamodelos así como el sistema de ventanas asociado que nos permite la descripción de las clases que forman el núcleo del modelo MOF.

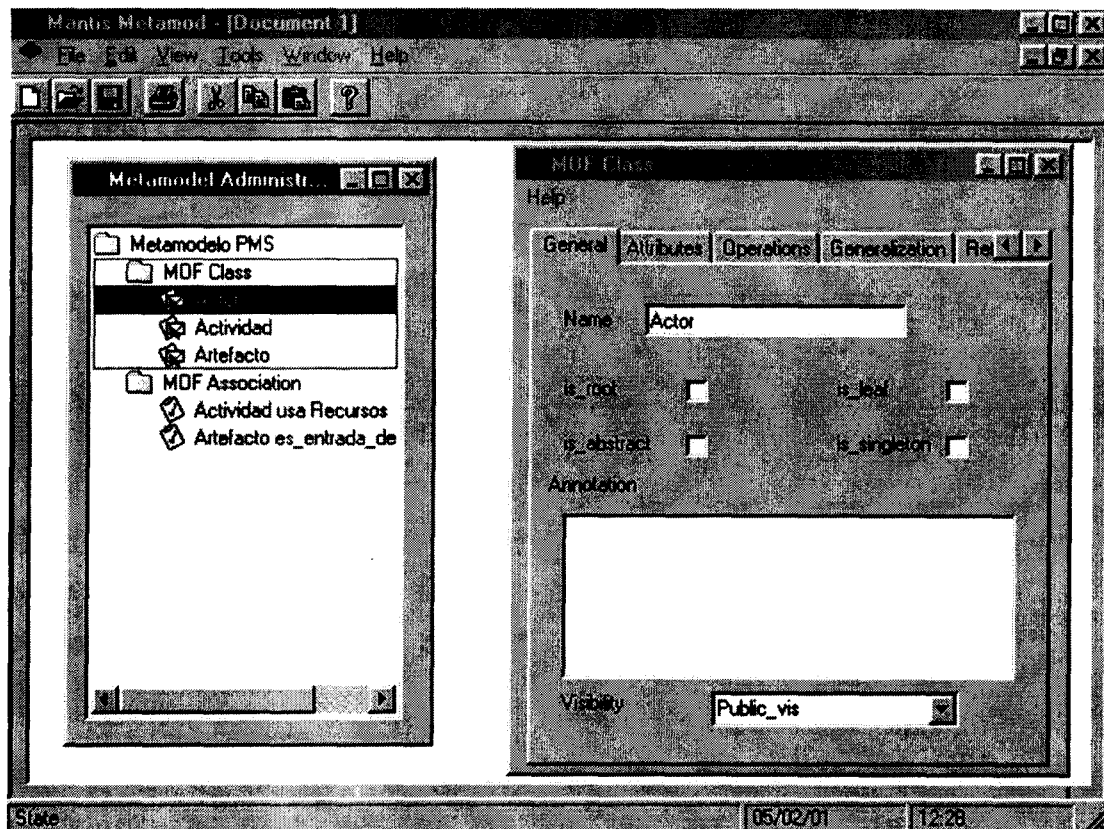


Figura 2- Aspecto de Mantis Metamod

Por ejemplo, una instancia de clase MOF podría ser la clase del nivel M2 "Actor" perteneciente al metamodelo de PMS. Dicha instancia de clase MOF, como se ve en la figura, permite que cualquier elemento del modelo pueda usarla (visibility=public_vis), no es abstracta (is_abstract=false), puede tener supertipos (is_Root=false), puede ser supertipo (is_Leaf=false) y como mucho puede existir una instancia del nivel M1 (is_singleton=false).

Los modelos definidos por el usuario a través de la interfaz, son validados y se representan internamente según la jerarquía de clases que describe el modelo MOF, especificada en el documento IDL adjunto al estándar MOF.

Para el almacenamiento de los modelos y metamodelos se debe usar una representación que facilite el intercambio de información y, por lo tanto, proporcione una gran flexibilidad a la herramienta. Para tal fin se usa XMI (OMG, 1999). El principal propósito de XMI es permitir fácil intercambio de metadatos entre herramientas de modelización (basadas en UML) y entre herramientas y repositorios de metadatos (basados en MOF) en entornos distribuidos heterogéneos. XMI usa XML como sintaxis para la transferencia y formato de intercambio.

Para la gestión de datos y metadatos almacenados en documentos XMI hemos desarrollado un componente vertical, al que hemos llamado gestor del repositorio.

El gestor del repositorio de metadatos es un componente diseñado para proveer la infraestructura y soporte necesarios para el almacenamiento de componentes de información interrelacionados. Por tanto las funciones básicas

del gestor del repositorio son las mismas que las que proporciona un sistema gestor de bases de datos con la particularidad de que lo que se almacenan son documentos XMI. Estas funciones son soportadas mediante un conjunto de llamadas al sistema, y son básicamente:

- 1- Almacenamiento de los modelos MOF definidos en la herramienta en un repositorio local de metadatos representados en XMI (para exportación).
- 2- Importación de metamodelos.

En la siguiente figura se puede observar el esquema básico del gestor del repositorio con sus funciones principales:

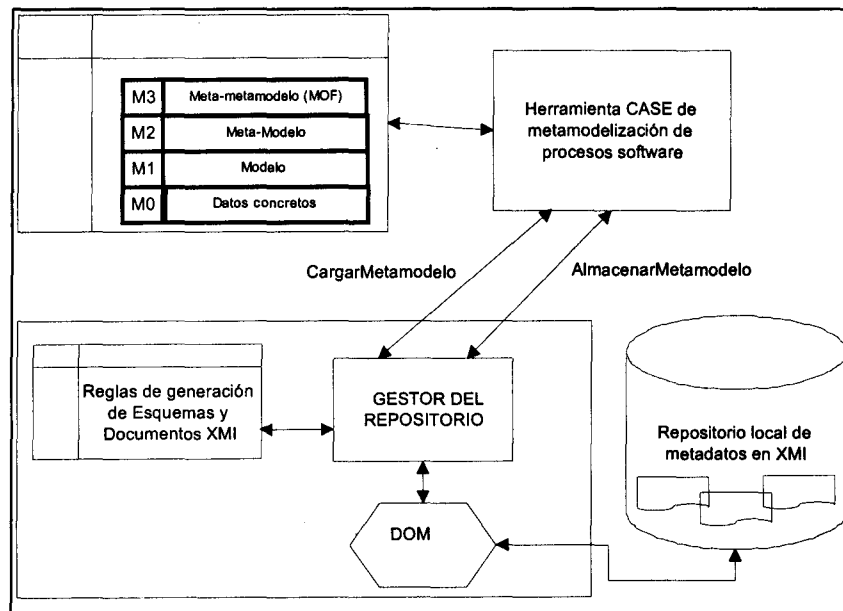


Figura 3- Arquitectura general del gestor del repositorio

Como se puede observar en la figura 3, el gestor proporciona a la aplicación dos servicios básicos, que son el almacenamiento y la importación de metamodelos.

Para la manipulación del repositorio, es decir, para el almacenamiento y recuperación de la información a nivel de documento se usa el XML DOM (Document Object Model). Este modelo proporciona una colección de clases que representan la estructura jerárquica en que se basa todo documento XML.

El gestor del repositorio dispone de una interfaz claramente definida para la comunicación con la capa de dominio.

4- Conclusiones

En este artículo hemos presentado MANTIS-Metamod, herramienta para la descripción, importación y exportación de metamodelos de procesos software.

El objetivo perseguido con este componente de MANTIS es sacar provecho de las ventajas probadas que supone el uso de metamodelos para la gestión integral de procesos software. En MANTIS dicha integración está basada en el uso de una terminología común y en el mecanismo de abstracción que proporciona la arquitectura de niveles conceptuales descrita por el estándar MOF.

Una utilidad fundamental de la herramienta es que constituye el soporte necesario para la mejora de procesos software. En el contexto en que se enmarca esta herramienta es posible, por ejemplo, gracias a la definición de las proyecciones adecuadas entre el nivel 3 y el nivel 2 de MOF, incorporar al metamodelo de proceso de mantenimiento del software (PMS) la subontología de la medida, mediante la agregación de las clases y asociaciones necesarias.

Al usar XMI como formato de representación e intercambio, toda herramienta de modelado que utilice el lenguaje abstracto que proporciona MOF y XMI para la definición y almacenamiento de metamodelos podrá colaborar con la herramienta propuesta aumentando la eficacia de la gestión de procesos software.

Agradecimientos:

Este trabajo está realizado en colaboración con la empresa Atos ODS con cargo a los proyectos MANTIS y MPM. MANTIS ha sido parcialmente financiado por la Union Europea y CICYT-España (1FD97-1608TIC). MPM ha sido parcialmente financiado por el Ministerio de Ciencia y Tecnología de España (FIT-070000-2000-307).

Referencias:

- (Becker y Webby, 1999) Becker-K., U., y Webby, R., *A Comprehensive Schema Integrating Software Process Modeling and Software Measurement*, Fraunhofer Institute, IESE report N° 047.99/E. v1.2, 1999.
- (ISO/IEC, 1995) ISO/IEC 12207: Information Technology - Software Life Cycle Processes, 1995.
- (ISO/IEC, 2000) ISO/IEC JTC1/SC7/WG4 15940 working draft 5: *Information Technology - Software Engineering Environment Services*, juny-2000.
- (Larman, 1998) Larman, C. *Applying UML and Patterns*. Up-per Saddle River, NJ: Prentice-Hall, 1998.
- (OMG, 1999) *OMG XML Metadata Interchange (XMI)*, v. 1.1, oct-1999.
- (OMG, 2000) *OMG Meta Object Facility (MOF) Specification*, v. 1.3 RTF, sep-1999. In <http://www.omg.org>.
- (Penadés et al., 1999) Penadés, M.C.; Canós, J.; Carsí, J.A.; *Hacia una Herramienta de Soporte al Proceso Software basada en la tecnología de Workflow*. IV Jornadas de Ingeniería del Software y Bases de Datos; Cáceres, España, 1999.
- (Piattini et al., 1998) Piattini, M.; Ruiz, F.; Polo, M., Bastanchury, T., Fernández, I., Martínez, M.A. *Mantenimiento del Software: Conceptos, Métodos, Herramientas y Outsourcing*. Ed Ra-Ma, España 1988.
- (Pigoski, 1996) Pigoski, T.M., *Practical Software Maintenance. Best Practices for Managing your Investment*. Ed. John Wiley & Sons, USA 1996.

- (Polo et al., 1999) Polo, M.; Piattini, M.; Ruiz, F.; Calero, C.; *MANTEMA versión 2.0: una Metodología para el Mantenimiento del Software*. UCLM, Dep. de Informática, informe técnico UCLM-DI-99-01.
- (Ruiz et al., 2000) Ruiz, F.; Piattini, M.; Polo, M.; Calero, C.; Audit of Software Maintenance. In "Auditing Information Systems". Idea Group Publishing, USA 2000.
- (Ruiz et al., 2001) Ruiz, F., Piattini, M., García, F. y Polo, M. "Metamodelos y Flujos de Trabajo para la Gestión del Proceso de Mantenimiento del Software". 4ª Jornadas Iberoamericanas de Ingeniería de Requisitos y Ambientes de Software. San José (Costa Rica), pp. 132-142.
- (WfMC, 1995) WfMC TC00-1003 1.1: Workflow Management Coalition. *The WorkflowReference Model*, jan-1995.