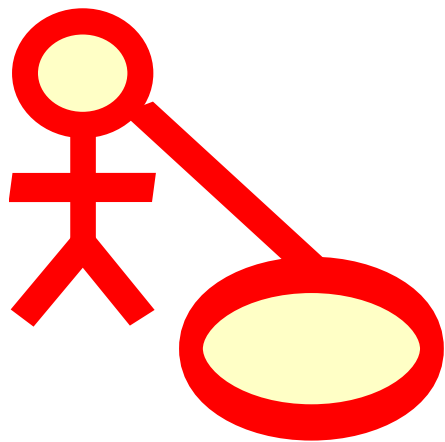


Manual de Umbrello UML Modeller



Manual de Umbrello UML Modeller

Índice xeral

1. Introducción	7
2. Fundamentos de UML	8
2.1. Acerca de UML	8
2.2. Elementos UML	9
2.2.1. Diagrama de caso de uso	9
2.2.1.1. Caso de uso	9
2.2.1.2. Actor	10
2.2.1.3. Descripción do caso de uso	10
2.2.2. Diagramas de clase	10
2.2.2.1. Clase	11
2.2.2.1.1. Atributos	11
2.2.2.1.2. Operacións	11
2.2.2.1.3. Modelos	11
2.2.2.2. Asociacións entre clases	11
2.2.2.2.1. Xeneralización	12
2.2.2.2.2. Asociacións	12
2.2.2.2.3. Agregación	12
2.2.2.2.4. Composición	13
2.2.2.3. Outros elementos dos diagramas de clase	13
2.2.2.3.1. Interfaces	13
2.2.2.3.2. Tipos de datos	13
2.2.2.3.3. Enumeracións	13
2.2.2.3.4. Paquetes	13
2.2.3. Diagramas de secuencia	13
2.2.4. Diagramas de colaboración	14
2.2.5. Diagrama de estado	15
2.2.5.1. Estado	16
2.2.6. Diagrama de actividade	16
2.2.6.1. Actividade	17
2.2.7. Elementos auxiliares	17
2.2.8. Diagrama de componentes	18
2.2.9. Diagramas de implementación	18

2.2.10. Diagramas de relacións entre entidades	18
2.2.10.1. Entidade	19
2.2.10.1.1. Atributos da entidade	19
2.2.10.1.2. Restricións	19
2.2.10.2. Conceitos dos diagramas de relacións ampliadas entre entidades (EER)	20
2.2.10.2.1. Especialización	20
2.2.10.2.1.1. Especialización disxunta	20
2.2.10.2.1.2. Especialización sobreposicionante	21
2.2.10.2.1.3. Categoría	21
3. Traballar con Umbrello UML Modeller	23
3.1. A interface do usuario	23
3.1.1. A vista en árbore	24
3.1.2. Fiestra de documentación	24
3.1.3. Área de traballo	24
3.2. Criar, cargar e gardar modelos	25
3.2.1. Novo modelo	25
3.2.2. Gardar o modelo	25
3.2.3. Cargar modelo	25
3.3. Editar modelos	25
3.4. Engadir e eliminar diagramas	26
3.4.1. Criar diagramas	26
3.4.2. Eliminar diagramas	26
3.4.3. Mudar o nome dos diagramas	26
3.5. Editar diagramas	26
3.5.1. Inserir elementos	27
3.5.2. Borrar elementos	27
3.5.3. Editar elementos	27
3.5.4. Editar clases	28
3.5.4.1. Configuración xeral da clase	28
3.5.4.2. Configuración dos atributos da clase	28
3.5.4.3. Configuración das operacións da clase	28
3.5.4.4. Configuración do modelo da clase	28
3.5.4.5. Páxina de asociacións da clase	28
3.5.4.6. Páxina de visualización da clase	29
3.5.4.7. Páxina de cor da clase	29
3.5.5. Asociacións	29
3.5.5.1. Pontos de enganche	29
3.5.6. Notas, texto e caixas	30
3.5.6.1. Enganches	30

4. Importación de fontes e xeración de código fonte	31
4.1. Xerazón de fontes	31
4.1.1. Xerar o código fonte	31
4.1.1.1. Opcións de xerazón	32
4.1.1.1.1. Verbosidade do código	32
4.1.1.1.2. Cartafoles	32
4.1.1.1.3. Política de sobrescritura	33
4.1.1.1.4. Linguaxe	33
4.1.1.2. Asistente de xerazón de código	33
4.2. Importación de código	33
5. Outras funcionalidades	35
5.1. Outras funcionalidades de Umbrello UML Modeller	35
5.1.1. Copiar obxectos como imaxes PNG	35
5.1.2. Exportar para unha imaxe	35
5.1.3. Imprimir	35
5.1.4. Cartafoles lóxicos	35
6. Autores e historial	37
7. Copyright	38

Resumo

Umbrello UML Modeller axúdao no desenvolvemento de sóftware usando o estándar industrial chamado Unified Modelling Language (UML) que lle permite crear diagramas para o deseño e documentación dos sistemas.

Capítulo 1

Introdución

Umbrello UML Modeller é unha ferramenta de diagramas UML que pode axudalo no proceso de desenvolvemento de sóftware. En particular durante as fases de análise e deseño, Umbrello UML Modeller axudarao a obter un produto de alta calidade. UML tamén pode ser usado para documentar os seus deseños de sóftware, como axuda para si e para os compañeiros no desenvolvemento.

Ter un bon modelo do seu sóftware é a mellor maneira de comunicarse con outros desenvolventes que traballen no proxecto e cos seus clientes. Un bon modelo é extremadamente importante para proxectos grandes e medianos, pero tamén é moi útil para os pequenos. Mesmo se está a traballar nun pequeno proxecto dunha só persoa beneficiarase de ter un bon modelo porque lle dará unha visión global que o axudará a programar ben as cousas á primeira.

UML é a linguaxe de diagramas usada para describer tais modelos. Pode representar as súas ideas en UML usando diferentes tipos de diagramas. Umbrello UML Modeller 1.2 permite os seguintes tipos:

- Diagrama de clases
- Diagrama de secuencia
- Diagrama de colaboración
- Diagrama de casos de uso
- Diagrama de estado
- Diagrama de actividade
- Diagrama de componentes
- Diagrama de implementación

Pode atopar máis información acerca de UML no sitio web de **OMG**, <http://www.omg.org> que foi quen criou o estándar UML.

Desexamos que desfrute de Umbrello UML Modeller e que o axude a crear sóftware de alta calidade. Umbrello UML Modeller é Sóftware Libre e está dispoñíbel de balde, a única cousa que lle pedimos é que informe de calquer erro, problema ou suxestión aos desenvolventes de Umbrello UML Modeller en uml-devel@lists.sourceforge.net ou <http://bugs.kde.org>.

Capítulo 2

Fundamentos de UML

2.1. Acerca de UML

Este capítulo daralle unha introdución rápida aos fundamentos de UML. Teña en conta que este non é un tutorial extenso de UML senón unha breve introdución que pode ser lida como un tutorial. Se quer aprender máis acerca do Unified Modelling Language, ou en xeral acerca da análise e deseño de sóftware, consulte un dos moitos libros dispoñíbeis que tratan deste tema. Tamén hai moitos tutoriais en internet que pode tomar como un punto de comezo.

O “Unified Modelling Language” (UML) é unha linguaxe ou notación de deseño para especificar, visualizar e documentar modelos de sistemas de sóftware Orientados a Obxectos. UML non é un método de desenvolvemento, o que significa que non lle di o que debe facer primeiro e o que debe facer despois nen como deseñar o seu sistema, senón que o axuda a visualizar o deseño e a comunicarllo a outros. UML está controlado polo “Object Management Group” (OMG) e é o estándar da industria para describer graficamente o sóftware.

UML foi pensado para o deseño de sóftware Orientado a Obxectos e ten unha utilidade moi limitada para outros paradigmas de programación.

UML está composto de diversos elementos de modelo que representan as diferentes partes dun sistema de sóftware. Os elementos UML son usados para crear diagramas, que representan unha parte, ou un punto de vista do sistema. Umbrello UML Modeller soporta os seguintes tipos de diagrama:

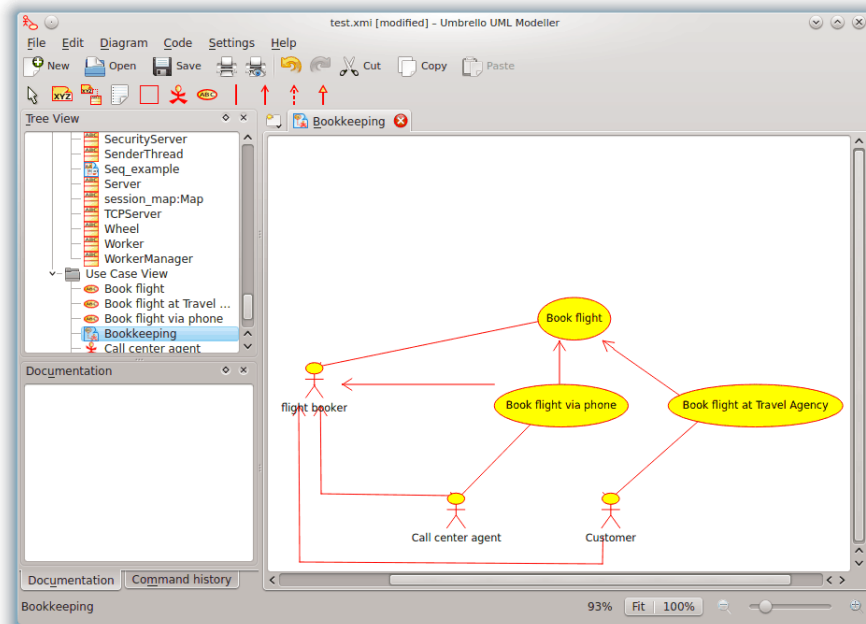
- *Diagramas de caso de uso* mostra actores (xente ou outros usuarios do sistema), casos de uso (os escenarios en que o sistema é usado), e as súas relacións
- *Diagramas de clase* mostra clases e as relacións entre elas
- *Diagramas de secuencia* mostra obxectos e a secuencia de chamadas a métodos de outros obxectos que fan.
- *Diagramas de colaboración* mostra obxectos e a súa relación, enfatizando os obxectos que participan no intercambio de mensaxes
- *Diagramas de estado* mostra estados, mudanzas de estado e eventos nun obxecto ou unha parte do sistema
- *Diagramas de actividade* mostra actividades e os cambios dunha activade para outra cos eventos que acontecen nalgunha parte do sistema
- *Diagramas de componentes* mostra as componentes de alto nivel (como KParts ou Java Beans).
- *Diagramas de implementación* mostra as instancias das componentes e as súas relacións.
- *Os Diagramas de relacións entre entidades* mostran datos e as relacións e restricións entre os datos.

2.2. Elementos UML

2.2.1. Diagrama de caso de uso

Os diagramas de caso de uso describen as relacións e dependencias entre un grupo de *Casos de Uso* e os Actores a participar no proceso.

É importante decatarse de que os diagramas de casos de uso non son apropiados para representar o deseño, e non poden describer o interior dun sistema. Estes diagramas serven para facilitar a comunicación cos futuros usuarios do sistema, e cos clientes, e son especialmente útiles para determinar as funcionalidades que o sistema debe ter. Os diagramas de casos de uso indican o *que* o sistema deba ter, pero non especifica como o acada, —nen pode—.



Umbrello UML Modeller mostrando un diagrama de caso de uso

2.2.1.1. Caso de uso

Un *caso de uso* describe —desde o punto de vista dos actores— un grupo de actividades nun sistema que produce un resultado concreto, tanxíbel.

Os casos de uso son descrições das interaccións típicas entre os usuarios dun sistema e o sistema en si. Representan a interface externa do sistema e especifican unha forma de requerimentos do que debe facer o sistema (lembre, só o que, non como).

Cando se traballa cos casos de uso, é importante lembrar algunhas regras simples:

- Cada caso de uso estará relacionado con polo menos un actor
- Cada caso de uso ten un iniciador (isto é un actor)
- Cada caso de uso conduce a un resultado relevante (un resultado con « valor de negocio »)

Os casos de uso tamén teñen relacións con outros casos de uso. Os tres tipos de relación máis típicas entre os casos son:

- «*inclusión*» que indica que un caso de uso acontece *dentro* doutro caso de uso
- «*extensión*» que indica que en certas situacións, ou nalgún punto (chamado punto de extensión) un caso de uso será ampliado por outro.
- *Xeneralización* indica que un caso de uso herda as características do « Super » caso de uso, e pode sobrescreber algunhas deles ou engadir outras novas, de xeito similar á herdanza entre clases.

2.2.1.2. Actor

Un actor é unha entidade externa (fora do sistema) que interactúa co sistema ao participar (e a miúdo iniciar) un caso de uso. Os actores poden ser xente real (por exemplo os usuarios do sistema), outros sistemas informáticos ou eventos externos.

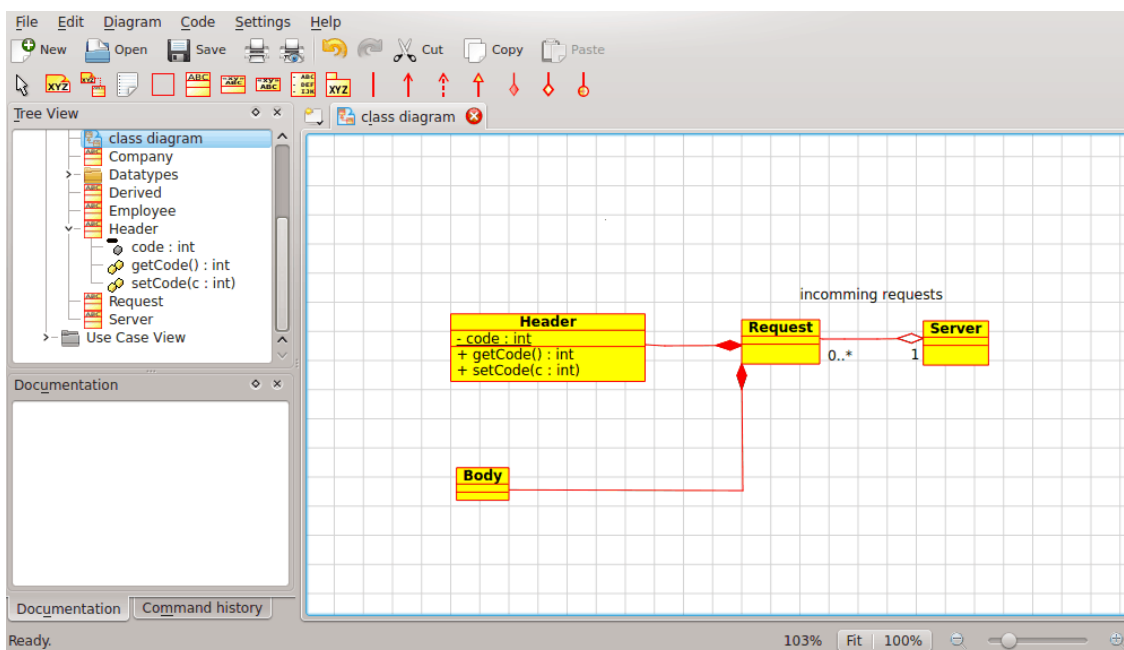
Os actores non representan a xente ou sistemas *físicos*, senón o seu *papel*. Isto significa que cando unha persoa interactúa co sistema de diferentes xeitos (asumindo diferentes papeis) estará representada por varios actores. Por exemplo unha persoa que dá servizo técnico aos clientes por teléfono e recibe pedidos dos clientes, no sistema estaría representado por un actor « Servizo Técnico » e un actor « Comercial »

2.2.1.3. Descrición do caso de uso

As descrições dos casos de uso son textos que o explican o caso. Polo xeral toman a forma dunha nota ou documento que está ligado dalgún xeito ao caso de uso, e explica os procesos ou actividades que acontecen nese caso de uso.

2.2.2. Diagramas de clase

Os diagramas de clase mostran as diferentes clases que conforman o sistema e como se relacionan entre si. Sóse dicir que os diagramas de clase son « estáticos » porque mostran as clases, xunto cos seus métodos e atributos así como as relacións estáticas entre elas: que clases « coñecen » da existencia de outras ou que clases « son parte » de outras clases, pero non mostran as chamadas a métodos entre elas.

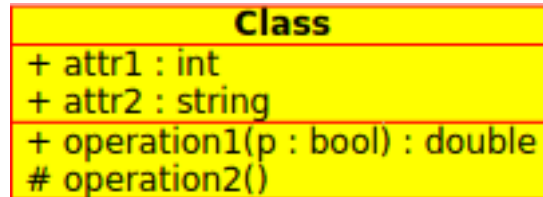


Umbrello UML Modeller mostrando un diagrama de clases

2.2.2.1. Clase

Unha clase define os atributos e métodos dun conxunto de obxectos. Todos os obxectos desa clase (instancias da clase) teñen o mesmo comportamento, e teñen o mesmo conxunto de atributos (cada obxecto ten o seu propio conxunto). Ás veces chámaselles « Tipo » no canto de Clase, pero é importante mencionar que non son o mesmo, e que Tipo é un termo máis xeral.

En UML as clases son representadas por rectángulos, co nome da clase, que poden mostrar tamén os atributos e operacións da clase en outros dous « compartimentos » dentro do rectángulo.



Representación visual dunha clase en UML

2.2.2.1.1. Atributos

En UML os atributos son mostrados con polo menos o seu nome, e tamén pode mostrar o tipo, valor inicial e outras propiedades. Os atributos tamén poden ser mostrados coa súa visibilidade:

- + Significa que o atributo é *público*
- # Significa que o atributo é *protexido*
- - Significa que o atributo é *privado*

2.2.2.1.2. Operacións

As operacións (métodos) tamén son mostradas con polo menos o seu nome, e tamén pode mostrar os seus parámetros e tipo de retorno. Os igual que cos atributos, tamén pode ver a súa visibilidade:

- + Significa que a operación é *pública*
- # Significa que a operación é *protexida*
- - Significa que a operación é *privada*

2.2.2.1.3. Modelos

As clases poden ter modelos, un valor que é usada para unha clase ou tipo non especificado. O tipo do modelo é especificado cando a clase é iniciada (isto é cando se cría un obxecto). Os modelos existen en C++ e serán introducidos en Java 1.5 onde serán chamados Generics.

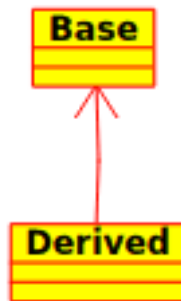
2.2.2.2. Asociacións entre clases

As clases poden relacionarse (estar asociadas con) unhas coas outras de distintos xeitos:

2.2.2.2.1. Xeneralización

A herdanza é un dos conceptos fundamentais da programación orientada a obxectos, na que unha clase « gaña » todos os atributos e operacións da clase da que herda, e pode sobrescreber/modificar algúns deles, así como engadir máis atributos e operacións propias.

En UML, unha asociación de *Xeneralización* entre dúas clases ponnas nunha hierarquía que representa o concepto de herdanza dunha clase derivada respecto dunha clase base. En UML, as xeneralizacións son representadas por unha liña que conecta as dúas clases, cunha frecha ao lado da clase base.



Representación visual dunha xeneralización en UML

2.2.2.2.2. Asociacións

Unha asociación representa unha relación entre clases, e dá a semántica común e a estrutura de moitos tipos de « conexións » entre obxectos.

As asociacións son o mecanismo que lles permite aos obxectos comunicarse entre si. Describen a conexión entre diferentes clases (a conexión entre os obxectos reais é denominada conexión entre obxectos, ou *ligazón*).

As asociacións poden ter un papel que especifica o propósito da asociación e pode ser uni ou bidireccional (indica se os dous obxectos participantes na relación poden enviar mensaxes uns aos outros, ou se só un deles sabe do outro). Cada extremo da asociación pode ter tamén unha multiplicidade, que dictamina cantos obxectos neste lado da asociación poden relacionarse cun obxecto do outro lado).

En UML as asociacións represéntanse como liñas que conectan as clases participantes na relación, e tamén poden mostrar o papel e a multiplicidade de cada un dos participantes. A multiplicidade é mostrada como un rango [mín..máx] de valores non negativos, onde un asterísco (*) no máximo representa o infinito.



Representación visual dunha asociación en UML

2.2.2.2.3. Agregación

As agregacións son un tipo especial de asociación na que as dúas clases participantes non teñen o mesmo estatus, senón que teñen unha relación « todo-parte ». Unha agregación describe como a clase que adopta o papel de todo, está composta por(ten) outras clases, que adoptan o papel de partes. Nas agregacións, a clase que actúa como o todo sempre ten unha multiplicidade de un.

En UML as agregacións represéntanse mediante unha asociación cun rombo no lado do todo.

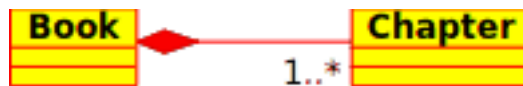


Representación visual dunha relación de agregación en UML

2.2.2.2.4. Composición

As composicións son asociación que representan agregacións *moi fortes*. Isto significa que as composicións tamén representan relacións todo-parte, pero cunha relación tan forte que as partes non poden existir por si mesmas. Só existen dentro do todo, e se o todo é destruído as partes tamén.

En UML as Composicións represéntanse cun rombo sólido no lado do todo.



2.2.2.3. Outros elementos dos diagramas de clase

Os diagramas de clase ademais de clases poden conter outro elementos.

2.2.2.3.1. Interfaces

As interfaces son clases abstractas que representan instancias que non poden ser criadas directamente. Poden conter operacións pero non atributos. As clases poden herdar de interfaces (mediante unha asociación de realización) e as instancias pode estar feitas destes diagramas.

2.2.2.3.2. Tipos de datos

Os tipos de datos son primitivas que tipicamente veñen incorporadas na linguaxe. Exemplos habituais son os inteiros e os booleanos. Non poden ter relacións con clases, pero as clases poden ter relacións con eles.

2.2.2.3.3. Enumeracións

As enumeracións son listas simples de valores. Un exemplo típico é unha enumeración dos días da semana. As opcións dunha enumeración son denominadas Literais da enumeración. Ao igual cos tipos de datos non poden ter relacións coas clases pero as clases poden ter relacións con elas.

2.2.2.3.4. Paquetes

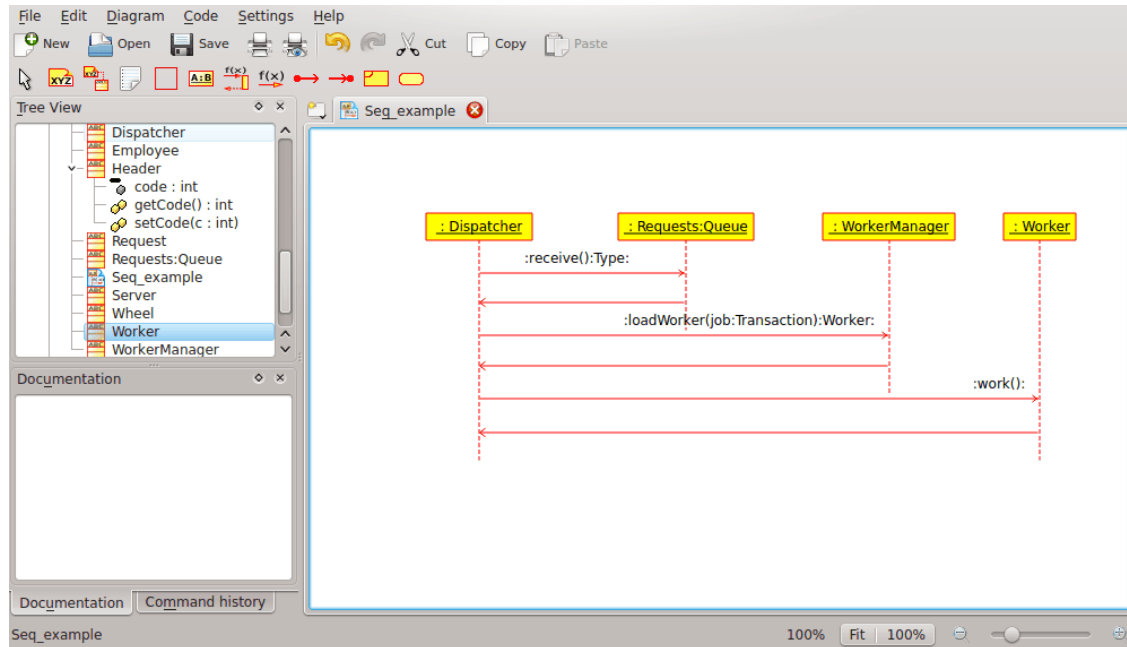
Os paquetes representan un espazo de nomes na linguaxe de programación. Nos diagramas son usados para representar partes dun sistema que conteñen máis dunha clase, mesmo centos delas.

2.2.3. Diagramas de secuencia

Os diagramas de secuencia mostran o intercambio de mensaxes (isto é chamadas a métodos) entre varios obxectos nunha situación concreta delimitada no tempo. Os obxectos son instancias das clases. Os diagramas de secuencia fan énfase na orden e tempos nos que se envían as mensaxes aos obxectos.

Nos diagramas de secuencia os obxectos son representados mediante liñas verticais de puntos co nome do obxecto no cume. O eixo do tempo tamén é vertical, aumentando cara abaixo, de tal

xeito que as mensaxes son enviadas dun obxecto para outro na forma de frechas co nome e os parámetros da operación.



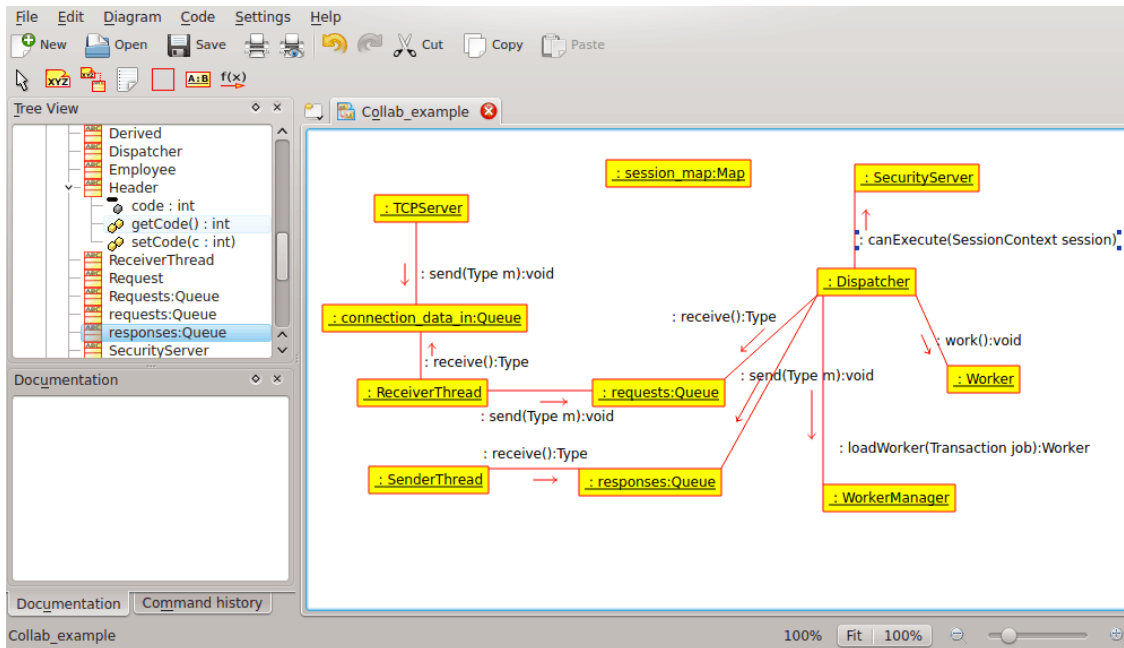
Umbrello UML Modeller mostrando un diagrama de secuencia

As mensaxes poden ser síncronas, o tipo normal de mensaxe no que o control é pasado á obxecto chamado até que o seu método remate a execución, ou asíncronas, onde o control é devolto directamente ao obxecto que fai a chamada. As mensaxes síncronas teñen unha caixa vertical ao lado do obxecto chamado para mostrar o fluxo de control do programa.

2.2.4. Diagramas de colaboración

Os diagramas de colaboración mostran as interaccións que acontecen entre os obxectos que participan nunha situación dada. Isto é similar á información que se mostra nun diagrama de secuencia pero neste faise fincapé en como acontecen as interaccións no tempo mentres que nos diagramas de colaboración poñen de relevo as relacións entre os obxectos e a súa topoloxía.

Nos diagramas de colaboración as mensaxes enviadas entre obxectos son representadas por frechas, co nome da mensaxe, os parámetros e a secuencia da mensaxe. Os diagramas de colaboración son adecuados para mostrar un fluxo específico do programa ou unha situación e son un dos mellores tipos de diagrama para demostrar rapidamente ou explicar un proceso na lóxica do programa.



Umbrello UML Modeller mostrando un diagrama de colaboración

2.2.5. Diagrama de estado

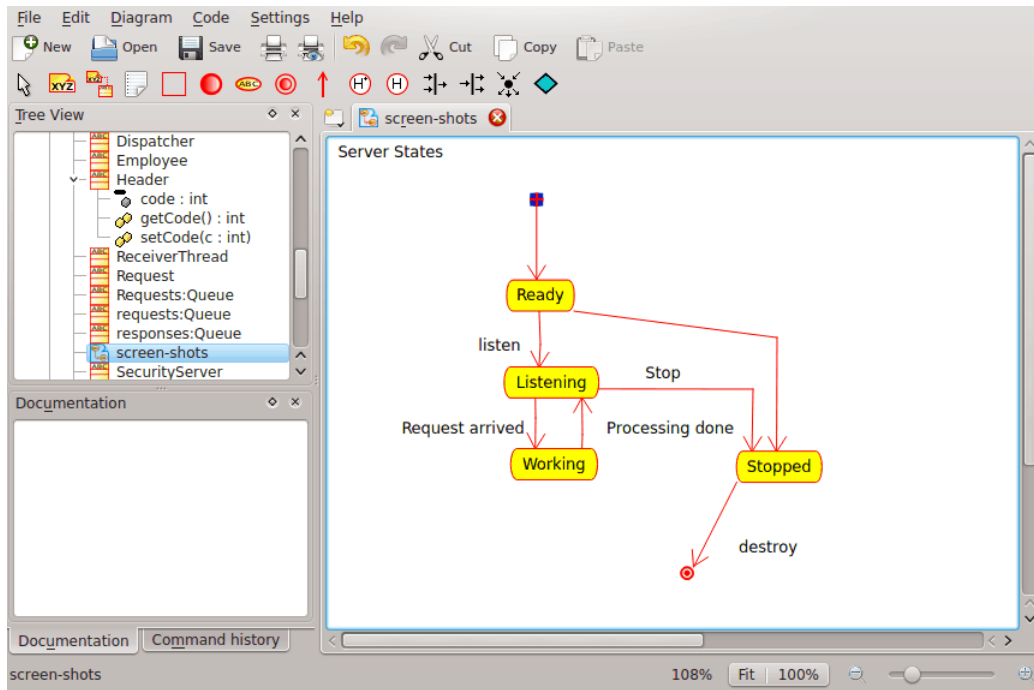
Os diagramas de estado mostran os diferentes estados dun obxecto durante a súa existencia e os estímulos que fan que o obxecto mude o seu estado.

Os diagramas de estado consideran os obxectos como *máquinas de estados* ou autómatas finitos que poden estar nun dun conxunto finito de estados e que poden mudar o seu estado mediante un conxunto finito de estímulos. Por exemplo un obxecto do tipo *NetServer* pode estar nun dos seguintes estados durante a súa vida:

- En agarda
- A escoitar
- A traballar
- Parado

e os eventos que poden facer que o obxecto mude de estado son

- O obxecto é criado
- O obxecto recibe a mensaxe “escoitar”
- Un cliente pide unha conexión sobre a rede
- Un cliente finaliza unha conexión
- A petición é executada e finaliza
- O obxecto recibe a mensaxe “parar”
- etc



Umbrello UML Modeller mostrando un diagrama de estado

2.2.5.1. Estado

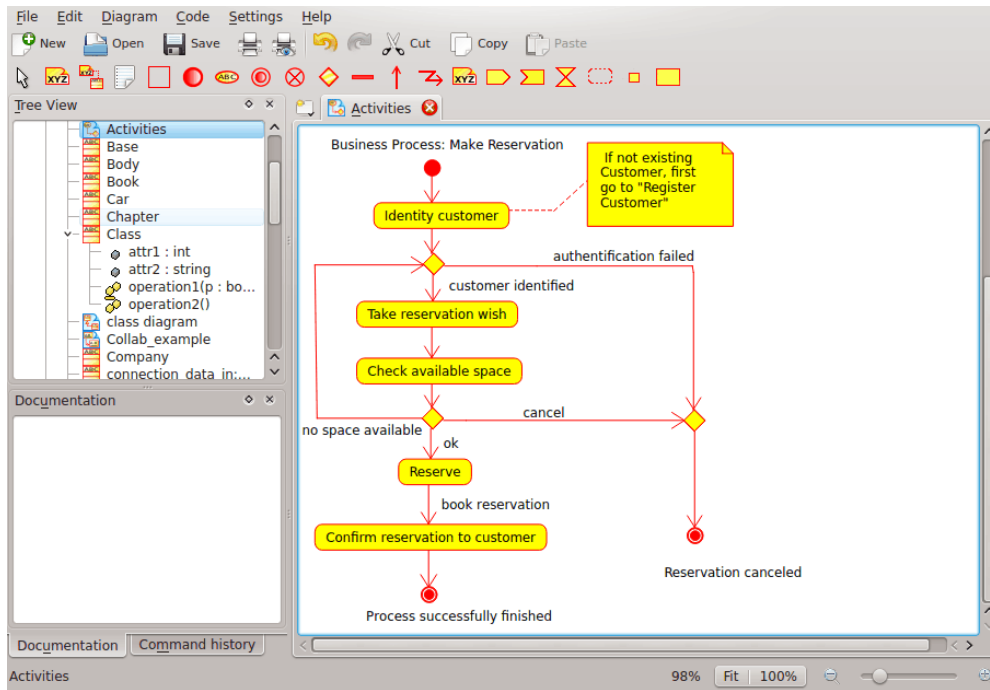
Os estados son os tixolos cos que se fan os diagramas de estado. Un estado pertence a exactamente unha clase e representa un resumo dos valores que poden tomar os atributos dunha clase. Un estado en UML representa o estado interno dun obxecto nunha clase particular

Lembre que non todas as alteración nun atributo dun obxecto debe ser representado por un estado senón só aquelas modificación que poden afectar de xeito significativo a como traballe o obxecto

Hai dous tipos especiais de estado: Inicio e Fin. Son especiais porque nengún evento pode facer que un obxecto volte ao seu estado de Inicio, do mesmo xeito que nengún evento pode facer que un obxecto volte do estado de Fin unha vez que é acadado.

2.2.6. Diagrama de actividade

Os diagramas de actividade describen a secuencia do traballo nun sistema coa axuda de Actividades. Os diagramas de actividade son unha forma especial de diagrama de estados, que só (ou maiormente) contén Actividades.



Umbrello UML Modeller mostrando un diagrama de actividade

Os diagramas de actividade son similares aos Diagramas de Fluxo dos procedimentos, coa diferenza de que todas as Actividades están claramente ligadas a Obxectos.

Os diagramas de actividade sempre están asociados ou a unha *Clase*, a unha *Operación* ou a un *Caso de uso*.

Os diagramas de actividade admiten actividades tanto secuenciais como paralelas. A execución en paralelo é representada mediante ícones Ramificar/Xuntar, e para as actividades con execución en paralelo, non é importante a orden na que son realizadas (poden ser executadas á vez ou unha tras da outra)

2.2.6.1. Actividade

Unha actividade é unha etapa simples dun proceso. Unha actividade é un estado do sistema activo e, polo menos, unha transición saínte. As actividades tamén poden ter máis dunha transición saínte se teñen diferentes condicións.

As actividades poden formar hierarquías, isto significa que unha actividade pode estar composta por varias actividades « de detalle », neste caso as transicións entrantes e saíntes deben coincidir coas transicións de entrada e saída do diagrama de detalle.

2.2.7. Elementos auxiliares

Hai uns poucos elementos UML que non teñen valor semántico real para o modelo, pero axudan a aclarar partes do diagrama. Estes elementos son

- Liñas de texto
- Notas de texto e enganches
- Caixas

As liñas de texto son útiles para engadir un texto informativo breve ao diagrama. Non están ligadas a ningún elemento e non teñen significado para o modelo en si.

As notas son útiles para engadir información máis detallada acerca dun obxecto ou situación específica. Teñen a grande vantaxe de que poden estar ligadas a elementos UML para mostrar que a nota « pertence »a un obxecto ou situación específica.

As caixas son rectángulos non ligados a nada que poden ser usadas para agrupar elementos e facer os diagramas máis lexíbeis. Non teñen significado lóxico no modelo.

2.2.8. Diagrama de componentes

Os diagramas de componentes mostrans os componentes do sóftware (sexantecnoloxías de componentes como KParts, componentes CORBA ou Java Beans ou só seccións do sistema claramente diferenciábeis) e os artefactos cos que están feitos, como ficheiros de código fonte, bibliotecas de programación ou táboas de bases de datos relacionais.

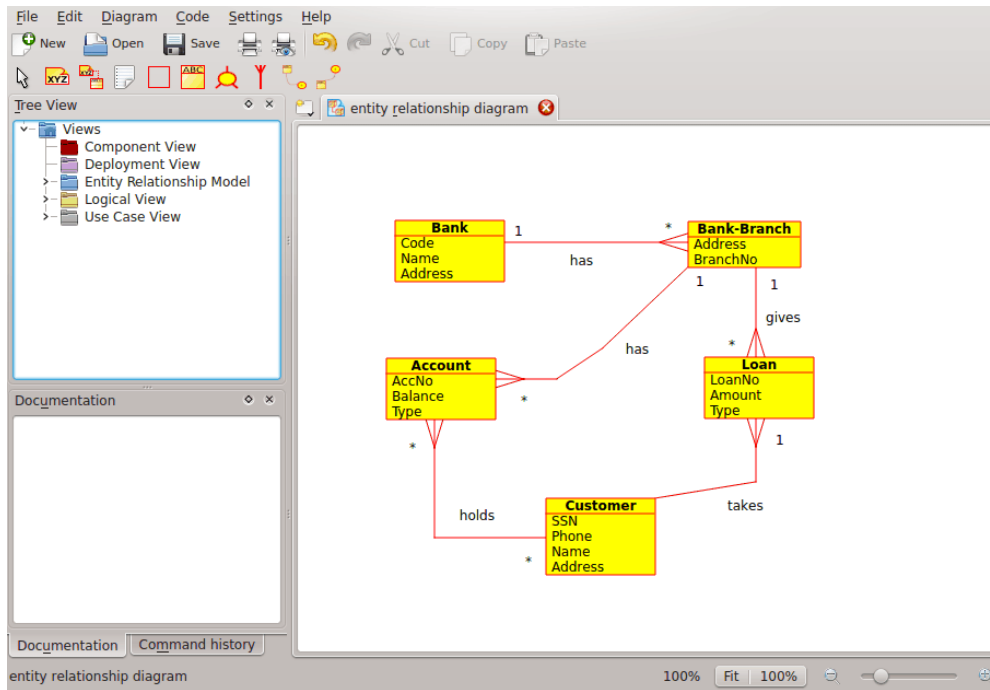
As componentes poden ter interfaces (isto é clases abstractas con operacións) que permiten asociacións entre componentes.

2.2.9. Diagramas de implementación

Os diagramas de implementación mostran as instancias en tempo de execución e as súas asociación. Inclúen Nós, que son recursos físicos, tipicamente un ordenador. Tamén mostran as interfaces e os obxectos (instancias de clases).

2.2.10. Diagramas de relacións entre entidades

Os diagramas de relacións entre entidades (Diagramas ER) mostran o deseño conceptual das bases de datos. Con eles tense unha visión das diferentes entidades (conceitos) do sistema de información e das relacións e restricións existentes entre elas. Emprégase unha extensión dos diagramas de relacións entre entidades, chamada “Diagramas ampliados de relacións entre entidades” (EER), para incorporar técnicas de deseño orientado a obxectos nos diagramas ER.



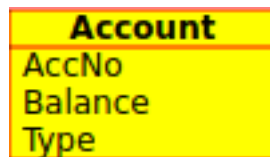
Umbrello mostrando un diagrama de relacións entre entidades

2.2.10.1. Entidade

Unha *entidade* é calquer concepto do mundo real que teña existencia independente. Pode ser un obxecto físico (por exemplo un ordenador, ou un bolígrafo) ou só ter existencia conceptual (p.ex: un curso da universidade). Cada entidade ten un conxunto de atributos que describen as propiedades da entidade.

Nota: Non hai unha notación estándar para os diagramas ER. Os diferentes textos que tratan desta materia empregan notacións distintas. Os conceptos e notacións dos diagramas EER empregados en Umbrello son tomados do seguinte libro: *Elmasri R. e Navathe S. (2004). Fundamentals of Database Systems 4th edn. Addison Wesley*

Nun diagrama ER, as entidades son representadas por rectángulos, co nome da entidade na parte superior, e tamén pode mostras os atributos da entidade en outro « compartimento » dentro do rectángulo.



Representación visual dunha entidade nun diagrama ER

2.2.10.1.1. Atributos da entidade

Nos diagramas ER, os atributos das entidades represéntanse polo seu nome nun compartimento diferente da entidade á que pertencen.

2.2.10.1.2. Restricións

As restricións nos diagramas ER especifican as restricións nos datos do esquema de información.

Umbrello admite catro tipos de restricións:

- *Chave primaria* O conxunto de atributos declarados como *chave primaria* son únicos da entidade. Só pode haber unha chave primaria nunha entidade e nengún dos atributos que a constitúan pode ser NULO.
- *Chave única*: O conxunto de atributos declarados como *únicos* son específicos da entidade. Poda haber varias restricións de unicidade nunha entidade. Os atributos que a constitúen poden ser NULO. As chaves únicas e as chaves primarias identifican univocamente unha fila dunha táboa (entidade)
- *Chave externa*: Unha chave externa é unha restrición referencial entre dúas táboas. A chave externa identifica unha columna ou conxunto de columnas nunha táboa (referente) que se refete a unha columna ou conxunto de columnas de outra táboa (referida). As columnas na táboa referida deben formar unha chave primaria ou única.
- *Restrición por comprobación*: Unha restrición por comprobación (tamén coñecida por restrición de comprobación da táboa) é unha condición que define os datos válidos cando se engade ou actualiza unha entrada nunha táboa dunha base de datos relacional. Estas restricións aplícanse a cada fila da táboa. A restrición debe ser un predicado. Pode referirse a unha só ou a varias columnas da táboa.
Exemplo: $\text{prezo} \geq 0$

2.2.10.2. Conceitos dos diagramas de relacións ampliadas entre entidades (EER)

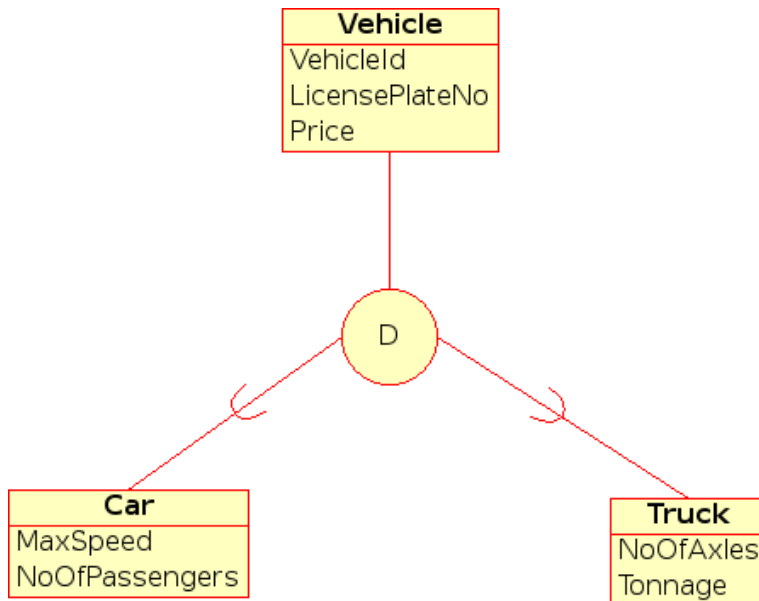
2.2.10.2.1. Especialización

A especialización é unha maneira de formar novas entidades empregando entidades xa definidas. As entidades novas, coñecidas como entidades derivadas, toman (herdan) atributos das preexistentes, ás que se coñece por entidades base. O propósito é reutilizar datos xa existentes con pouca ou nengunha modificación.

En Umbrello, pódese especificar que a especialización sexa Disxunta ou Sobreposicionante

2.2.10.2.1.1. Especialización disxunta

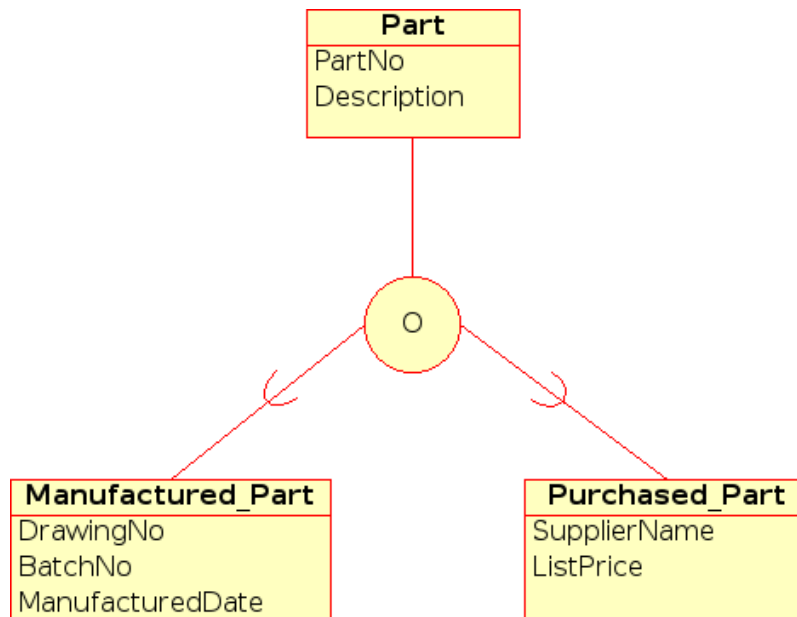
A especialización disxunta indica que as subclases da especialización deben ser disxuntas. Isto significa que unha entidade pode ser membro de como moito unha das entidades derivadas da especialización



Representación visual da especialización disxunta nun diagrama EER

2.2.10.2.1.2. Especialización sobreposicionante

Cando as entidades derivadas non teñen a restrición de ser disxuntas, dise que o seu conxunto de entidades está en especialización con sobreposición. Isto significa que a mesma entidade do mundo real pode ser membro de máis dunha entidade derivada da especialización.

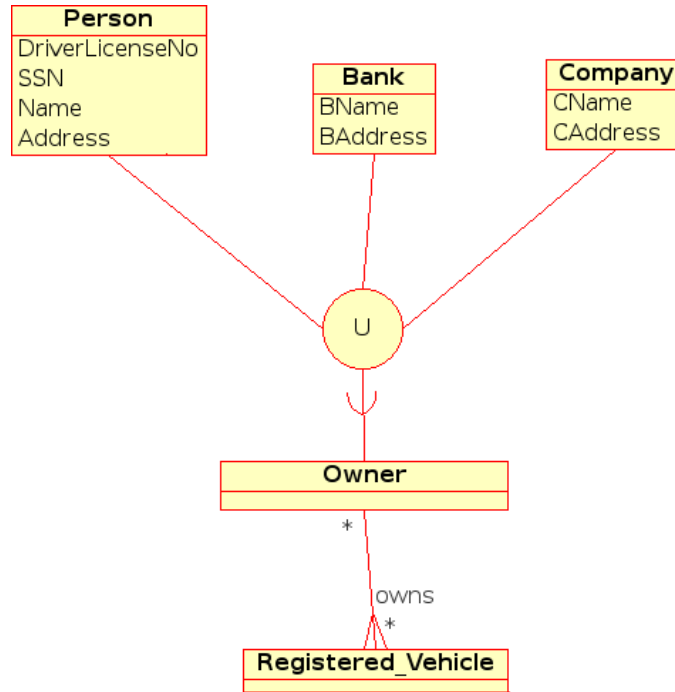


Representación visual da especialización sobreposicionante nun diagrama EER

2.2.10.2.1.3. Categoría

Dise que unha entidade derivada é unha *Categoría* cando representa unha colección de obxectos que é un subconxunto da unión dos distintos tipos de dato. A categoría modelízase cando surxe

a necesidade dunha única relación superclase/subclase con máis dunha superclase, onde a superclase representa diferentes tipos de entidades. (Como a herdanza múltipla na programación orientada a obxectos).



Representación visual dunha Categoría nun diagrama EER

Capítulo 3

Traballar con Umbrello UML Modeller

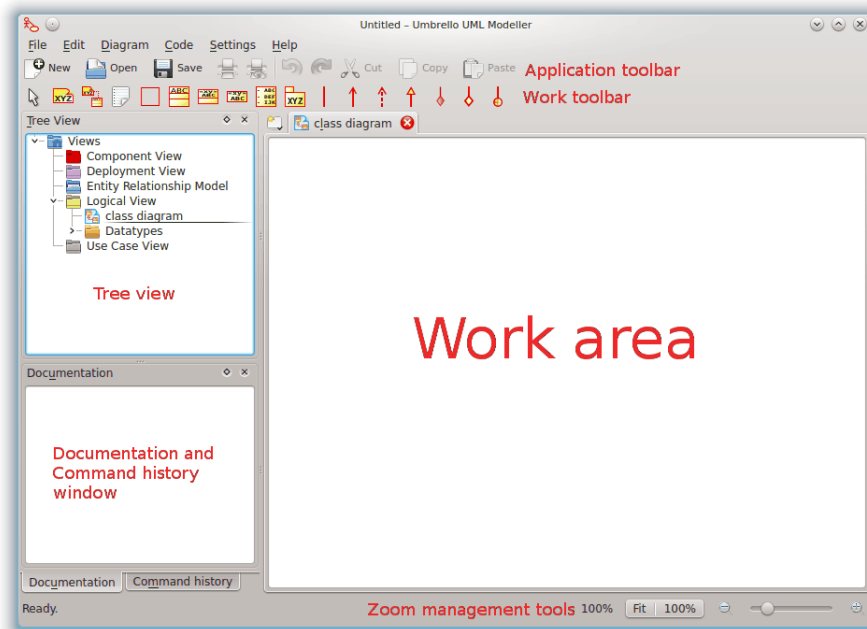
Este capítulo preséntalle a interface de usuario de Umbrello UML Modeller e aprenderalle todo o que precisa saber para comezar a modelar. Todas as acción de Umbrello UML Modeller están accesíbeis mediante o menú e as barras de ferramentas, pero Umbrello UML Modeller tamén fai uso dos menús contextuais do botón dereito do rato. Pode premer co botón dereito do rato en case calquer elemento da área de traballo de Umbrello UML Modeller ou vista en árbore para obter un menú coas funcións máis úteis que poden ser aplicadas ao elemento en concreto no que está a traballar. Algúns usuarios atopan isto algo lioso ao primeiro porque están afeitos a traballar co menú e coas barras de ferramentas, pero unha vez que se acostuma a premer no botón dereito acelerará moito o seu traballo.

3.1. A interface do usuario

A fiestra principal de Umbrello UML Modeller está dividida en tres áreas que o axudarán a ter unha vista global do sistema e a acceder aos diferentes diagramas rapidamente mentres traballa no seu modelo.

Estas áreas son denominadas:

- Vista en árbore
- Área de traballo
- Fiestra de documentación



A interface de usuario de Umbrello UML Modeller

3.1.1. A vista en árbore

A vista en árbore polo xeral está localizada no lado de riba á esquerda da xanela e mostra todos os diagramas, clases, actores e casos de uso que conforman o modelo. A vista en árbore permítelle obter unha vista global rápida dos elementos que compoñen o modelo. Esta vista tamén fornece un xeito rápido de cambiar entre os diagramas do modelo e de inserir elementos do modelo no diagrama actual.

Se está a traballar nun modelo con máis que só unhas poucas clases e diagramas, a vista en árbore pode axudalo a non perderse, organizando os elementos do modelo en cartafoles. Pode crear cartafoles escollendo a opción apropiada do menú contextual (prema co botón dereito do rato nun dos cartafoles da vista en árbore) e pode organizar os elementos movéndoo para o cartafol apropiado (arrastrar e soltar)

3.1.2. Fiestra de documentación

A fiestra da documentación é a pequena fiestra situada no fondo á esquerda en Umbrello UML Modeller, e dalle unha antevisión rápida da documentación do ítem actualmente escollido. A fiestra de documentación é bastante pequena porque foi pensada para permitirlle unha visión por riba da documentación do elemento e para ocupar o mínimo espazo posíbel. Se precisa consultar a documentación en máis detalle poderá abrir as propiedades do ítem.

3.1.3. Área de traballo

A área de traballo é a fiestra principal de Umbrello UML Modeller e é onde acontece a actividade real. Use a área de traballo para ver os diagramas do modelo. Esta área mostra o diagrama activo. Na actualidade só se pode ver un diagrama á vez nesta área.

3.2. Criar, cargar e gardar modelos

A primeira cousa que precisa para comezar a facer algo útil con Umbrello UML Modeller é crear un modelo no que traballar. Cando inicie Umbrello UML Modeller, este sempre carga o último modelo usado ou crea un modelo novo baleiro (dependendo das súas preferencias configuradas no diálogo de configuración). Isto permítelle comezar o traballo sen demoras.

3.2.1. Novo modelo

No momento no que precise crear un modelo novo poderá facelo mediante a opción **Novo** do menú **Ficheiro**, ou premendo no ícone **Novo** na barra de ferramentas da aplicación. Se está a traballar nun modelo que ten modificacións sen gardar Umbrello UML Modeller preguntarlle que debe guardar os cambiaos antes de comezar o novo.

3.2.2. Gardar o modelo

Pode gardar o modelo en calquer momento mediante a opción **Gardar** do menú **Ficheiro** ou premendo no botón **Gardar** na barra de ferramentas do programa. Se precisa gardar o modelo cun nome diferentes pode usar a opción **Gardar como** no menú **Ficheiro**.

Umbrello UML Modeller tamén lle ofrece a posibilidade de gardar automaticamente o traballo cada certo tempo. Pode indicar se quer esta opción así como o período de tempo na **Configuración** de Umbrello UML Modeller

3.2.3. Cargar modelo

Para cargar un modelo xa existente pode ou escoller a opción **Abrir** no menú **Ficheiro** ou premer no ícone **Abrir** na barra de ferramentas do programa. Os modelos usados máis recentemente tamén están dispoñíbeis no submenú **Abrir recente** no menú **Ficheiro** para facilitar o acceso aos modelos usados máis habitualmente.

Umbrello UML Modeller só pode traballar nun modelo á vez, polo que se lle pide ao programa que cargue un modelo e o modelo que teña aberto foi modificado tras a última vez que o gardou, Umbrello UML Modeller preguntarlle se debe gardar as modificacións para evitar perder o traballo. Pode iniciar dúas ou máis instancias de Umbrello UML Modeller, tamén pode copiar e apegar entre instancias.

3.3. Editar modelos

En Umbrello UML Modeller, hai esencialmente dúas maneiras de editar os elementos no modelo.

- Editar os elementos do modelo directamente mediante a Vista en árbore
- Editar os elementos do modelo mediante un Diagrama

Se usa o menú contextual dos diferentes itens na Vista en árbore poderá engadir, borrar e modificar case todos os elementos do modelo. Se preme co botón dereito nos cartafoles na Vista en árbore terá a opción de crear os diferentes tipos de diagramas e tamén, dependendo de se o cartafol é unha *Vista de casos de uso* ou unha *Vista lóxica*, Actores, Casos de uso, Clases, etc.

Cando teña engadidos elementos ao modelo tamén poderá editar un elemento acedendo ao diálogo de propiedades, que se mostrará se escolle a opción *Propiedades* no menú contextual mostrado cando preme do botón dereito nos elementos na Vista en árbore.

Tamén pode editar o modelo criando ou modificando os elementos mediante diagramas. Poderá encontrar máis detalles acerca de como facer isto nas seguintes seccións.

3.4. Engadir e eliminar diagramas

O modelo UML consiste nun conxunto de elementos UML e asociacións entre eles. Porén non pode ver o modelo directamente, debe usar os *Diagramas* para velo.

3.4.1. Criar diagramas

Para criar un diagrama novo no modelo simplemente escolla o tipo de diagrama que precisa en **Diagrama Menú** e bápíteo. O diagrama será criado e activarase, e poderá velo inmediatamente na Vista en árbore.

Lembre que Umbrello UML Modeller fai moito uso dos menús contextuais: tamén pode premer co botón dereito do rato nun cartafol na Vista en árbore e escoller o tipo de diagrama apropiado no submenú **Novo** do menú contextual. Lembre que só pode criar Diagramas de caso de uso nos cartafoles de vista de casos de uso, e os outros tipos de diagrama só poden ser criado nos cartafoles de vista lóxica.

3.4.2. Eliminar diagramas

Se ten que eliminar un diagrama do seu modelo, poderá facelo se o activa e escolle **Borrar** no menú **Diagrama**. Tamén pode facer isto se escolle **Borrar** no menú contextual do diagrama na Vista en árbore

Xa que eliminar un diagrama é algo serio que pode provocar a perda de traballo se é feito por accidente, Umbrello UML Modeller requiriralle que confirme a operación antes de efectuala. Cando se borra un diagrama, se o ficheiro é gardado non hai forma de reverter esta acción.

3.4.3. Mudar o nome dos diagramas

Se quer mudar o nome dun diagrama pode facelo facilmente escollendo a opción Mudar o nome no menú do botón dereito do rato na Vista en árbore.

Outro xeito de mudar o nome dun diagrama é facer isto mediante o seu diálogo de propiedades, que se obtén escollendo Propiedades no menú contextual ou facendo duplo-click nel na vista en árbore.

3.5. Editar diagramas

Cando se traballa nun diagrama, Umbrello UML Modeller tentará guialo aplicando algunhas regras de validez dos elemento no tipo de diagrama, así como as relacións que poden existir entre eles. Se é un experto en UML probablemente nen se decate, pero isto axudará aos inexpertos en UML a criar diagramas que sigan os estándares.

Cando teña criados os diagrama é hora de comezar a editalos. Aquí poderá detectarse da diferenza (para os inexpertos subtil) entre editar o diagrama e e editar o *modelo*. Como xa sabe, os diagramas son *vistas* do modelo. Por exemplo, se cria unha clase editando un diagrama de clase, en realidade está a editar ambos os dous, o diagrama e o modelo. Se troca a cor ou outra opción de visualización dunha clase no diagrama de clase, só estará a editar o diagrama, pero nada terá mudado no modelo.

3.5.1. Inserir elementos

Unha das primeiras cousas que fará cando edite un diagrama novo é inserir elementos nel (clases, actores, casos de uso, etc.). Basicamente hai dous xeitos de facer isto:

- Arrastrar elementos existentes no modelo desde a vista en árbore
- Criar elementos novos no modelo e engadilos ao seu diagrama ao mesmo tempo, usando unha das ferramentas de edición na barra de ferramentas de traballo

Para inserir elementos que xa existan no modelo, simplemente arrástreos desde a vista en árbore e déixeos onde os desexe pór no diagrama. Sempre poderá desprazar os elementos no diagrama usando a ferramenta de selección

A segunda maneira de engadir elementos ao diagrama é usando as ferramentas de edición da barra de traballo (lembre que isto tamén engadirá os elementos ao modelo).

A barra de ferramentas de traballo antes estaba localizada por omisión no extremo dereito da fiestra da aplicación, desde Umbrello UML Modeller 1.2 está na parte de riba da fiestra. Pode acoplala en outro lugar ou tela flutuante se o prefere. As ferramentas dispoñíbeis nesta barra de ferramentas (os botóns que ve nela) cambian dependendo do tipo de diagrama no que estexa a traballar. O botón da ferramenta actualmente escollida está activo na barra de ferramentas. Pode cambiar para a ferramenta de selección premendo a tecla **Esc**.

Cando teña escollida unha ferramenta de selección na barra de ferramentas de traballo (por exemplo, a ferramenta para inserir clases) o ponteiro do rato converterase nunha cruz, e poderá inserir os elementos no modelo só con premer no diagrama. Lembre que os elementos en UML deben ter un *Nome único*. Polo que se nun diagrama ten unha clase de nome « ClaseA » e logo usa a ferramenta de inserir clase para inserir unha clase noutro diagrama non lle poderá dar tamén a esta o nome « ClaseA ». De suporse que estes dous elementos son diferentes, terá que darlles nomes únicos. Se está a tentar engadir o *mesmo* elemento no diagrama, entón a ferramenta de Inserir clase non é a correcta. Deberá usar arrastrar e soltar para inserir a clase desde a vista en árbore.

3.5.2. Borrar elementos

Pode borrar calquer elemento escollendo a opción **Eliminar** no seu menú contextual.

Tamén aquí hai unha *grande* diferenza entre eliminar un obxecto dun diagrama, e eliminar un obxecto do modelo: se elimina un obxecto dun diagrama, só está a borrarlo dese diagrama en particular: o elemento aínda será parte do modelo e se hai outros diagramas a usar o mesmo elemento non irán sufrir ningún cambio. Aliás, se elimina o elemento da vista en árbore, en realidade está a borrar o elemento do *modelo*. Xa que o elemento deixa de existir no modelo, será automaticamente eliminado de todos os diagramas nos que apareza.

3.5.3. Editar elementos

Pode editar a maioría dos elementos UML do modelo e diagramas abrindo o diálogo de propiedades e escollendo as opcións apropiadas. Para editar as propiedades dun obxecto, escolla **Propiedades** no menú contextual (prema o botón dereito do rato). Cada elemento ten un diálogo que consiste en varias páxinas onde pode configurar as opcións correspondentes a ese elemento. Nalgúns elementos, como os actores, só pode configurar un par de opción, como o nome do obxecto e a documentación, mentres que noutros elementos, como as clases, pode editar os seus atributos e operacións, escoller o que quer que se vexa no diagrama (toda a sinatura da operación ou só o nome, etc) e mesmo as cores que quer usar para a liña e recheo da representación da clase no diagrama.

Na maioría dos elementos UML tamén poderá abrir o diálogo de propiedades se fai duplo-click nel se está a usar a ferramenta de selección (frecha). A excepción a isto son as Asociacións, neste

caso un duplo-click crea un punto de enganche. Para as asociación deberá usar o menú contextual do botón dereito do rato para ver o diálogo de propiedades.

Lembre que tamén pode escolles a opción das propiedades no menú contextual dos elementos na vista en árbore. Isto tamén lle permitirá editar as propiedades dos diagramas, como configurar se a grella debe ser mostrada ou non.

3.5.4. Editar clases

Aínda que a edición da propiedades dos obxectos xa foi tratada na anterior sección, as clases merecen unha sección especial porque son un pouco máis complexas e teñen máis opcións que a maioría dos elementos UML.

No diálogo de propiedades da clase pode configurar todo, desde a cor que usa ás operacións e atributos que ten.

3.5.4.1. Configuración xeral da clase

A páxina de configuración xeral do diálogo de propiedades explícase só. Aquí pode mudar o nome da clase, a súa visibilidade, documentación, etc. Esta páxina sempre está dispoñíbel.

3.5.4.2. Configuración dos atributos da clase

Na páxina de configuración dos atributos poderá engadir, editar ou eliminar atributos (variábeis) da clase. Pode subir ou baixar os atributos na lista cos botóns con frechas. Esta páxina sempre está dispoñíbel.

3.5.4.3. Configuración das operacións da clase

De xeito similar á páxina de configuración dos atributos, na de Operación poderá engadir, editar ou eliminar operacións para a clase. Cando engade ou edita unha operación, introduce os datos básicos no diálogo de *Propiedades da operación*. Se quer engadir parámetros á operación deberá premer o botón **Parámetro novo**, que mostrará o diálogo de *Propiedades do parámetro*. Esta páxina sempre está dispoñíbel.

3.5.4.4. Configuración do modelo da clase

Esta páxina permítelle engadir modelos de clase que son clases ou tipos de datos sen especificar. En Java 1.5 son denominados "Generics".

3.5.4.5. Páxina de asociacións da clase

A páxina de **Asociacións da clase** mostra todas as asociacións desta clase no diagrama actual. Se fai duplo-click nunha asociación mostrará as propiedades, e dependendo do tipo de asociación poderá modificar algúns parámetros como a multiplicidade e o nome do papel. Se a asociación non permite que se modifiquen estas opcións, o diálogo de Propiedades da asociación será de só-leitura e só poderá modificar a documentación asociada con esta asociación.

Esta páxina só está dispoñíbel se abre as Propiedades da clase desde un diagrama. Se escolle as propiedades da clase no menú contextual na vista en árbore non estará dispoñíbel.

3.5.4.6. Páxina de visualización da clase

Na páxina de **Opcións de visualización**, poderá indicar o que se debe mostrar no diagrama. Unha clase pode ser mostrada como un simple rectángulo co nome da clase nel (útil se ten moitas clases no seu diagrama, ou se polo de agora non está interesado nos detalles de cada clase) ou tan completo como mostrando paquetes, estereotipos e atributos e operación coa sinatura completa e a súa visibilidade

Dependendo da cantidade de información que queira ver poderá escoller as opcións correspondentes nesta páxina. As modificacións que faga aquí só son *opcións de visualización* para o diagrama. Isto significa que « acochar » as operacións dunha clase só fará que non se mostren no diagrama, pero as operacións aínda serán parte do seu modelo. Esta opción só está dispoñible se escolle as propiedades da clase desde un diagrama. Se simplemente as abre desde a vista en árbore esta páxina non estará porque tais opcións de visualización non teñen sentido neste caso

3.5.4.7. Páxina de cor da clase

Na páxina de **Cor do widget** poderá configurar as cores que quer para a liña e recheo do widget. Esta opción só ten sentido para as clases que están a ser mostradas nos diagramas, e non estará se abre o diálogo de propiedades da clase na vista en árbore.

3.5.5. Asociacións

As asociacións relacionan dous obxectos UML entre si. Normalmente as asociacións son definidas entre dúas clase, pero algúns tipos de asociación pode existir tamén entre casos de uso e actores.

Para criar unha asociación escolla a ferramenta axeitada na barra de ferramentas de traballo (asociación xenérica, xeralización, agregación, etc.) e prema no primeiro elemento que participe na asociación e logo no segundo elemento participante. Lembre que son dous clicks, un en cada un dos obxectos a participar na asociación, *non* arrastre desde un obxecto para o outro.

Se tenta usar unha asociación dun xeito que viole a especificación UML, Umbrello UML Modeller rexeitará criar a asociación e mostrará unha mensaxe de erro. Este será o caso se, por exemplo, existindo unha xeralización desde a clase A para a clase B, tenta criar outra xeralización da clase B para a clase A

Se preme co botón dereito nunha asociación mostrará un menú contextual coas acción que pode aplicarlle. Se precisa borrar unha asociación, escolla a opción **Eliminar** no seu menú contextual. Tamén pode escoller a opción **Propiedades** e, dependendo do tipo de asociación, editar os atributos tais como papeis e multiplicidade.

3.5.5.1. Pontos de enganche

As asociacións son debuxadas, por omisión, como liñas rectas que conectan os dous obxectos.

Pode engadir puntos de enganche para dobrar unha asociación se fai duplo-click sobre a liña de asociación. Isto inserirá un punto de enganche (mostrado como un punto azul cando escolle a liña da asociación) que pode desprazar para darlle forma á asociación

Se precisa eliminar un punto de enganche, faga duplo-click nel para eliminalo

Lembre que a única maneira de editar as propiedades dunha asociación é mediante o menú contextual. Se tenta facer duplo-click nela como cos outros obxectos UML, inserirá un punto de enganche.

3.5.6. Notas, texto e caixas

Notas, liñas de texto e caixas son elementos que poden estar presentes en calquer tipo de diagrama e non teñen valor semántico real, pero son moi útiles para engadir comentarios adicionais ou explicacións que poden simplificar a comprensión do diagrama.

Para engadir unha nota ou liña de texto, escolla a ferramenta correspondente na barra de ferramentas de traballo e prema na localización do diagrama onde queira pór o comentario. Pode editar o texto abrindo o elemento mediante o menú contextual ou no caso de notas tamén cun duplo-click sobre ela.

3.5.6.1. Enganches

Os enganches son usados para ligar un nó de texto e outro elemento UML. Por exemplo, normalmente usará un nó de texto para explicar ou facer algún comentario acerca dunha clase ou asociación particular, neste caso pode usar o enganche para aclarar que a nota « pertence » a ese elemento en concreto.

Para engadir un enganche entre unha nota e outro elemento UML, use a ferramenta de enganche na barra de ferramentas de traballo. Primeiro debe premer na nota e logo no elemento UML co que a quer ligar.

Capítulo 4

Importación de fontes e xeración de código fonte

Umbrello UML Modeller é unha ferramenta de modelado UML, e como tal o seu propósito principal é axudalo na *análise e deseño* de sistemas. Porén, para facer a transición entre o deseño e a *implementación*, Umbrello UML Modeller permítelle xerar código fonte en diferentes linguaxes de programación. Tamén, se quer comezar a usar UML nun proxecto en C++ xa iniciado, Umbrello UML Modeller pode axudalo a criar un modelo do sistema a partir do código fonte analisándoo e importando as clases que atope nel.

4.1. Xerazón de fontes

Umbrello UML Modeller pode xerar código fonte en varias linguaxes de programación baseándose no seu modelo UML para axudalo a comezar a implementación do seu proxecto. As fontes xeradas consisten nas declaracións da clase, cos seus métodos e atributos para que poda « preencher os ocos » fornecendo a funcionalidade das operacións das clases.

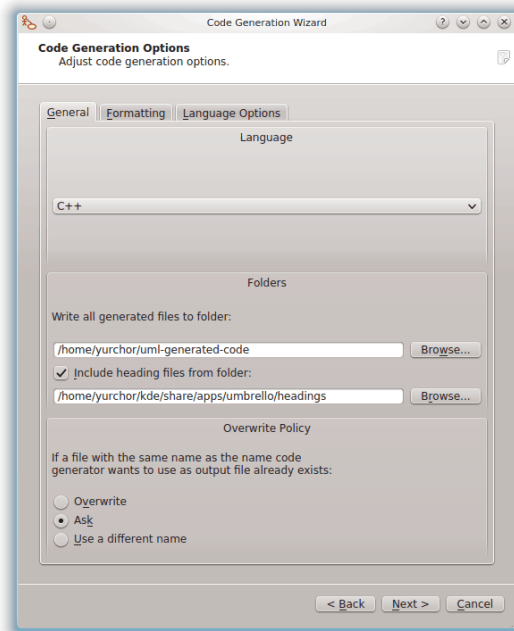
Umbrello UML Modeller 1.2 permite a xerazón de código fonte en ActionScript, Ada, C++, CORBA IDL, Java™, JavaScript, PHP, Perl, Python, SQL e XMLSchema.

4.1.1. Xerar o código fonte

Para xerar código fonte con Umbrello UML Modeller, primeiro debe criar ou cargar un modelo que conteña polo menos unha clase. Cando estexa listo para comezar a escribir algo de código fonte, escolla o **Asistente de xerazón de código fonte** no menú **Código** para iniciar un asistente que o irá guiar através do proceso de xerazón das fontes.

O primeiro paso é escolle as clases para as que queira xerar código fonte. Por omisión escolle-ranse todas as clases do modelo, e poderá quitar aquelas para as que non queira xerar fontes movéndooas para a lista do lado esquerdo.

O seguinte paso do asistente permítelle modificar os parámetros que o xerador de fontes usará para escribir as fontes. Están dispoñíbeis as seguintes opcións:



Opcións da xerazón de fontes en Umbrello UML Modeller

4.1.1.1. Opcións de xerazón

4.1.1.1.1. Verbosidade do código

A opción **Escribir comentarios de documentación aínda que estexa vacío** indícalle ao xerador de fontes que escriba comentarios do estilo `/** bla,bla,bla */` aínda que os bloques cos comentarios estexan baleiros. Se engadiu documentación ás clases, métodos ou atributos no modelo, o Xerador de código escribirá eses comentarios como documentación Doxygen con independencia do que indique aquí, pero se escolle esta opción Umbrello UML Modeller escribirá bloques de comentarios para todas as clases, métodos e atributos mesmo se non hai documentación no modelo, neste caso deberá documentar as clases máis tarde directamente nas fontes.

Escribir os comentarios das seccións aínda que estas estexan vacías fai que Umbrello UML Modeller escriba comentarios nas fontes para delimitar as diferentes seccións dunha clase. Por exemplo « métodos públicos » ou « atributos » antes das seccións correspondentes. Se escolle esta opción Umbrello UML Modeller escribirá comentarios en todas as seccións da clase mesmo se a sección está baleira. Por exemplo, escribirá un comentario dicindo « métodos protexidos » mesmo se a clase non ten métodos protexidos.

4.1.1.1.2. Cartafoles

Escribir todos os ficheiros xerados no cartafol. Aquí pode escoller o cartafol onde quer que Umbrello UML Modeller poña as fontes xeradas.

A opción **Incluír os ficheiros de definicións no cartafol** permítelle inserir un cabezal ao comezo de cada ficheiro xerado. Os ficheiros de definicións poden conter información acerca do copyright ou licenza e conteñen variábeis que son avaliadas no momento da xerazón. Pode botar unha ollada aos modelos de ficheiro de definicións distribuídos con Umbrello UML Modeller para ver como usar estas variábeis para substituír o seu nome ou a data actual cando se produza a xerazón.

4.1.1.1.3. Política de sobrescritura

Esta opción indícalle a Umbrello UML Modeller o que facer se o ficheiro que quer criar xa existe no cartafol de destino. Umbrello UML Modeller *non pode modificar os ficheiros de fontes xa existentes*, polo que terá que escoller entre sobrescribir o ficheiro existente, saltar a xerazón dese ficheiro ou permitirlle a Umbrello UML Modeller escoller un nome diferente para o ficheiro. Se escolle a opción de usar un nome diferente, Umbrello UML Modeller engadirá un sufixo ao nome do ficheiro.

4.1.1.1.4. Linguaxe

Por omisión, Umbrello UML Modeller xerará o código na linguaxe que escolleu como Linguaxe activa, pero co asistente para xerazón de código terá a opción de mudar isto para outra linguaxe.

4.1.1.2. Asistente de xerazón de código

O terceiro e último paso do asistente mostra o estado do proceso de xerazón de código fonte. Só precisa premer no botón de xerar para obter as clases.

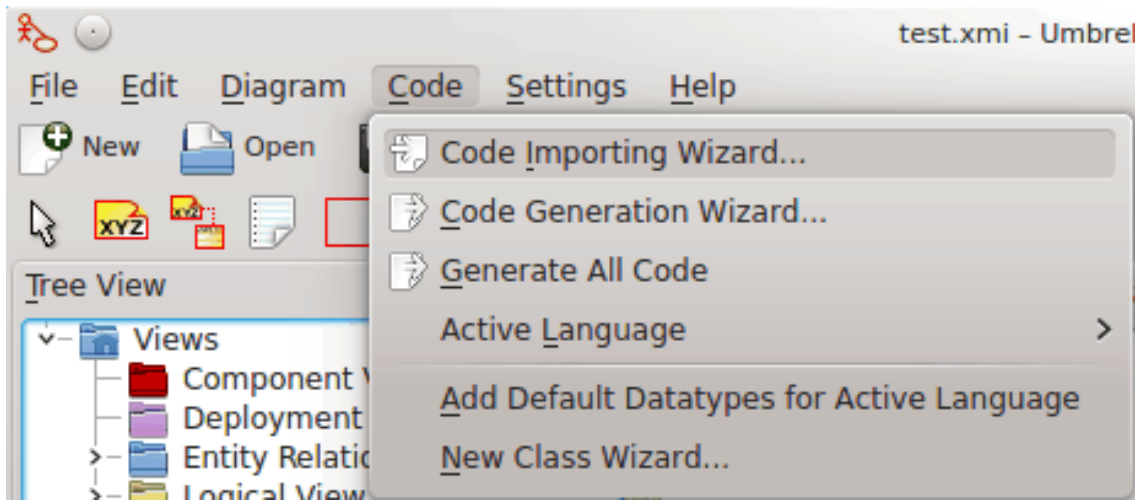
Lembre que as opcións que escolla no asistente de xerazón de código só son válidas para a xerazón actual. A próxima vez que execute o asistente deberá escoller de novo as opcións (o cartafol para as definicións, a política de sobrescritura, e demais). Pode escoller os valores por omisión usados por Umbrello UML Modeller na sección **Xerazón de código** coa configuración de Umbrello UML Modeller, dispoñíbel en **Configuración** → **Configurar Umbrello UML Modeller...**

Se configurou as opcións de xerazón de código correctamente e quer xerar algún código sen usar o asistente, pode escoller **Xerar todo o código** no menú Código. Isto xerará as fontes de todas as clases no modelo usando a configuración actual (incluíndo o Cartafol de saída e política de sobrescritura, polo que o debe usar con coidado).

4.2. Importación de código

Umbrello UML Modeller pode importar fontes de proxectos existentes para axudalo a elaborar o modelo dos sistemas. Umbrello UML Modeller 1.2 só soporta código C++, pero outras linguaxes estarán dispoñíbeis en versións futuras.

Para importar clases para o modelo, escolla a entrada **Importar clases...** no menú **Código**. No diálogo de ficheiro escolla os ficheiros que conteñan as declaracións das clases C++ e prema OK. As clases serán importadas e atoparaas como parte do modelo na vista en árbore. Lembre que Umbrello UML Modeller non criará ningún tipo de diagrama para mostrar as clases, só serán importadas no modelo polo que poderá usalas máis tarde en calquer diagrama que queira.



Menú para importar código fuente en Umbrello UML Modeller

Capítulo 5

Outras funcionalidades

5.1. Outras funcionalidades de Umbrello UML Modeller

Este capítulo describe por riba algunhas funcionalidades adicionais que ofrece Umbrello UML Modeller.

5.1.1. Copiar obxectos como imaxes PNG

Ademais das funcións normais de copiar, cortar e apegar que serven para copiar obxectos entre distintos diagramas, Umbrello UML Modeller pode copiar os obxectos como imaxes PNG para que os poda inserir en outros tipos de documento. Non precisa facer nada especial para usar esta funcionalidade, só escolla un obxecto no diagrama (Clase, Actor, etc.) e cópieo (**Ctrl-C**, ou mediante o menú), logo abra un documento de KWord (ou de calquer programa onde poda apegar imaxes) e escolla **Apegar**. Esta é unha funcionalidade perfecta para exportar partes do diagrama como imaxes simples.

5.1.2. Exportar para unha imaxe

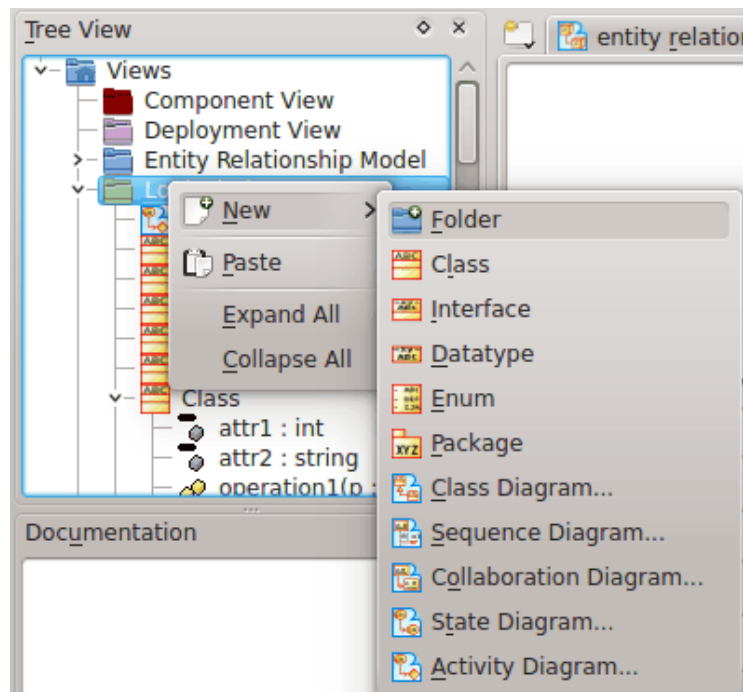
Tamén pode exportar un diagrama inteiro como unha imaxe. A única cousa que precisa facer é escoller o diagrama que quer exportar, e logo a opción **Exportar como imaxe...** no menú **Diagrama**.

5.1.3. Imprimir

Umbrello UML Modeller permítelle imprimir diagramas individualmente. Prema no botón **Imprimir** na barra de ferramentas da aplicación ou escolla a opción **Imprimir** no menú **Ficheiro** e mostrarase o diálogo estándar de KDE para imprimir.

5.1.4. Cartafoles lóxicos

Para organizar mellor o modelo, en especial en proxectos grandes, pode crear cartafoles lóxicos na Vista en Árbore. Simplesmente escolla a opción **Novo** → **Cartafol** no menú contextual dos cartafoles por omisión na vista en árbore para crialos. Os cartafoles poden estar uns dentro dos outros, e pode mover obxectos entre eles arrastrándoo dun para o outro.



Organizando un modelo con cartafolios lógicos en Umbrello UML Modeller

Capítulo 6

Autores e historial

Este proxecto foi iniciado por Paul Hensgen como un dos seus proxectos na universidade. O nome orixinal da aplicación era UML Modeller. Paul fixo todo o desenvolvemento até finais de 2001 cando o programa acadou a versión 1.0.

A versión 1.0 xa ofrecía moitas funcionalidades, pero tras a revisión do proxecto na Universidade onde Paul estudaba, outros desenvolvementes puideron unirse e comezar a facer importantes contribucións a UML Modeller, como trocar dun formato binario de ficheiro a un ficheiro XML, soportar máis tipos de Diagramas UML, Xeración de código e Importación de código, só por citar unhas poucas.

Paul tivo que retirarse da equipa de desenvolventes no verao de 2002, pero, como sóftware libre e de código aberto, o programa continúa a mellorar e evoluir xa que está a ser mantido por un grupo de desenvolventes de diferentes partes do mundo. En setembro de 2002 o proxecto mudou o nome de UML Modeller para Umbrello UML Modeller. Hai varias razóns para o troco do nome, sendo as máis importantes que só « uml » —como era comunmente coñecido— era xenérico de máis e causaba problemas nalgunhas distribucións. Outra razón importante é que os desenvolventes pensan que Umbrello é un nome moito mellor.

O desenvolvemento de Umbrello UML Modeller e as discusións acerca de aonde debe dirixirse o desenvolvemento das futuras versións, é aberto e realízase en internet. Se desexa contribuir ao proxecto, non dubide en contactar cos desenvolventes. Hai moitos xeitos nos que pode axudar a Umbrello UML Modeller:

- Informando de erros ou enviando suxestións
- Corrixindo erros ou engadindo funcionalidades
- Escribindo documentación ou traducíndoo a outros idiomas
- E por suposto... programando con nós!

Como pode ver, hai moitos de axudar. Todos son moi importantes e calquera é benvido para participar.

Os desenvolventes de Umbrello UML Modeller poden ser localizados en uml-devel@lists.sourceforge.net.

Capítulo 7

Copyright

Copyright 2001, Paul Hensgen

Copyright 2002, 2003 Os autores de Umbrello UML Modeller. Consulte <http://uml.sf.net/developers.php> para máis información

Esta documentación está licenciada sob os termos da [GNU Free Documentation License](#).

Este programa está licenciado sob os termos da [Licenza Pública Xeral de GNU](#).