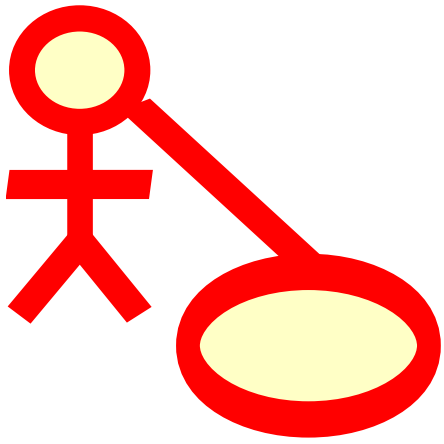
Manuale di Umbrello UML Modeller



Indice

1 Introduzione **8**

2 Fondamenti di UML **9**

2.1 Informazioni su UML . . . 9
2.2 Elementi UML . . . 10
2.2.1 Diagramma di caso d'uso . . . 10
2.2.1.1 Caso d'uso . . . 10
2.2.1.2 Attore . . . 11
2.2.1.3 Descrizione di caso d'uso . . . 11
2.2.2 Diagramma di classe . . . 11
2.2.2.1 Classe . . . 12
2.2.2.1.1 Attributi . . . 12
2.2.2.1.2 Operazioni . . . 12
2.2.2.1.3 Modelli . . . 12
2.2.2.2 Associazioni di classi . . . 12
2.2.2.2.1 Generalizzazione . . . 13
2.2.2.2.2 Associazioni . . . 13
2.2.2.2.3 Aggregazione . . . 13
2.2.2.2.4 Composizione . . . 14
2.2.2.3 Altri elementi dei diagrammi di classe . . . 14
2.2.2.3.1 Interfacce . . . 14
2.2.2.3.2 Tipi di dati . . . 14
2.2.2.3.3 Enumerazioni . . . 14
2.2.2.3.4 Pacchetti . . . 14
2.2.3 Diagrammi di sequenza . . . 15
2.2.4 Diagrammi di collaborazione . . . 15
2.2.5 Diagramma di stato . . . 16
2.2.5.1 Stato . . . 17
2.2.6 Diagrammi di attività . . . 17
2.2.6.1 Attività . . . 18
2.2.7 Elementi ausiliari . . . 18
2.2.8 Diagrammi dei componenti . . . 19
2.2.9 Diagrammi di dispiegamento . . . 19

2.2.10 Diagrammi di relazioni tra entità . . . 19
2.2.10.1 Entità . . . 20
2.2.10.1.1 Attributi delle entità . . . 20
2.2.10.1.2 Vincoli . . . 20
2.2.11 Concetti dei diagrammi di relazioni delle entità estesi (REE) . . . 20
2.2.11.1 Specializzazione . . . 20
2.2.11.1.1 Specializzazione disgiunta . . . 21
2.2.11.1.2 Specializzazione sovrapposta . . . 21
2.2.11.1.3 Categoria . . . 22

3 Lavorare con Umbrello UML Modeller **23**
3.1 Interfaccia utente . . . 23
3.1.1 Vista ad albero . . . 24
3.1.2 Finestra di documentazione e cronologia dei comandi . . . 24
3.1.3 Area di lavoro . . . 24
3.2 Creare, caricare e salvare i modelli . . . 25
3.2.1 Nuovo modello . . . 25
3.2.2 Salva modello . . . 25
3.2.3 Carica modello . . . 25
3.3 Modificare i modelli . . . 25
3.4 Aggiungere e rimuovere i diagrammi . . . 26
3.4.1 Creare i diagrammi . . . 26
3.4.2 Rimuovere i diagrammi . . . 26
3.4.3 Rinominare i diagrammi . . . 26
3.5 Modificare i diagrammi . . . 26
3.5.1 Inserire gli elementi . . . 27
3.5.2 Eliminare gli elementi . . . 27
3.5.3 Modificare gli elementi . . . 27
3.5.4 Modificare le classi . . . 28
3.5.4.1 Impostazioni generali delle classi . . . 28
3.5.4.2 Impostazione degli attributi delle classi . . . 28
3.5.4.3 Impostazioni delle operazioni delle classi . . . 28
3.5.4.4 Impostazioni dei modelli delle classi . . . 28
3.5.4.5 Pagina delle associazioni delle classi . . . 28
3.5.4.6 Pagina di visualizzazione delle classi . . . 29
3.5.4.7 Pagina degli stili delle classi . . . 29
3.5.5 Associazioni . . . 29
3.5.5.1 Punti di ancoraggio . . . 29
3.5.6 Note, testo e riquadri . . . 30
3.5.6.1 Ancore . . . 30

4 Importazione e generazione di codice **31**
4.1 Generazione di codice . . . 31
4.1.1 Generare il codice . . . 31
4.1.1.1 Opzioni di generazione . . . 32
4.1.1.1.1 Verbosità dei commenti . . . 32
4.1.1.1.2 Cartelle . . . 32
4.1.1.1.3 Politica di sovrascrittura . . . 33
4.1.1.1.4 Linguaggio . . . 33
4.1.1.2 Generazione con l'assistente di generazione . . . 33
4.2 Importazione di codice . . . 33

5 Altre funzionalità **35**
5.1 Altre funzionalità di Umbrello UML Modeller . . . 35
5.1.1 Copiare oggetti come immagini PNG . . . 35
5.1.2 Esportare come immagine . . . 35
5.1.3 Stampa . . . 35
5.1.4 Cartelle logiche . . . 35

6 Impostazioni **37**
6.1 Impostazioni generali . . . 37
6.1.1 Varie . . . 37
6.1.2 Salvataggio automatico . . . 38
6.1.3 Avvio . . . 38
6.1.4 Notifiche . . . 38
6.2 Impostazioni dei caratteri . . . 38
6.3 Impostazioni per l'interfaccia utente . . . 39
6.3.1 Generale . . . 39
6.3.2 Associazioni . . . 39
6.3.3 Colore . . . 39
6.4 Impostazioni della classe . . . 40
6.4.1 Mostra . . . 40
6.4.2 Ambito di partenza . . . 40
6.5 Impostazioni di importazione del codice . . . 41
6.5.1 Percorsi di ricerca dei file di inclusione . . . 41
6.5.2 Importazione C++ . . . 41
6.6 Impostazioni per la generazione del codice . . . 42
6.6.1 Scheda generale delle impostazioni per la generazione del codice . . . 42
6.6.1.1 Linguaggio . . . 42
6.6.1.2 Cartelle . . . 42
6.6.1.3 Linee guida di sovrascrittura . . . 42
6.6.2 Scheda formattazione delle impostazioni per la generazione del codice . . . 43
6.6.2.1 Verbosità dei commenti . . . 43

6.6.2.2 Linee 43
6.6.3 Opzioni di linguaggio 44
6.6.3.1 Generazione di codice C++ 44
6.6.3.1.1 Documentazione 44
6.6.3.1.2 Generale 44
6.6.3.1.3 Metodo di generazione del corpo 45
6.7 Impostazioni per la visualizzazione del codice 46
6.8 Impostazioni della disposizione automatica 47

7 Autori e storia **48**

8 Copyright **49**

Sommario

Umbrello UML Modeller aiuta il processo di sviluppo del software usando il Linguaggio di Modellazione Unificato (UML), uno standard industriale, per permetterti di creare diagrammi per progettare e documentare i tuoi sistemi.

Capitolo 1

Introduzione

Umbrello UML Modeller è uno strumento per diagrammi UML che può aiutarti nel processo di sviluppo del software. Specialmente durante le fasi di analisi e progettazione di questo processo, Umbrello UML Modeller ti aiuterà a ottenere un prodotto di alta qualità. UML può anche essere usato per documentare i tuoi progetti software per aiutare te e i tuoi colleghi sviluppatori.

Avere un buon modello del tuo software è il modo migliore per comunicare con altri sviluppatori che lavorano al progetto e con i tuoi clienti. Un buon modello è estremamente importante per i progetti di dimensioni medie e grandi, ma è anche molto utile per quelli piccoli. Anche se stai lavorando su un piccolo progetto da una sola persona, un buon modello ti gioverà perché ti darà una panoramica che ti aiuterà a codificare le cose subito nel modo giusto.

UML è il linguaggio di diagrammi usato per descrivere questi modelli. Puoi rappresentare le tue idee in UML usando diversi tipi di diagrammi. Umbrello UML Modeller 2.11 supporta i seguenti tipi:

- Diagrammi di classi
- Diagrammi di sequenza
- Diagrammi di collaborazione
- Diagrammi di caso d'uso
- Diagrammi di stato
- Diagrammi di attività
- Diagrammi di componenti
- Diagrammi di dispiegamento
- Diagramma di relazione tra entità

Si possono trovare maggiori informazioni su UML al sito Web di OMG, http://www.omg.org che ha creato lo standard UML.

Speriamo che Umbrello UML Modeller ti piaccia e che ti aiuti a creare software di alta qualità. Umbrello UML Modeller è software libero ed è disponibile gratis, l'unica cosa che ti chiediamo è di comunicare qualsiasi errore, problema o suggerimento agli sviluppatori di Umbrello UML Modeller a umbrello-devel@kde.org o https://bugs.kde.org.

Capitolo 2

Fondamenti di UML

2.1 Informazioni su UML

Questo capitolo ti darà una rapida panoramica dei fondamenti di UML. Tieni presente che questo non è un corso completo di UML, ma piuttosto una breve introduzione a UML, che può essere letta come un corso di UML. Se vuoi saperne di più sul Linguaggio di Modellazione Unificato, o in generale sull'analisi e la progettazione del software, riferisciti a uno dei molti libri disponibili sull'argomento. Ci sono anche molti corsi su Internet che puoi usare come punto di partenza.

Il Linguaggio di Modellazione Unificato (UML) è un linguaggio o notazione di diagrammi per specificare, visualizzare e documentare modelli di sistemi di software a oggetti. UML non è un metodo di sviluppo, cioè non ti dice cosa fare prima e dopo o come progettare il tuo sistema, ma ti aiuta a visualizzare il tuo progetto e a comunicare con gli altri. UML è controllato dal Gruppo di Gestione Oggetti (OMG) ed è lo standard industriale per descrivere graficamente il software.

UML è progettato per la progettazione di software a oggetti, e ha un uso limitato per altri paradigmi di programmazione.

UML si compone di molti elementi di modelli che rappresentano le diverse parti di un sistema software. Gli elementi UML sono usati per creare diagrammi, che rappresentano una certa parte, o punto di vista del sistema. Umbrello UML Modeller supporta i seguenti tipi di diagrammi:

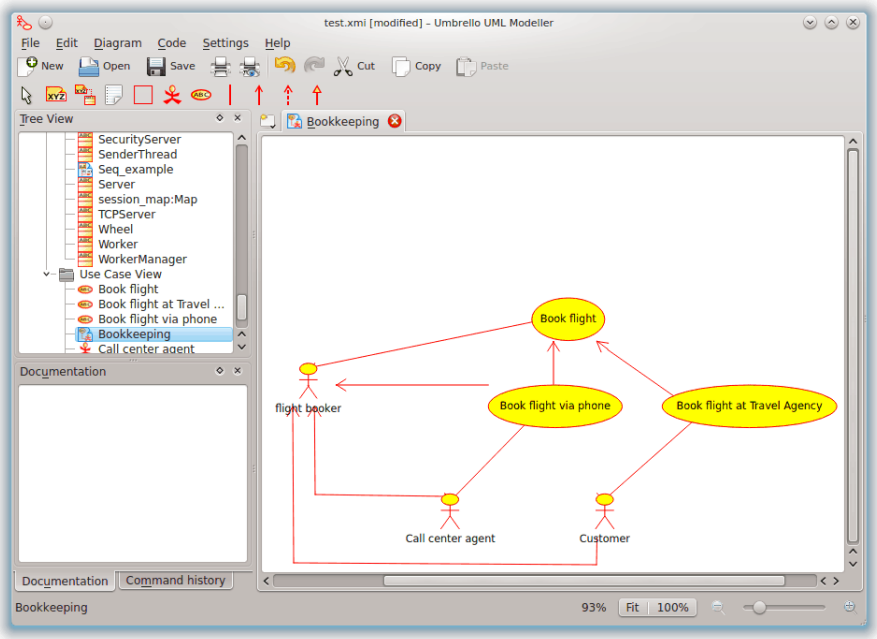
- *I diagrammi di caso d'uso* mostrano gli attori (persone o altri utenti del sistema), i casi d'uso (gli scenari di quando usano il sistema), e le loro relazioni
- *I diagrammi di classe* mostrano le classi e le relazioni tra loro
- *I diagrammi di sequenza* mostrano gli oggetti e una sequenza di chiamate a metodi che essi fanno ad altri oggetti.
- *I diagrammi di collaborazione* mostrano gli oggetti e le loro relazioni, ponendo l'attenzione sugli oggetti che partecipano nello scambio di messaggi
- *I diagrammi di stato* mostrano gli stati, i cambi di stato e gli eventi in un oggetto o in una parte del sistema
- *I diagrammi di attività* mostrano le attività e i cambiamenti da un'attività all'altra con gli eventi che accadono in qualche parte del sistema
- *I diagrammi dei componenti* mostrano i componenti della programmazione di alto livello (come KPart o Java Beans).
- *I diagrammi di dispiegamento* mostrano le istanze dei componenti e le loro relazioni.
- *I diagrammi di relazioni tra entità* mostrano i dati con le loro relazioni e vincoli.

2.2 Elementi UML

2.2.1 Diagramma di caso d'uso

I diagrammi di caso d'uso descrivono le relazioni e le dipendenze tra un gruppo di *Casi d'uso* e gli attori partecipanti al processo.

È importante notare che i diagrammi di caso d'uso non sono adatti a rappresentare la progettazione, e non possono descrivere le parti interne di un sistema. I diagrammi di caso d'uso servono a facilitare la comunicazione con gli utenti futuri del sistema e con il cliente, e sono particolarmente utili a determinare le funzionalità necessarie che il sistema deve avere. I diagrammi di caso d'uso dicono *cosa* deve fare il sistema, ma non specificano *come* si deve fare (e non possono).



Umbrello UML Modeller che mostra un diagramma di caso d'uso

2.2.1.1 Caso d'uso

Un *Caso d'uso* descrive, dal punto di vista degli attori, un gruppo di attività in un sistema che produce un risultato concreto e tangibile.

I casi d'uso sono descrizioni delle interazioni tipiche tra gli utenti di un sistema e il sistema stesso. Rappresentano l'interfaccia esterna del sistema, e specificano un modulo di requisiti di cosa il sistema deve fare (ricorda, solo cosa, non come).

Lavorando con i casi d'uso, è importante ricordare alcune semplici regole:

- Ogni caso d'uso è relativo ad almeno un attore
- Ogni caso d'uso ha un iniziatore (cioè un attore)
- Ogni caso d'uso porta a un risultato rilevante (un risultato con 'valore di business')

I casi d'uso possono anche avere relazioni con altri casi d'uso. I tre tipi più comuni di relazioni tra casi d'uso sono:

- «*inclusione*» che specifica che un caso d'uso avviene *all'interno* di un altro caso d'uso
- «*estensione*» che specifica che in certe situazioni, o a un certo punto (chiamato punto di estensione) un caso d'uso sarà esteso da un altro.
- La *generalizzazione* specifica che un caso d'uso eredita le caratteristiche del 'sovra'caso d'uso, e può sostituirne alcune o aggiungerne nuove in un modo simile all'ereditarietà tra classi.

2.2.1.2 Attore

Un attore è un'entità esterna (fuori dal sistema) che interagisce con il sistema partecipando a (e spesso iniziando) un caso d'uso. Gli attori possono essere persone reali (per esempio utenti del sistema), altri sistemi o eventi esterni.

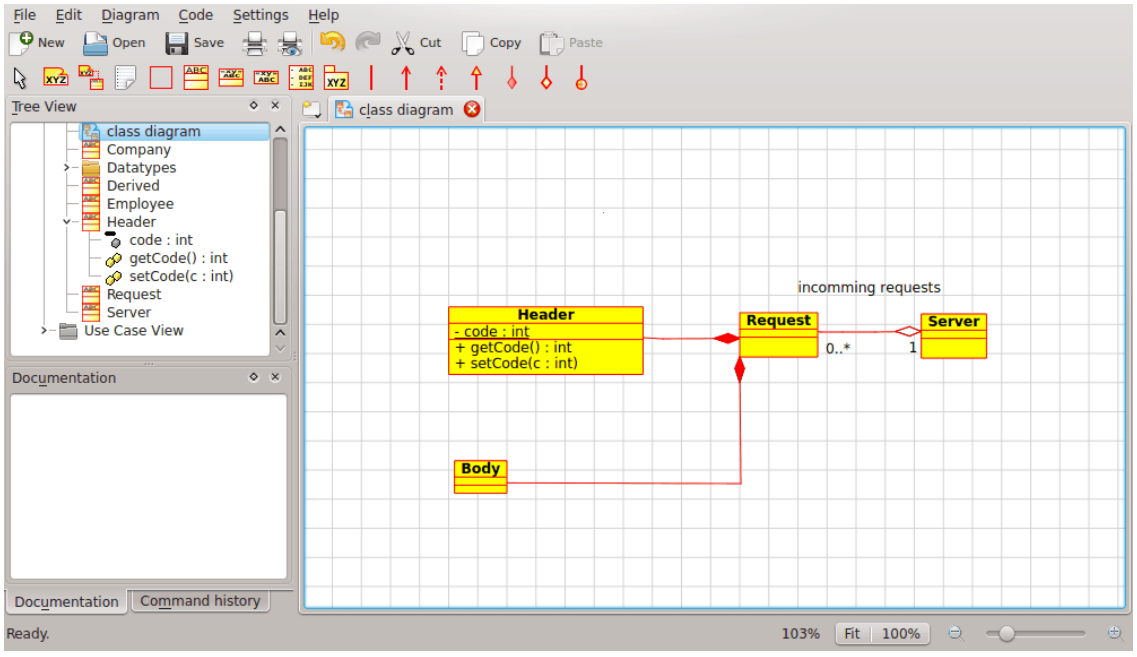
Gli attori non rappresentano le persone *fisiche* o i sistemi, ma il loro *ruolo*. Ciò significa che quando una persona interagisce con il sistema in modi diversi (assumendo ruoli diversi) sarà rappresentata da diversi attori. Per esempio una persona che fornisce supporto per telefono e immette ordini dal cliente nel sistema sarebbe rappresentato da un attore 'Squadra di supporto' e un attore 'Rappresentante di vendita'

2.2.1.3 Descrizione di caso d'uso

Le descrizioni di caso d'uso sono racconti testuali del caso d'uso. Prendono normalmente la forma di una nota o di un documento che è in qualche modo collegato al caso d'uso, e ne spiega i processi o le attività.

2.2.2 Diagramma di classe

I diagrammi di classe mostrano le diverse classi che costituiscono un sistema e come si relazionano una all'altra. I diagrammi di classe sono detti essere diagrammi 'statici' perché mostrano le classi, insieme ai loro metodi e attributi oltre alle relazioni statiche tra loro: quali classi 'sanno' di quali classi o quali classi 'fanno parte' di un'altra classe, ma non mostrano le chiamate ai metodi tra di loro.

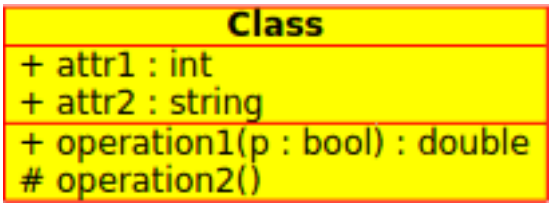


Umbrello UML Modeller che mostra un diagramma di classe

2.2.2.1 Classe

Una classe definisce gli attributi e i metodi di un insieme di oggetti. Tutti gli oggetti di questa classe (istanze di questa classe) condividono lo stesso comportamento, e hanno lo stesso insieme di attributi (ogni oggetto ha il suo insieme). A volte il termine 'Tipo' è usato invece di classe, ma è importante ricordare che questi due non sono la stessa cosa, e Tipo è un termine più generale.

In UML le classi sono rappresentate da rettangoli con il nome della classe, e possono anche mostrare gli attributi e le operazioni della classe in due altri 'scompartimenti' all'interno del rettangolo.



Rappresentazione visiva di una classe in UML

2.2.2.1.1 Attributi

In UML gli attributi sono mostrati con almeno il loro nome, e possono mostrare anche il loro tipo, il valore iniziale e altre proprietà. Gli attributi possono anche essere mostrati con la loro visibilità:

- + sta per attributi *pubblici*
- # sta per attributi *protetti*
- - sta per attributi *privati*

2.2.2.1.2 Operazioni

Anche le operazioni (metodi) sono mostrate con almeno il loro nome, e possono mostrare anche i loro parametri e i tipi restituiti. Le operazioni possono, come gli attributi, mostrare la loro visibilità:

- + sta per operazioni *pubbliche*
- # sta per operazioni *protette*
- - sta per operazioni *private*

2.2.2.1.3 Modelli

Le classi possono avere modelli, un valore che è usato per una classe o untipo non specificato. Il tipo di modello è specificato quando viene iniziata una classe (cioè cioè viene creato un oggetto). I modelli esistono nel C++ moderno e saranno introdotti in Java 1.5 dove saranno chiamati Generics.

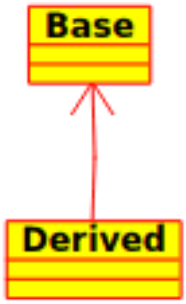
2.2.2.2 Associazioni di classi

Le classi possono relazionarsi (essere associate) una con l'altra in diversi modi:

2.2.2.2.1 Generalizzazione

L'ereditarietà è uno dei concetti fondamentali della programmazione a oggetti, in cui una classe 'acquisisce' tutti gli attributi e le operazioni della classe da cui eredita, e può sostituire o modificare alcuni di loro, oltre ad aggiungere altri attributi e operazioni proprie.

In UML, un'associazione di *Generalizzazione* tra due classi le mette in una gerarchia rappresentante il concetto di ereditarietà di una classe derivata da una classe di base. In UML le generalizzazioni sono rappresentate da una linea che connette le due classi, con una freccia sul lato della classe di base.



Rappresentazione visiva di una generalizzazione in UML

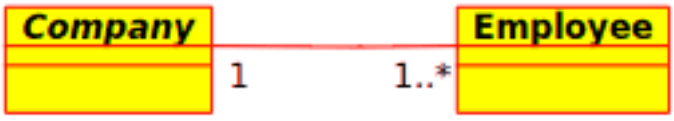
2.2.2.2.2 Associazioni

Un'associazione rappresenta una relazione tra classi, e dà la semantica e la struttura comuni a molti tipi di 'connessioni' tra oggetti.

Le associazioni sono i meccanismi che permettono agli oggetti di comunicare tra loro. Descrivono la connessione tra diverse classi (la connessione tra gli oggetti reali è chiamata una connessione tra oggetti, o *collegamento*).

Le associazioni possono avere un ruolo che specifica lo scopo dell'associazione e può essere uni- o bidirezionale (indica se i due oggetti che partecipano alla relazione possono mandare messaggi all'altro, o se solo uno di loro sa dell'altro). Ogni parte dell'associazione ha un valore di molteplicità, che detta quanti oggetti su questo lato dell'associazione possono relazionarsi a un oggetto sull'altro lato.

In UML le associazioni sono rappresentate come linee che connettono le classi che partecipano alla relazione, e possono anche mostrare il ruolo e la molteplicità di ciascuno dei partecipanti. La molteplicità è mostrata come un intervallo [minimo..massimo] di valori non negativi, con un asterisco (*) sul lato massimo per rappresentare l'infinito.



Rappresentazione visiva di un'associazione in UML

2.2.2.2.3 Aggregazione

Le aggregazioni sono un tipo speciale di associazioni nel quale le due classi partecipanti non hanno un rango uguale, ma hanno una relazione 'intero-parte'. Un'aggregazione descrive come la classe che ha il ruolo dell'intero è composta di (ha) altre classi, che hanno il ruolo di parti. Per le aggregazioni, la classe che fa da intero ha sempre molteplicità di uno.

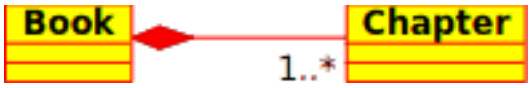
In UML le aggregazioni sono rappresentate da un'associazione che mostra un rombo sul lato dell'intero.

Rappresentazione visiva di una relazione di aggregazione in UML

2.2.2.2.4 Composizione

Le composizioni sono associazioni che rappresentano aggregazioni *molto forti*. Ciò vuol dire che anche le composizioni formano relazioni intero-parte, ma la relazione è così forte che la parte non può esistere di per sé. Esistono solo all'interno dell'intero, e se l'intero è distrutto anche le parti muoiono.

In UML le composizioni sono rappresentate da un rombo solido sul lato dell'intero.



2.2.2.3 Altri elementi dei diagrammi di classe

I diagrammi di classe possono contenere diversi altri elementi oltre alle classi.

2.2.2.3.1 Interfacce

Le interfacce sono classi astratte, che vuol dire che non se ne possono creare istanze direttamente. Possono contenere operazioni ma non attributi. Le classi possono ereditare dalle interfacce (attraverso una associazione di realizzazione) e possono essere create delle istanze di queste classi.

2.2.2.3.2 Tipi di dati

I tipi di dati sono delle primitive tipicamente incorporate in un linguaggio di programmazione. Gli esempi comuni includono gli interi e i booleani. Non possono avere relazioni con le classi, ma le classi possono avere relazioni con loro.

2.2.2.3.3 Enumerazioni

Le enumerazioni sono semplici elenchi di valori. Un esempio tipico è un'enumerazione dei giorni della settimana. Le opzioni di un'enumerazione sono chiamate letterali dell'enumerazione. Come i tipi di dati non possono avere relazioni con le classi, ma le classi possono avere relazioni con loro.

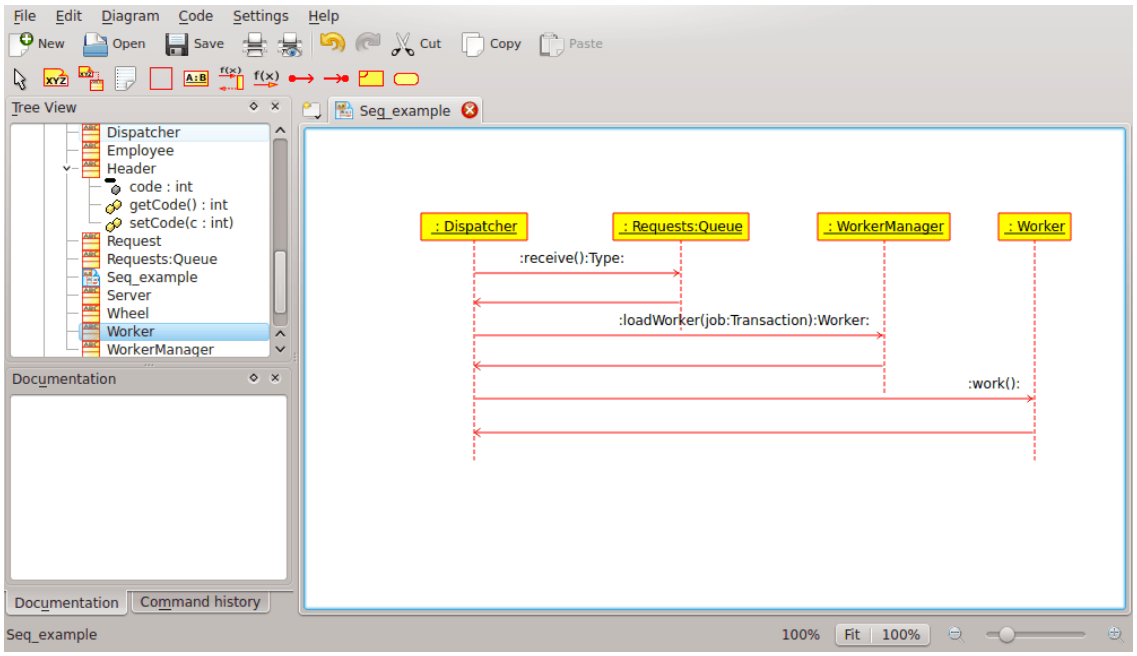
2.2.2.3.4 Pacchetti

I pacchetti rappresentano un namespace in un linguaggio di programmazione. In un diagramma sono usati per rappresentare parti di un sistema che contiene più di una classe, forse centinaia di classi.

2.2.3 Diagrammi di sequenza

I diagrammi di sequenza mostrano lo scambio di messaggi (cioè la chiamata ai metodi) tra diversi oggetti in una situazione temporale precisa. Gli oggetti sono istanze di classi. I diagrammi di sequenza mettono particolare enfasi sull'ordine e l'ora a cui i messaggi sono inviati agli oggetti.

Nei diagrammi di sequenza gli oggetti sono rappresentati con linee verticali tratteggiate, con il nome dell'oggetto in cima. Anche l'asse temporale è verticale, e aumenta andando in giù, in modo che i messaggi sono inviati da un oggetto all'altro nella forma di frecce con l'operazione e il nome del parametro.



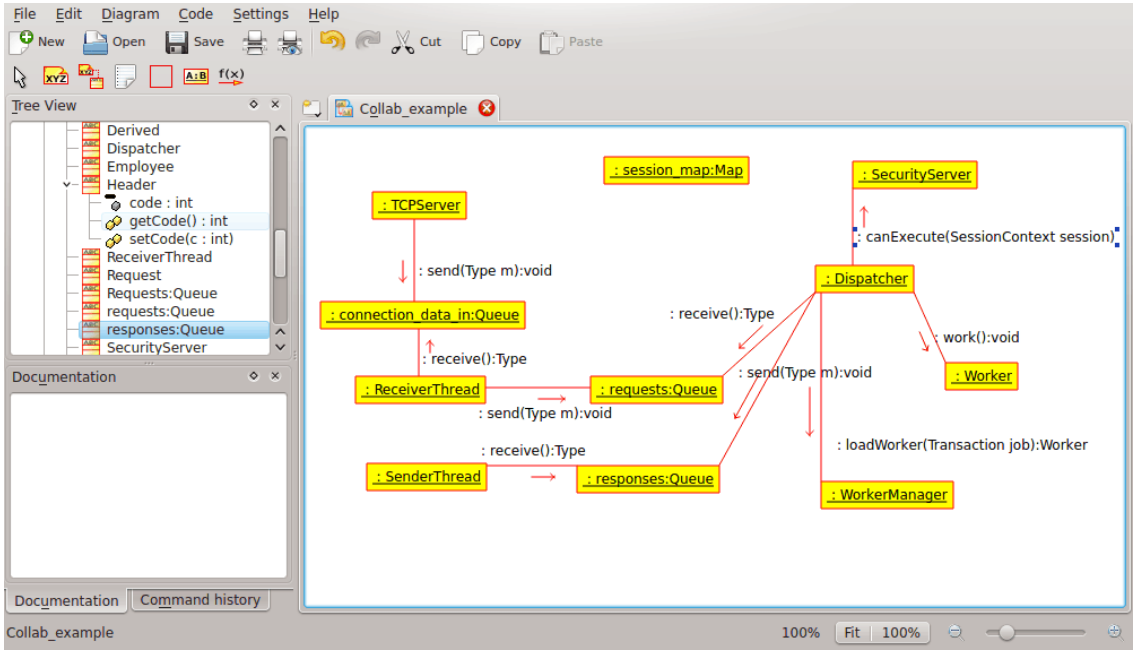
Umbrello UML Modeller che mostra un diagramma di sequenza

I messaggi possono o essere sincroni, il tipo normale di chiamate ai messaggi dove il controllo è passato all'oggetto chiamato fino a quando quel metodo ha finito l'esecuzione, o asincrono, dove il controllo è ripassato direttamente all'oggetto chiamante. I messaggi sincroni hanno un riquadro verticale a lato dell'oggetto chiamato per mostrare il flusso di controllo del programma.

2.2.4 Diagrammi di collaborazione

I diagrammi di collaborazione mostrano le interazioni che avvengono tra gli oggetti che partecipano a una situazione specifica. È più o meno la stessa informazione mostrata nei diagrammi di sequenza, ma lì l'enfasi è messa su come le interazioni avvengono nel tempo, mentre i diagrammi di collaborazione mettono in primo piano le relazioni tra gli oggetti e la loro topologia.

Nei diagrammi di collaborazione i messaggi inviati da un oggetto a un altro sono rappresentati da frecce, che mostrano il nome, i parametri e la sequenza del messaggio. I diagrammi di collaborazione sono specialmente adatti a mostrare un particolare flusso o situazione di programma e sono uno dei migliori tipi di diagramma per dimostrare o spiegare rapidamente un processo nella logica del programma.



Umbrello UML Modeller che mostra un diagramma di collaborazione

2.2.5 Diagramma di stato

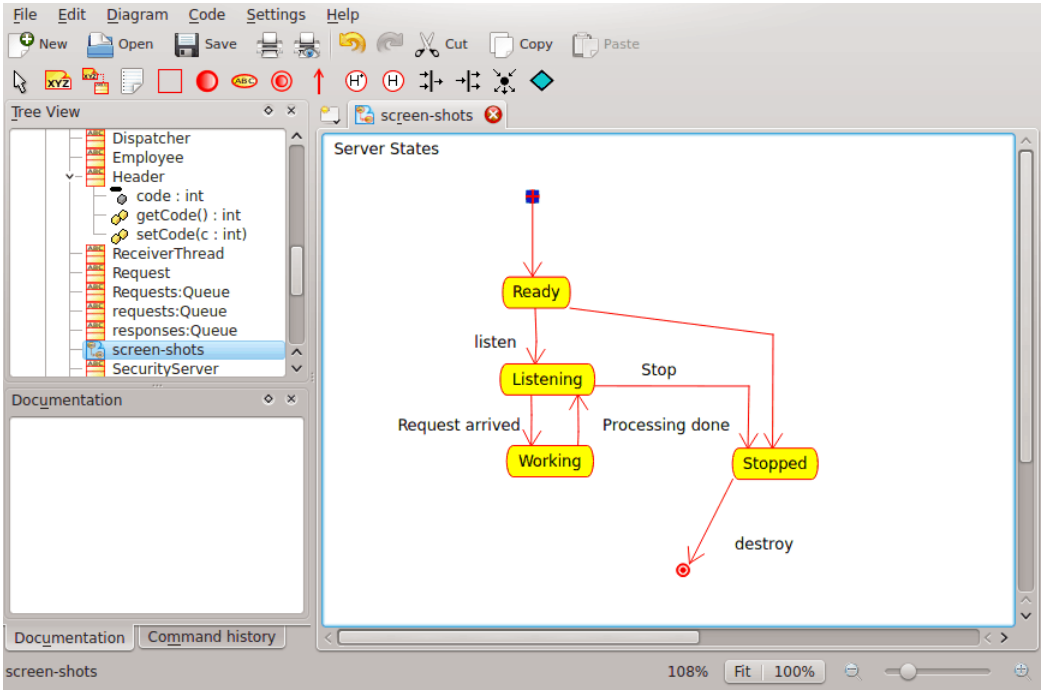
I diagrammi di stato mostrano i diversi stati di un oggetto durante la sua vita e gli stimoli che gli fatto cambiare stato.

I diagrammi di stato vedono gli oggetti come *macchine di stato* o automi finiti che possono essere in uno stato di un insieme di stati finiti e che possono cambiare stato con uno stimolo di un insieme finito di stimoli. Per esempio un oggetto di tipo *ServerDiRete* può essere in uno dei seguenti stati durante la sua vita:

- Pronto
- In ascolto
- Attivo
- Fermato

e gli eventi che possono far cambiare stato all'oggetto sono

- L'oggetto è creato
- L'oggetto riceve il messaggio di ascolto
- Un client richiede una connessione sulla rete
- Un client termina una richiesta
- La richiesta è eseguita e conclusa
- L'oggetto riceve un messaggio di fermata
- ecc.



Umbrello UML Modeller che mostra un diagramma di stato

2.2.5.1 Stato

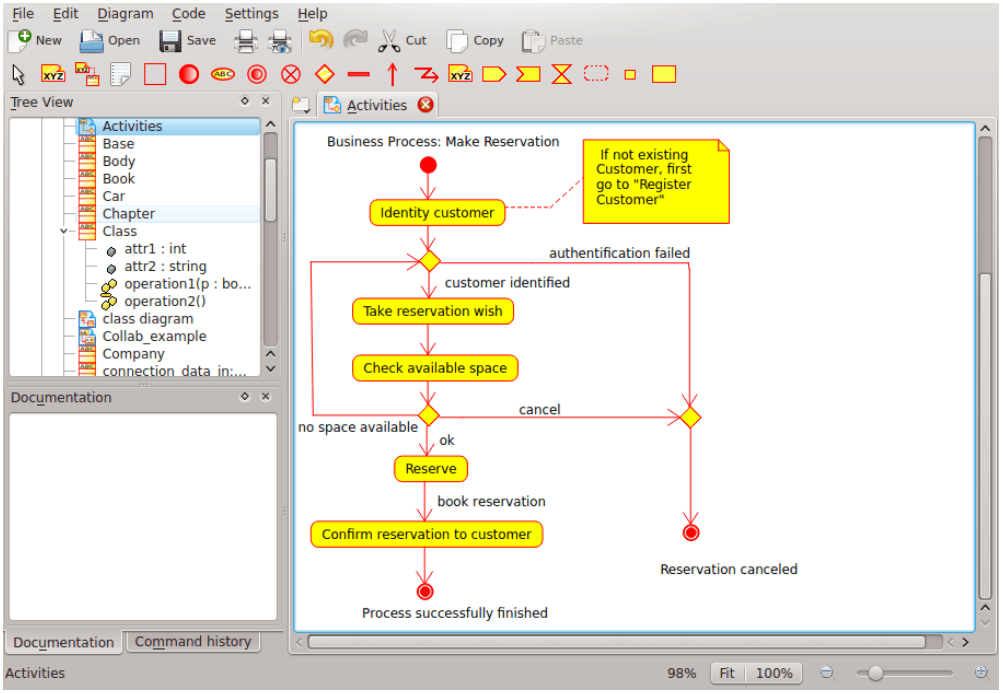
Gli stati sono i mattoni del diagramma di stato. Uno stato appartiene a esattamente una classe e rappresenta un riassunto dei valori che gli attributi di una classe possono assumere. Uno stato UML descrive lo stato interno di un oggetto di una classe particolare

Nota che non tutti i cambiamenti negli attributi di un oggetto dovrebbe essere rappresentato da uno stato, ma solo quei cambiamenti che influenzano significativamente il funzionamento dell'oggetto

Ci sono due tipi speciali di stati: Inizio e Fine. Sono speciali perché non ci sono eventi che possono far tornare un oggetto al suo stato di Inizio, allo stesso modo in cui non è possibile riportare un oggetto dal suo stato di Fine una volta che lo raggiunge.

2.2.6 Diagrammi di attività

I diagrammi di attività descrivono la sequenza di attività in un sistema con l'aiuto delle attività. I diagrammi di attività sono una forma speciale dei diagrammi di stato, che contengono solo (o per lo più) attività.



Umbrello UML Modeller che mostra un diagramma di attività

I diagrammi di attività sono simili ai diagrammi di flusso procedurali, con la differenza che tutte le attività sono chiaramente connesse a degli oggetti.

I diagrammi di attività sono sempre associati a una *classe*, un'*operazione* o un *caso d'uso*.

I diagrammi di attività supportano attività sequenziali e parallele. L'esecuzione parallela è rappresentata con le icone Separa/Attendi; per le attività in esecuzione in parallelo, non è importante l'ordine in cui sono eseguite (possono essere eseguite allo stesso tempo o una dopo l'altra)

2.2.6.1 Attività

Un'attività è un singolo passo in un processo. Un'attività è uno stato del sistema con attività interna e, almeno, una transizione all'esterno. Le attività possono anche avere più di una transizione all'esterno se hanno più condizioni.

Le attività possono formare gerarchie, vale a dire che un'attività può essere composta di più attività di 'dettaglio', nel qual caso le transizioni all'interno e all'esterno dovrebbero corrispondere alle transizioni all'interno e all'esterno del diagramma dei dettagli.

2.2.7 Elementi ausiliari

Ci sono alcuni elementi in UML che non hanno una vera e propria semantica per il modello, ma aiutano a chiarificare parti del diagramma. Questi elementi sono

- Righe di testo
- Note di testo e ancore
- Riquadri

Le righe di testo sono utili per aggiungere brevi informazioni testuali a un diagramma. È testo libero e non ha significato per il modello stesso.

Le note sono utili per aggiungere informazioni più dettagliate su un oggetto o una situazione specifica. Hanno il grande vantaggio che le note possono essere ancorate a elementi UML per mostrare che la nota 'appartiene' a un oggetto o una situazione specifici.

I riquadri sono rettangoli liberi che possono essere usati per raggruppare insieme degli elementi per rendere i diagrammi più leggibili. Non hanno nessun significato logico nel modello.

2.2.8 Diagrammi dei componenti

I diagrammi dei componenti mostrano i componenti software (tecnologie componenti come KPart, componenti CORBA, Java Beans o solo sezioni del sistema che sono chiaramente distinguibili) e i manufatti di cui sono fatti come file di codice sorgente, librerie di programmazione o tabelle di banche dati relazionali.

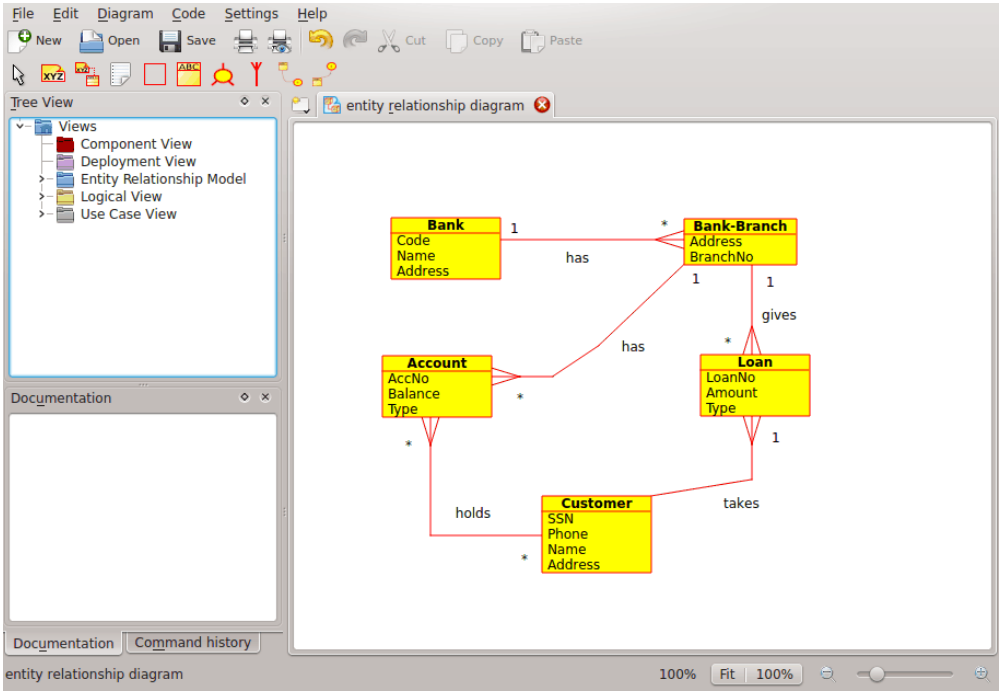
I componenti possono avere interfacce (cioè classi astratte con operazioni) che permettono associazioni tra componenti.

2.2.9 Diagrammi di dispiegamento

I diagrammi di dispiegamento mostrano le istanze dei componenti durante l'esecuzione e le loro associazioni. Includono i nodi, che sono risorse fisiche, tipicamente un computer singolo. Mostrano anche interfacce e oggetti (istanze di classi).

2.2.10 Diagrammi di relazioni tra entità

I diagrammi di relazioni tra entità (diagrammi RE) mostrano la progettazione concettuale delle applicazioni di banche dati. Mostrano le varie entità (concetti) nel sistema informativo e le relazioni e i vincoli esistenti tra di loro. Un'estensione dei diagrammi di relazioni tra entità, chiamata «diagrammi di relazioni tra entità estesi» (REE) o «diagrammi di relazioni tra entità migliorati», è usata per incorporare le tecniche di progettazione orientata agli oggetti nei diagrammi RE.



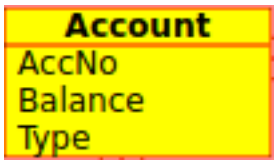
Umbrello UML Modeller che mostra un diagramma di relazioni tra entità

2.2.10.1 Entità

Un'*entità* è un qualsiasi concetto del mondo reale con un'esistenza indipendente. Può essere un oggetto con forma fisica (per esempio, un computer o un robot) o può essere un oggetto concettuale (come un corso universitario). Ogni entità ha un insieme di attributi che ne descrivono le proprietà.

Nota: non esiste nessuna notazione standard per tracciare i diagrammi RE. Diversi testi su questo tema usano notazioni diverse. I concetti e le notazioni dei diagrammi REE usati in Umbrello sono presi da questo libro: *Elmasri R. and Navathe S. (2004). Fundamentals of Database Systems 4th edn. Addison Wesley*.

In un diagramma RE, le entità sono rappresentate da rettangoli con il nome dell'entità sopra, e possono anche mostrare gli attributi dell'entità in un altro 'scompartimento' all'interno del rettangolo.



Rappresentazione visiva di un'entità in un diagramma RE

2.2.10.1.1 Attributi delle entità

Nei diagrammi RE, gli attributi delle entità sono mostrati con il loro nome in uno scompartimento diverso dell'entità a cui appartengono.

2.2.10.1.2 Vincoli

I vincoli nei diagrammi RE specificano le restrizioni sui dati nello schema informativo.

Ci sono quattro tipi di vincoli supportati da Umbrello:

- *Chiave primaria*: l'insieme di attributi dichiarati come *chiave primaria* è unico per l'entità. Ci può essere solo una chiave primari in un'entità e nessuno dei suoi membri può essere nullo.
- *Chiave univoca*: l'insieme di attributi dichiarati come *chiave univoca* è unico per l'entità. Ci possono essere molti vincoli univoci in un'entità. I loro membri possono essere nulli. Le chiavi univoche e primarie identificano univocamente una riga in una tabella (entità).
- *Chiave esterna*: una chiave esterna è un vincolo referenziale tra due tabelle. La chiave esterna identifica una colonna o un insieme di colonne in una tabella (referente) che si riferisce a una colonna o insieme di colonne in un'altra tabella (referenziata). Le colonne nella tabella referenziata devono formare una chiave primaria o univoca.
- *Vincolo di controllo*: un vincolo di controllo (noto anche come vincolo di controllo della tabella) è una condizione che definisce i dati validi quando si aggiunge o si aggiorna una voce di una tabella in una banca dati relazionale. Un vincolo di controllo viene applicato a ogni riga nella tabella. Il vincolo deve essere un predicato. Può riferirsi a una sola o a più colonne della tabella.
  Esempio: prezzo >= 0

2.2.11 Concetti dei diagrammi di relazioni delle entità estesi (REE)

2.2.11.1 Specializzazione

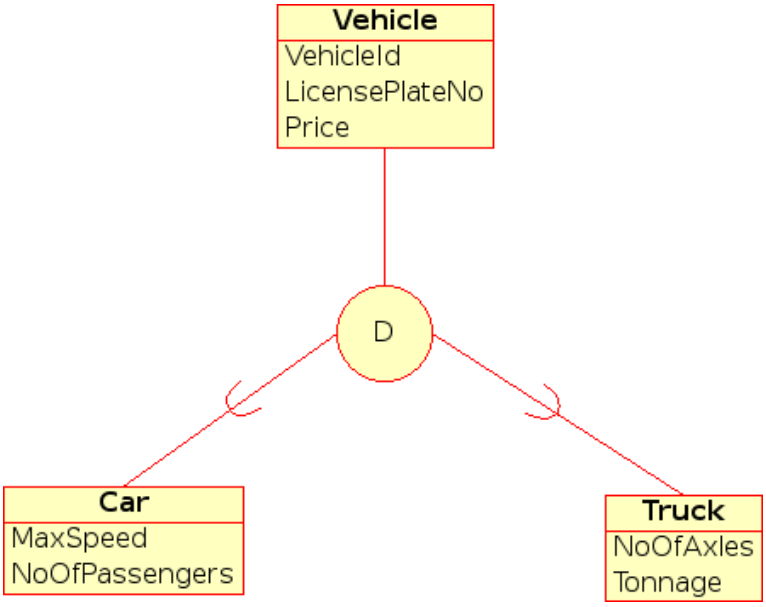
La specializzazione è un modo di formare nuove entità usando entità che sono già state definite. Le nuove entità, note come entità derivate, sostituiscono (o ereditano) gli attributi delle entità

preesistenti, che sono dette entità di base. Ciò è inteso per riutilizzare i dati esistenti con poche o nessuna modifica.

In Umbrello, si possono definire le specializzazioni di «Scollegamento» e «Sovrapposizione».

2.2.11.1.1 Specializzazione disgiunta

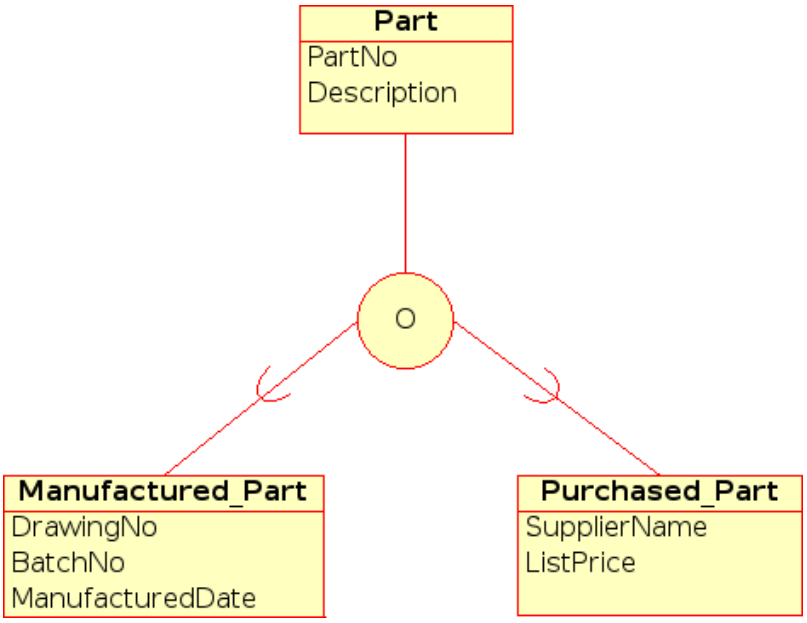
La specializzazione disgiunta specifica che le sottoclassi della specializzazione devono essere disgiunte. Ciò significa che un'entità può fare parte di al più una delle entità derivate dalla specializzazione.



Rappresentazione visiva della specializzazione disgiunta in un diagramma REE

2.2.11.1.2 Specializzazione sovrapposta

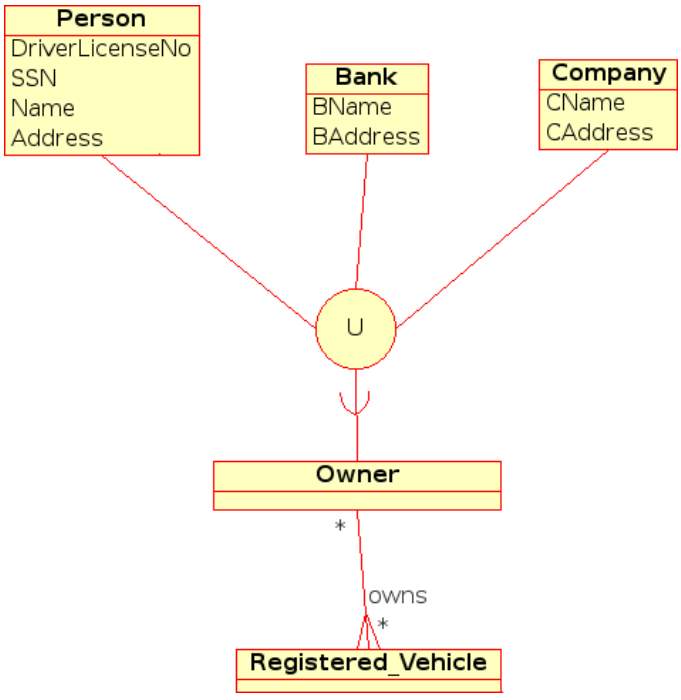
Quando le entità derivate non sono vincolate ad essere disgiunte, il loro insieme di entità è detto essere una specializzazione sovrapposta. Ciò significa che la stessa entità, nel mondo reale, può fare parte di più di una entità derivata dalla specializzazione.



Rappresentazione visiva della specializzazione sovrapposta in un diagramma REE

2.2.11.1.3 Categoria

Un'entità derivata è detta essere una *categoria* quando rappresenta una raccolta di oggetti costituente un sottoinsieme dell'unione dei tipi di entità distinti. Una categoria è modellata quando serve una relazione singola tra superclasse e sottoclasse con più di una superclasse, dove le superclassi rappresentano diversi tipi di entità (in modo simile all'ereditarietà multipla nella programmazione ad oggetti).



Rappresentazione visiva di una categoria in un diagramma REE

Capitolo 3

Lavorare con Umbrello UML Modeller

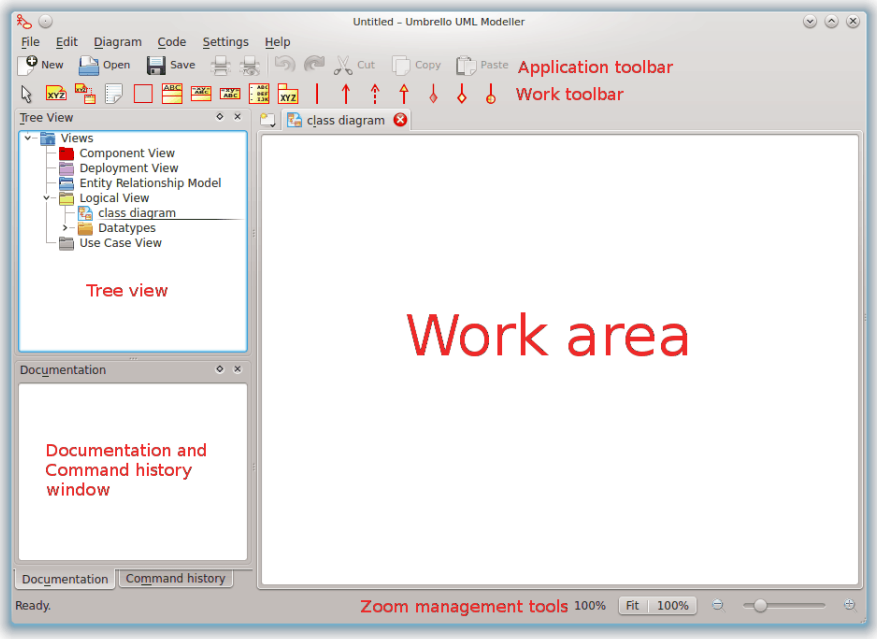
Questo capitolo ti introdurrà all'interfaccia utente di Umbrello UML Modeller e ti dirà tutto quello di cui hai bisogno per iniziare a modellare. Tutte le azioni in Umbrello UML Modeller sono accessibili attraverso il menu e le barre degli strumenti, ma Umbrello UML Modeller usa molto anche i menu contestuali del tasto destro del mouse. Puoi fare clic con il tasto destro del mouse su quasi ogni elemento nell'area di lavoro di Umbrello UML Modeller o nella vista ad albero, per ottenere un menu con le funzioni più utili che possono essere applicate al particolare elemento su cui stai lavorando. Alcuni utenti si sentono un po' confusi all'inizio perché sono abituati a lavorare con il menu o le barre degli strumenti, ma una volta che ti abitui a fare clic con il pulsante destro il tuo lavoro sarà molto più veloce.

3.1 Interfaccia utente

La finestra principale di Umbrello UML Modeller è divisa in tre aree che ti aiuteranno a mantenere una panoramica del tuo intero sistema e ad accedere rapidamente ai diversi diagrammi mentre lavori sul tuo modello.

Queste aree sono chiamate:

- Vista ad albero
- Area di lavoro
- Finestra di documentazione e cronologia dei comandi



L'interfaccia utente di Umbrello UML Modeller

3.1.1 Vista ad albero

La Vista ad albero è normalmente situata all'angolo in alto a sinistra della finestra, e mostra tutti i diagrammi, le classi, gli attori e i casi d'uso che costituiscono il tuo modello. La Vista ad albero ti permette di avere una semplice panoramica degli elementi che compongono il tuo modello. La Vista ad albero ti dà anche un modo semplice per spostarti tra i diversi diagrammi del tuo modello, e inserire elementi dal tuo modello nel diagramma attuale.

Se stai lavorando a un modello con più che poche classi e diagrammi, la Vista ad albero ti può anche aiutare a orientarti organizzando il tuo modello nelle cartelle. Puoi creare delle cartelle selezionando l'opzione apposita dal menu contestuale (fai clic con il tasto destro del mouse su una delle cartelle nella vista ad albero), e puoi organizzare i tuoi elementi spostandoli alla cartella giusta (trascinandoli)

3.1.2 Finestra di documentazione e cronologia dei comandi

La finestra di documentazione e cronologia dei comandi è la finestrella situata nell'angolo in basso a sinistra di Umbrello UML Modeller, e dà una rapida anteprima della documentazione dell'elemento attualmente selezionato e la cronologia dei comandi della sessione di lavoro. La finestra di documentazione è piuttosto piccola perché serve solo a dare solo un'occhiata alla documentazione di un elemento e la panoramica della cronologia dei comandi, occupando il minimo spazio possibile. Se devi vedere la documentazione in maggior dettaglio, puoi sempre aprire le proprietà dell'elemento.

3.1.3 Area di lavoro

L'Area di lavoro è la finestra principale di Umbrello UML Modeller ed è dove avviene il lavoro vero e proprio. Si usa l'Area di lavoro per modificare e visualizzare i diagrammi del tuo modello. L'Area di lavoro mostra il diagramma attualmente attivo. Attualmente si può vedere solo un diagramma alla volta.

3.2 Creare, caricare e salvare i modelli

La prima cosa che devi fare per iniziare a fare qualcosa di utile con Umbrello UML Modeller è creare un modello su cui lavorare. Quando lo avvii, Umbrello UML Modeller carica sempre l'ultimo modello usato o ne crea uno nuovo vuoto (a seconda delle tue preferenze come impostate nella finestra di configurazione). Ciò ti permetterà di iniziare subito a lavorare.

3.2.1 Nuovo modello

Se in qualsiasi momento dovessi creare un nuovo modello, puoi farlo selezionando la voce **Nuovo** nel menu **File**, o facendo clic sull'icona **Nuovo** dalla barra delle applicazioni. Se al momento stai lavorando su un modello che è stato modificato, Umbrello UML Modeller ti chiederà se deve salvare le tue modifiche prima di caricare il nuovo modello.

3.2.2 Salva modello

Puoi salvare il tuo modello in qualsiasi momento selezionando l'opzione **Salva** nel menu **File**, o facendo clic sull'icona **Salva** nella barra delle applicazioni. Se devi salvare il tuo modello con un nome diverso puoi usare l'opzione **Salva come** nel menu **File**.

Per la tua comodità Umbrello UML Modeller ti offre anche l'opzione di salvare automaticamente il tuo lavoro a intervalli regolari. Puoi configurare se vuoi questa opzione, e anche l'intervallo di tempo, nelle **Impostazioni** di Umbrello UML Modeller

3.2.3 Carica modello

Per caricare un modello preesistente puoi selezionare l'opzione **Apri** nel menu **File** o fare clic sull'icona **Apri** nella barra delle applicazioni. I modelli più recentemente usati sono disponibili anche nel sottomenu **Apri recenti** nel menu **File** per facilitare l'accesso ai modelli che usi più spesso.

Umbrello UML Modeller può lavorare solo su un modello alla volta, quindi se chiedi al programma di caricarti un modello e il tuo modello è stato modificato dall'ultima volta che l'hai salvato, Umbrello UML Modeller ti chiederà se le tue modifiche debbano essere salvate per evitare qualsiasi perdita di lavoro. Puoi avviare due o più istanze di Umbrello UML Modeller in qualsiasi momento, e puoi anche copiare e incollare tra le istanze.

3.3 Modificare i modelli

In Umbrello UML Modeller ci sono fondamentalmente due modi per modificare gli elementi del tuo modello.

- Modificare gli elementi del modello direttamente attraverso la Vista ad albero
- Modificare gli elementi del modello attraverso un diagramma

Usando il menu contestuale dei diversi elementi nella Vista ad albero puoi aggiungere, rimuovere e modificare quasi tutti gli elementi nel tuo modello. Fare clic destro sulle cartelle nella Vista ad albero ti darà le opzioni per creare i diversi tipi di diagrammi e anche, a seconda che la cartella sia una *Vista dei casi d'uso* o una *Vista logica*, attori, casi d'uso, classi ecc.

Una volta che hai aggiunto gli elementi nel tuo modello, puoi anche modificarli accedendo alle loro finestre delle proprietà, che puoi trovare selezionando l'opzione *Proprietà* dal menu contestuale mostrato quando fai clic destro sugli elementi nella Vista ad albero.

Puoi anche modificare il tuo modello creando o modificando degli elementi per mezzo dei diagrammi. Maggiori dettagli su come farlo sono dati nelle sezioni successive.

3.4 Aggiungere e rimuovere i diagrammi

Il tuo modello UML consiste di un insieme di elementi UML e associazioni tra di loro. Tuttavia non puoi vedere il modello direttamente, ma usi dei *diagrammi* per vederlo.

3.4.1 Creare i diagrammi

Per creare un nuovo diagramma nel tuo modello, seleziona semplicemente il tipo di diagramma di cui hai bisogno dal sottomenu **Nuovo** nel menu **Diagramma** e dàgli un nome. Il diagramma sarà creato e attivato, e potrai immediatamente vederlo nella Vista ad albero.

Ricorda che Umbrello UML Modeller fa ampio uso dei menu contestuali: puoi anche fare clic con il tasto destro del mouse su una cartella nella Vista ad albero e selezionare il tipo appropriato di diagramma dal sottomenu **Nuovo** nel menu contestuale. Nota che puoi creare diagrammi di casi d'uso solo nelle cartelle di Vista dei casi d'uso, e gli altri tipi di diagrammi solo nelle cartelle di Vista logica.

3.4.2 Rimuovere i diagrammi

Se dovessi rimuovere un diagramma dal tuo modello, puoi farlo attivandolo e selezionando **Elimina** dal menu **Diagramma**. Puoi anche farlo selezionando **Elimina** dal menu contestuale del diagramma nella Vista ad albero

Siccome eliminare un diagramma è qualcosa di serio che potrebbe causare perdite di lavoro se fatto per sbaglio, Umbrello UML Modeller ti chiederà di confermare l'operazione di eliminazione prima di rimuovere per davvero il diagramma. Una volta che un diagramma è stato eliminato e il file salvato, non c'è modo di annullare questa azione.

3.4.3 Rinominare i diagrammi

Se vuoi cambiare il nome di un diagramma esistente puoi farlo facilmente selezionando l'opzione Rinomina dal suo menu del tasto destro del mouse nella vista ad albero.

Un altro modo di rinominare un diagramma è farlo dalla sua finestra delle proprietà, che ottieni selezionando Proprietà dal suo menu contestuale o facendoci doppio clic nella Vista ad albero.

3.5 Modificare i diagrammi

Quando si lavora su un diagramma, Umbrello UML Modeller cercherà di guidarti applicando alcune semplici regole, come quali elementi sono validi nei diversi tipi di diagrammi, e anche le relazioni che possono esistere tra di loro. Se sei un esperto di UML probabilmente non lo noterai nemmeno, ma questo aiuterà i novellini di UML a creare dei diagrammi conformi agli standard.

Una volta che hai creato i tuoi diagrammi è ora di iniziare a modificarli. Qui dovresti notare la differenza (sottile per i novellini) tra modificare il tuo diagramma, e modificare il *modello*. Come già saprai, i diagrammi sono *visualizzazioni* del tuo modello. Per esempio, se crei una classe modificando un diagramma di classe, stai in realtà modificando sia il tuo diagramma che il tuo modello. Se cambi il colore o un'altra opzione di visualizzazione di una classe nel tuo diagramma di classe, stai solo modificando il diagramma, ma non cambia nulla nel tuo modello.

3.5.1 Inserire gli elementi

Una delle prime cose che vorrai fare modificando un nuovo diagramma è inserirci elementi (classi, attori, casi d'uso, ecc.). Ci sono fondamentalmente due modi per farlo:

- Trascinare gli elementi esistenti nel tuo modello dalla Vista ad albero
- Creare i nuovi elementi nel tuo modello e aggiungerli ai tuoi diagrammi allo stesso tempo, usando uno degli strumenti di modifica nella barra degli strumenti di lavoro

Per inserire gli elementi che esistono già nel tuo modello, trascinali dalla Vista ad albero e sganciali dove vuoi che stiano nel tuo diagramma. Puoi sempre spostare gli elementi nel tuo diagramma usando lo strumento Seleziona

Il secondo modo di aggiungere elementi al tuo diagramma è usare gli strumenti di modifica della barra degli strumenti di lavoro (nota che, inoltre, ciò aggiungerà gli elementi al tuo modello).

La barra degli strumenti di lavoro, come impostazione predefinita, era posizionata in cima alla finestra. Gli strumenti disponibili in questa barra degli strumenti (i pulsanti che ci vedi sopra) cambiano a seconda del tipo di diagramma su cui stai lavorando attualmente. Il pulsante per lo strumento attualmente selezionato è attivato nella barra degli strumenti. Puoi passare allo strumento di selezione premendo il tasto **Esc**.

Quando hai selezionato uno strumento di modifica nella barra degli strumenti di lavoro (per esempio, lo strumento per inserire le classi) il puntatore del mouse diventa una croce, e puoi inserire gli elementi nel tuo modello facendo un solo clic nel tuo diagramma. Nota che gli elementi in UML devono avere un *nome univoco*. Quindi, se hai una classe in un diagramma che si chiama 'ClasseA', e usi lo strumento di inserimento classi per inserire una classe in un altro diagramma, non puoi chiamare anche questa nuova classe 'ClasseA'. Se queste due devono essere elementi diversi, devi dargli dei nomi univoci. Se stai cercando di aggiungere lo *stesso* elemento al tuo diagramma, Inserisci classe non è lo strumento giusto. Dovresti invece trascinare la classe dalla Vista ad albero.

3.5.2 Eliminare gli elementi

Puoi eliminare qualsiasi elemento selezionando l'opzione **Elimina** dal suo menu contestuale.

Di nuovo, c'è una *grossa* differenza tra rimuovere un oggetto da un diagramma, ed eliminare un oggetto dal tuo modello: se elimini un oggetto da un diagramma, stai solo rimuovendo l'oggetto da quel diagramma particolare: l'elemento sarà ancora parte del tuo modello, e se ci sono altri diagrammi che usano lo stesso elemento, non subiranno conseguenze. Se, d'altro canto, elimini l'elemento dalla Vista ad albero, stai effettivamente eliminando l'elemento dal tuo *modello*. Siccome l'elemento non esiste più nel tuo modello, sarà automaticamente rimosso da tutti i diagrammi in cui appare.

3.5.3 Modificare gli elementi

Puoi modificare la maggior parte degli elementi e dei diagrammi UML aprendo le loro finestre delle Proprietà e selezionando le opzioni appropriate. Per modificare le proprietà di un oggetto, seleziona **Proprietà** dal suo menu contestuale (fai clic con il tasto destro del mouse). Ogni elemento ha una finestra di dialogo consistente in diverse pagine dove puoi configurare le opzioni corrispondenti a quell'elemento. Per alcuni elementi, come gli attori, puoi impostare solo un paio di opzioni, come il nome e la documentazione dell'oggetto, mentre per altri elementi, come le classi, puoi modificare gli attributi e le operazioni, selezionare cosa vuoi sia mostrato nel diagramma (la firma completa delle operazioni o solo i nomi delle operazioni, ecc.) e anche i colori che vuoi usare per la linea e per il riempimento della rappresentazione della classe nel diagramma.

Per gli elementi UML puoi anche aprire la finestra delle proprietà facendoci doppio clic se stai usando lo strumento di selezione (freccia).

Nota che puoi anche selezionare le opzioni delle proprietà dal menu contestuale degli elementi nella vista ad albero. Ciò ti permette anche di modificare le proprietà dei diagrammi, come impostare se la griglia debba essere mostrata o meno.

3.5.4 Modificare le classi

Anche se è stato già descritto come modificare le proprietà di tutti gli oggetti nella sezione precedente, le classi meritano una sezione speciale perché sono un po' più complicate e hanno più opzioni della maggior parte degli altri elementi UML.

Nella finestra delle proprietà di una classe puoi impostare tutto, dal colore usato alle operazioni e attributi che ha.

3.5.4.1 Impostazioni generali delle classi

La pagina Impostazioni generali della finestra delle proprietà si spiega da sé. Qui puoi cambiare il nome della classe, la visibilità, la documentazione, ecc. Questa pagina è sempre disponibile.

3.5.4.2 Impostazione degli attributi delle classi

Nella pagina Attributi delle classi puoi aggiungere, modificare o eliminare gli attributi (le variabili) della classe. Puoi spostare gli attributi in su e in giù nell'elenco premendo il pulsante a freccia di fianco. Questa pagina è sempre disponibile.

3.5.4.3 Impostazioni delle operazioni delle classi

In modo simile alla pagina delle Impostazioni degli attributi, nella pagina delle Impostazioni delle operazioni puoi aggiungere, modificare o rimuovere le operazioni della tua classe. Quando aggiungi o modifichi un'operazione, inserisci i dati di base nella finestra *Proprietà delle operazioni*. Se vuoi aggiungere dei parametri alle tue operazioni devi fare clic sul pulsante **Nuovo parametro**, che mostrerà la finestra *Proprietà del parametro*. Questa pagina è sempre disponibile

3.5.4.4 Impostazioni dei modelli delle classi

Questa pagina ti permette di aggiungere modelli di classi, che sono classi o tipi di dati non specificati. In Java 1.5 saranno chiamati Generics.

3.5.4.5 Pagina delle associazioni delle classi

La pagina **Associazioni della classe** mostra tutte le associazioni di questa classe nel diagramma attuale. Fare doppio clic su un'associazione ne mostra le proprietà, e a seconda del tipo di associazione qui puoi modificare alcuni parametri, come l'impostazione di molteplicità e il nome del ruolo. Se l'associazione non permette la modifica di queste opzioni, la finestra Proprietà dell'associazione è di sola lettura, e puoi modificare solo la documentazione associata a questa associazione.

Questa pagina è disponibile solo se apri le Proprietà della classe da un diagramma. Se selezioni le proprietà della classe dal menu contestuale nella Vista ad albero questa pagina non è disponibile.

3.5.4.6 Pagina di visualizzazione delle classi

Nella pagina **Opzioni di visualizzazione**, puoi impostare cosa debba essere visualizzato nel diagramma. Una classe può essere mostrata solo come un rettangolo con dentro il nome della classe (utile se hai molte classi nel tuo diagramma, o per ora non sei interessato ai dettagli di ogni classe), o completamente, mostrando pacchetti, stereotipi, e attributi e operazioni con firma completa e visibilità

A seconda della quantità di informazione che vuoi vedere, puoi selezionare le opzioni corrispondenti in questa pagina. I cambiamenti che fai qui sono solo *opzioni di visualizzazione* del diagramma. Ciò vuol dire che 'nascondere' le operazioni di una classe non le fa apparire nel diagramma, ma le operazioni sono ancora parte del tuo modello. Questa opzione è disponibile solo se selezioni le proprietà della classe da un diagramma. Se apri le proprietà della classe dalla Vista ad albero, questa pagina manca, perché queste opzioni di visualizzazione, in quel caso, non hanno senso

3.5.4.7 Pagina degli stili delle classi

Nella pagina **Stile dell'oggetto** puoi configurare i colori che vuoi usare per la linea e il riempimento dell'oggetto. Questa opzione ovviamente ha senso solo per le classi visualizzate nei diagrammi, e manca se apri la finestra delle proprietà delle classi dalla Vista ad albero.

3.5.5 Associazioni

Le associazioni mettono in relazione due oggetti UML. Normalmente le associazioni sono definite tra due classi, ma alcuni tipi di associazioni possono esistere anche tra casi d'uso e attori.

Per creare un'associazione seleziona lo strumento appropriato dalla barra degli strumenti di lavoro (associazione generica, generalizzazione, aggregazione, ecc.), fai clic sul primo elemento che partecipa all'associazione e quindi fai clic sul secondo elemento partecipante. Nota che ci sono due clic, uno su ciascuno degli oggetti partecipanti all'associazione, *non* è un trascinamento da un oggetto all'altro.

Se provi a usare un'associazione in un modo contrario alla specificazione UML, Umbrello UML Modeller si rifiuterà di creare l'associazione e riceverai un messaggio d'errore. Sarebbe il caso se, per esempio, esistesse una generalizzazione dalla classe A alla classe B e quindi provassi a creare un'altra generalizzazione dalla classe B alla classe A

Fare clic destro su un'associazione mostrerà un menu contestuale con le azioni che puoi applicarci. Se devi eliminare un'associazione seleziona semplicemente l'opzione **Elimina** da questo menu contestuale. Puoi anche selezionare l'opzione **Proprietà** e, a seconda del tipo di associazione, modificare attributi come ruoli e molteplicità.

3.5.5.1 Punti di ancoraggio

Le associazioni sono disegnate, come impostazione predefinita, come una linea retta che connette i due oggetti nel diagramma.

Puoi aggiungere dei punti di ancoraggio per piegare un'associazione facendo doppio clic da qualche parte lungo la linea di associazione. Ciò inserirà un punto di ancoraggio (mostrato come un punto blu quando la linea di associazione è selezionata) che puoi spostare per dare forma all'associazione

Se hai bisogno di rimuovere un punto di ancoraggio, facci di nuovo doppio clic per rimuoverlo

Nota che l'unico modo di modificare le proprietà di un'associazione è con il menu contestuale. Se provi a farci doppio clic come su qualsiasi altro oggetto UML, verrà soltanto inserito un punto di ancoraggio.

3.5.6 Note, testo e riquadri

Le note, le righe di testo e i riquadri sono elementi che possono essere presenti in ogni tipo di diagramma e non avere nessun valore semantico, ma sono molto utili per aggiungere dei commenti o delle spiegazioni che possono rendere il tuo diagramma più facile da capire.

Per aggiungere una nota o una riga di testo, seleziona lo strumento corrispondente dalla barra degli strumenti di lavoro, e fai clic sul diagramma dove vuoi mettere il tuo commento. Puoi modificare il testo aprendo l'elemento con il menu contestuale o, nel caso delle note, anche facendoci doppio clic.

3.5.6.1 Ancore

Le ancore sono usate per collegare insieme una nota di testo e un altro elemento UML. Per esempio, normalmente usi una nota di testo per spiegare o fare qualche commento su una classe o un'associazione particolare, nel qual caso puoi usare l'àncora per mettere in chiaro che la nota 'appartiene' a quell'elemento particolare.

Per aggiungere un'àncora tra una nota e un altro elemento UML, usa lo strumento àncora dalla barra degli strumenti di lavoro. Devi innanzi tutto fare clic sulla nota e quindi sull'elemento UML che vuoi collegare alla nota.

Capitolo 4

Importazione e generazione di codice

Umbrello UML Modeller è uno strumento di modellazione UML, e come tale il suo obiettivo principale è aiutarti *nell'analisi e nella progettazione* dei tuoi sistemi. Tuttavia, per facilitare la transizione tra il tuo progetto e la tua *implementazione*, Umbrello UML Modeller ti permette di generare codice sorgente in diversi linguaggi di programmazione come inizio. Inoltre, se vuoi iniziare a usare UML in un progetto C++ già cominciato, Umbrello UML Modeller può aiutarti a creare un modello del tuo sistema dal codice sorgente analizzandolo e importando le classi trovateci.

4.1 Generazione di codice

Umbrello UML Modeller può generare codice sorgente per vari linguaggi di programmazione in base al tuo modello UML per aiutarti a iniziare l'implementazione del tuo progetto. Il codice generato consiste di dichiarazioni di classi, con i loro metodi e attributi in modo che tu possa 'riempire gli spazi vuoti' fornendo la funzionalità delle operazioni delle tue classi.

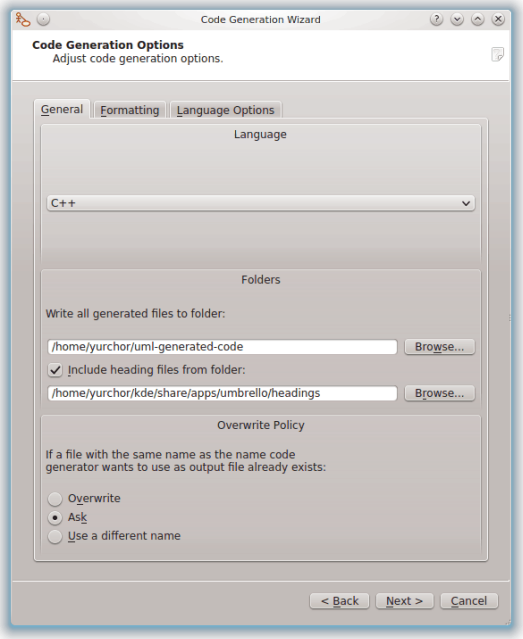
Umbrello UML Modeller 2 ha supporto per la generazione di codice ActionScript, Ada, C++, C#, D, IDL, Java™, JavaScript, MySQL, Pascal, Perl, PHP, PHP5, PostgreSQL, Python, Ruby, Tcl, Vala, e XMLSchema.

4.1.1 Generare il codice

Per generare il codice con Umbrello UML Modeller, devi prima creare o caricare un modello contenente almeno una classe. Quando sei pronto per iniziare a scrivere del codice, seleziona la voce **Assistente per la generazione del codice** dal menu **Codice** per avviare un assistente che ti guiderà attraverso il processo di generazione del codice.

Il primo passo è selezionare le classi per le quali vuoi generare del codice sorgente. Come impostazione predefinita tutte le classi del tuo modello sono selezionate, e puoi rimuovere quelle per le quali non vuoi generare codice spostandole nell'elenco sulla sinistra.

Il prossimo passo dell'assistente ti permette di modificare i parametri che il generatore di codice userà scrivendo il tuo codice. Sono disponibili le seguenti opzioni:



Opzioni per la generazione del codice in Umbrello UML Modeller

4.1.1.1 Opzioni di generazione

4.1.1.1.1 Verbosità dei commenti

L'opzione **Scrivi commenti di documentazione anche se vuoto** dice al generatore di codice di scrivere commenti nello stile /** bla */ anche se i blocchi di commento sono vuoti. Se hai aggiunto della documentazione alle tue classi, metodi o attributi nel tuo modello, il generatore di codice scriverà questi commenti come documentazione Doxygen, indipendentemente da cosa imposti qui, ma se selezioni questa opzione Umbrello UML Modeller scriverà blocchi di commento per tutte le classi, i metodi e gli attributi anche se non c'è documentazione nel modello, nel qual caso dovresti documentare le tue classi in seguito, direttamente nel codice sorgente.

Scrivi commenti per le sezioni anche se la sezione è vuota fa in modo che Umbrello UML Modeller scriva commenti nel codice sorgente per delimitare le diverse sezioni di una classe. Per esempio 'metodi pubblici' o 'Attributi' prima delle sezioni corrispondente. Se selezioni questa opzione Umbrello UML Modeller scriverà commenti per tutte le sezioni della classe anche se la sezione è vuota. Per esempio, scriverebbe un commento che dice 'metodi protetti' anche se non ci sono metodi protetti nella tua classe.

4.1.1.1.2 Cartelle

Scrivi tutti i file generati nella cartella. Qui dovresti scegliere la cartella dove vuoi che Umbrello UML Modeller metta il sorgente generato.

L'opzione **Includi i file di intestazione dalla cartella** ti permette di inserire un'intestazione all'inizio di ogni file generato. I file d'intestazione possono contenere informazioni di copyright o licenza e contenere variabili che sono valutate al momento della generazione. Puoi dare un'occhiata ai modelli di file d'intestazione forniti con Umbrello UML Modeller per vedere come usare queste variabili per sostituire il tuo nome o la data attuale al momento della generazione.

4.1.1.1.3 Politica di sovrascrittura

Questa opzione dice a Umbrello UML Modeller cosa fare se il file che vuole creare esiste già nella cartella di destinazione. Umbrello UML Modeller *non può modificare file sorgente esistenti*, quindi devi scegliere tra sovrascrivere il file esistente, saltare la generazione di un certo file o lasciare che Umbrello UML Modeller scelga un nome file diverso. Se scegli l'opzione di usare un nome diverso, Umbrello UML Modeller aggiungerà un suffisso al nome file.

4.1.1.1.4 Linguaggio

Umbrello UML Modeller genererà codice, come impostazione predefinita, nel linguaggio che hai selezionato come Linguaggio attivo, ma con l'Assistente per la generazione del codice hai l'opzione di cambiarlo in un altro linguaggio.

4.1.1.2 Generazione con l'assistente di generazione

Il terzo e ultimo passo dell'assistente mostra lo stato del processo di generazione del codice. Devi solo fare clic sul pulsante Genera per avere le tue classi scritte.

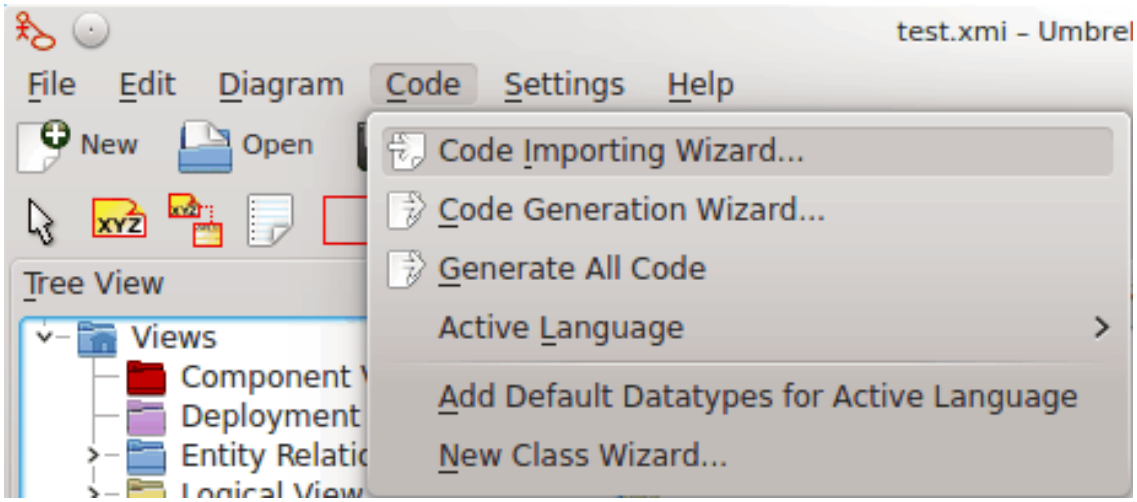
Nota che le opzioni che selezioni nell'assistente per la generazione del codice sono valide solo per la generazione corrente. La prossima volta che esegui l'assistente dovrai riselezionare tutte le opzioni (la cartella delle intestazioni, la politica di sovrascrittura, e così via). Puoi impostare i valori predefiniti usati da Umbrello UML Modeller nella sezione **Generazione del codice** nelle impostazioni di Umbrello UML Modeller, disponibile a **Impostazioni** → **Configura Umbrello UML Modeller...**

Se devi impostare le tue opzioni di generazione del codice ai valori giusti e vuoi subito generare del codice subito senza passare per l'assistente, puoi selezionare l'intero **Genera tutto il codice** dal menu **Codice**. Ciò genererà codice per tutte le classi nel tuo modello usando le impostazioni correnti (incluse la cartella di output e la politica di sovrascrittura, quindi usalo con attenzione).

4.2 Importazione di codice

Umbrello UML Modeller può importare codice sorgente dai tuoi progetti esistenti per aiutarti a costruire un modello del tuo sistema. Umbrello UML Modeller 2 supporta codice sorgente in ActionScript, Ada, C++, C#, D, IDL, Java™, Javascript, MySQL, Pascal, PHP, e Vala.

Per importare classi nel tuo modello, seleziona la voce **Assistente di importazione del codice** dal menu **Codice**. Nella finestra apposita, seleziona i file contenenti le dichiarazioni delle classi C++ e premi **Successivo**, quindi **Avvia importazione** e **Completa**. Le classi saranno importare e le troverai come parte del modello nella vista ad albero. Nota che Umbrello UML Modeller non creerà nessun tipo di diagramma per mostrarti le classi, esse saranno solo importate nel modello in modo che da poterle usare in seguito in qualsiasi diagramma tu voglia.



Menu per importare codice sorgente in Umbrello UML Modeller

Capitolo 5

Altre funzionalità

5.1 Altre funzionalità di Umbrello UML Modeller

Questo capitolo spiegherà brevemente alcune altre funzionalità che Umbrello UML Modeller ti offre.

5.1.1 Copiare oggetti come immagini PNG

Oltre a offrirti la funzionalità normale di copia, taglia e incolla che ti aspetteresti per copiare oggetti tra diversi diagrammi, Umbrello UML Modeller può copiare oggetti come immagini PNG in modo che tu possa inserirli in un altro tipo di documento. Non hai bisogno di fare nulla di speciale per usare questa funzionalità, seleziona semplicemente un oggetto da un diagramma (Classe, Attore, ecc.) e copialo (**Ctrl**-**C**, o usando il menu), quindi apri un documento di Calligra Words (o qualsiasi programma in cui puoi incollare immagini) e seleziona **Incolla**. Questa è un'ottima funzionalità per esportare parti del tuo diagramma come semplici immagini.

5.1.2 Esportare come immagine

Puoi anche esportare un diagramma completo come immagine. L'unica cosa che devi fare è selezionare il diagramma che vuoi esportare, e l'opzione **Esporta come immagine...** dal menu **Diagramma**.

Puoi esportare diversi diagrammi contemporaneamente usando l'opzione **Esporta i diagrammi come immagini...** nel menu **File**. Tramite essa puoi anche impostare la risoluzione dell'immagine, in modo che le immagini non saranno sfocate.

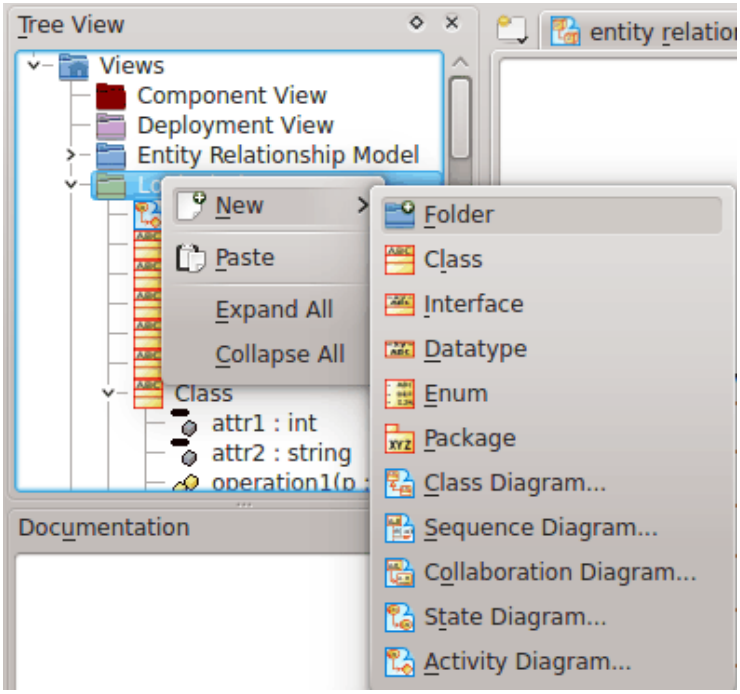
5.1.3 Stampa

Umbrello UML Modeller ti permette di stampare diagrammi individuali. Premere il pulsante **Stampa** sulla barra delle applicazioni o selezionare l'opzione **Stampa** dal menu **File** ti porterà a una finestra di dialogo di stampa standard di KDE, dalla quale puoi stampare i tuoi diagrammi.

5.1.4 Cartelle logiche

Per organizzare meglio il tuo documento, specialmente per i progetti più grandi, puoi creare cartelle logiche nella Vista ad albero. Seleziona l'opzione **Nuovo** → **Cartella** dal menu contestuale

delle cartelle predefinite nella Vista ad albero per crearle. Le cartelle possono essere annidate, e puoi spostare gli oggetti trascinandoli da una cartella e sganciandoli in un'altra.

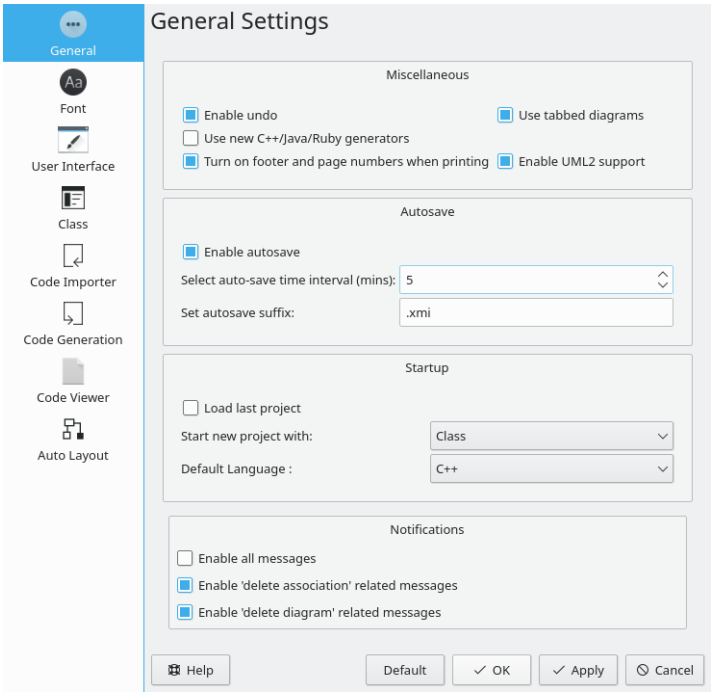


Organizzare un modello con le cartelle logiche in Umbrello UML Modeller

Capitolo 6

Impostazioni

6.1 Impostazioni generali



Opzioni per le impostazioni generali in Umbrello UML Modeller

6.1.1 Varie

- L'opzione **Abilita l'annullamento** permette di annullare un'azione precedente.
- **Usa i nuovi generatori C++/Java/Ruby** permette all'utente di selezionare i nuovi o i vecchi generatori
- Quando **Attiva i piè e i numeri di pagina durante la stampa** è selezionata, stampa le informazioni del diagramma per il diagramma in stampa e il numero di pagina.
- **Usa diagrammi a scheda** rende possibile avere diverse finestre a schede di diagrammi aperte nello stesso momento.

6.1.2 Salvataggio automatico

- **Abilita il salvataggio automatico** rende possibile effettuare il salvataggio automatico del file.
- **Seleziona l'intervallo per il salvataggio automatico (minuti):** permette di impostare il tempo prima che il file sia salvato automaticamente.
- Il valore predefinito di **Imposta il suffisso per il salvataggio automatico:** è .xmi ma l'opzione consente di impostare un'estensione di file diversa.

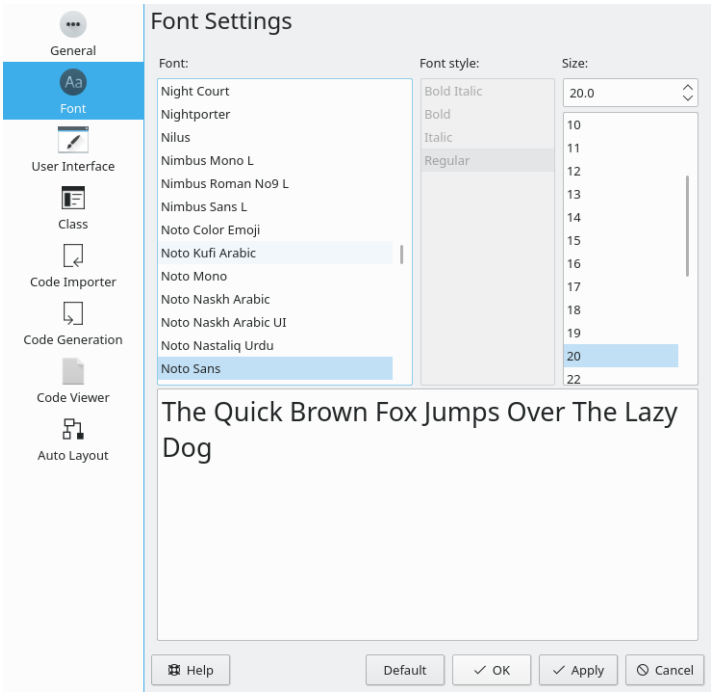
6.1.3 Avvio

- Se **Carica l'ultimo progetto** è impostata, carica sempre l'ultimo progetto aperto all'avvio del programma.
- **Avvia nuovo progetto con:** permette di scegliere con quale tipo di diagramma UML iniziare in un nuovo progetto.
- **Linguaggio predefinito:** è un'impostazione per il linguaggio di programmazione predefinito usato.

6.1.4 Notifiche

- **Attiva tutti i messaggi** è un'opzione per vedere tutte o un'insieme limitato di notifiche.
- **Abilita i messaggi relativi all'eliminazione di associazioni**, se selezionata, garantisce la ricezione di tutti i messaggi di questo tipo.
- **Abilita i messaggi relativi all'eliminazione di diagrammi**, se selezionata, abilita tutti i messaggi di questo tipo.

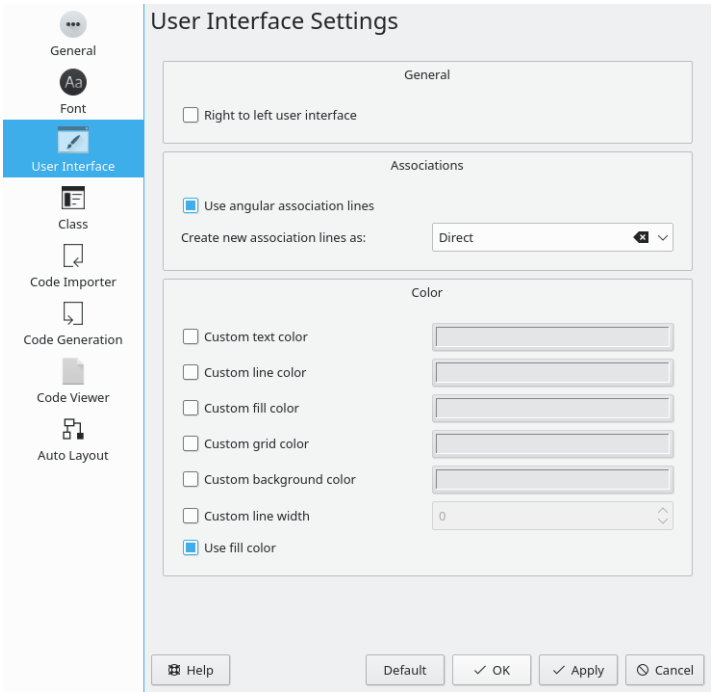
6.2 Impostazioni dei caratteri



Opzioni per le impostazioni del carattere del diagramma in Umbrello UML Modeller

Queste impostazioni del carattere impostano le caratteristiche del testo nei diagrammi. Lo stile e la dimensione del carattere sono le uniche opzioni selezionabili.

6.3 Impostazioni per l'interfaccia utente



Opzioni per le impostazioni per l'interfaccia utente in Umbrello UML Modeller

6.3.1 Generale

Interfaccia utente da destra a sinistra configura l'interfaccia per i linguaggi con scrittura da destra a sinistra.

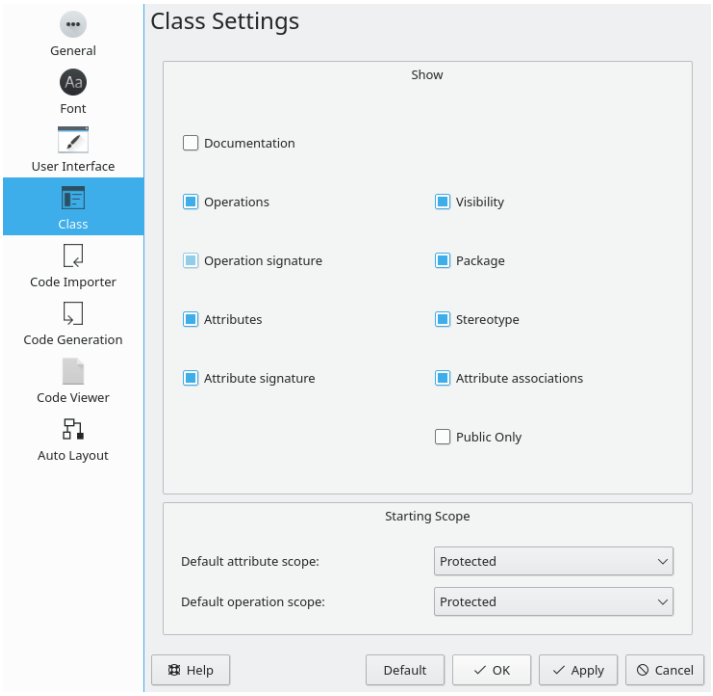
6.3.2 Associazioni

Crea nuove linee di associazione come: rende possibile cambiare lo stile delle linee di associazione.

6.3.3 Colore

La selezione del colore mette a disposizione diverse opzioni per cambiare i colori di testo, linee, riempimento, griglie e sfondo, nonché lo spessore delle linee.

6.4 Impostazioni della classe



Opzioni per le impostazioni della classe in Umbrello UML Modeller

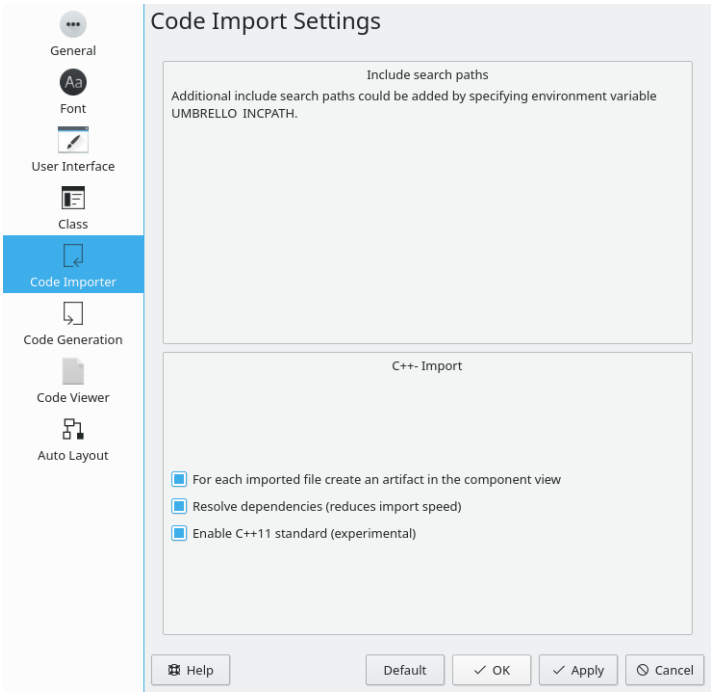
6.4.1 Mostra

La sezione Mostra presenta diverse impostazioni che determinano quali caratteristiche di una classe sono mostrate nel diagramma di classe.

6.4.2 Ambito di partenza

Scelte per le impostazioni predefinite di attributi e operazioni, se pubblici, privati o protetti.

6.5 Impostazioni di importazione del codice



Opzioni per le impostazioni di importazione del codice in Umbrello UML Modeller

6.5.1 Percorsi di ricerca dei file di inclusione

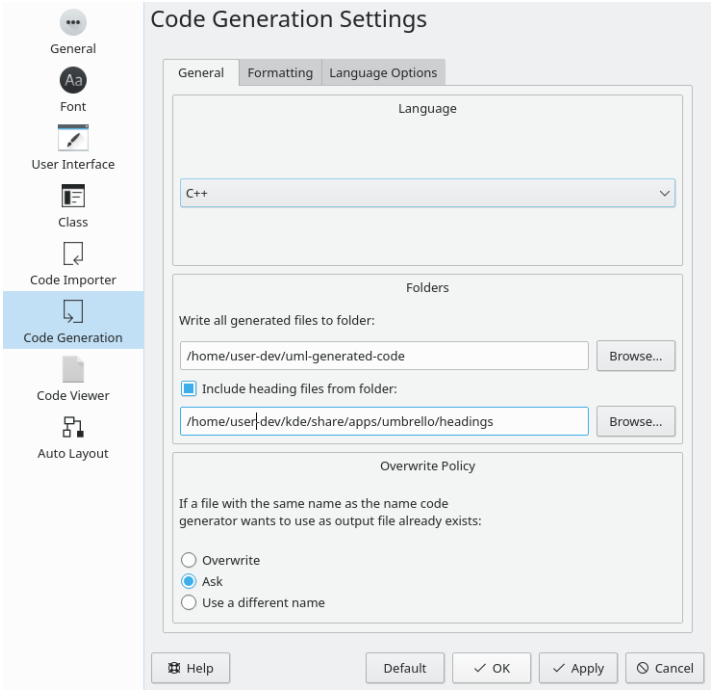
Viene fornita una raccomandazione generale per migliorare la ricerca tramite l'inclusione di UMBRELLO_INCPATH come variabile d'ambiente.

6.5.2 Importazione C++

- **Per ogni file importato crea un artefatto nella vista dei componenti** L'artefatto creato potrà quindi essere trascinato nella vista del diagramma delle classi, dove le dipendenze possono essere visualizzate in modo semplice assieme agli attributi e alle funzioni di ogni file.
- **Risolvi le dipendenze (riduce la velocità di importazione)** Verifica che tutte le dipendenze del file siano risolte, e quindi saranno visibili nelle dipendenze della classe del diagramma delle classi.
- **Abilita lo standard C++11 (sperimentale)** Una funzione sperimentale per la conformità allo standard C++11, disabilitare se non necessaria.

6.6 Impostazioni per la generazione del codice

6.6.1 Scheda generale delle impostazioni per la generazione del codice



Opzioni per le impostazioni generali per la generazione del codice in Umbrello UML Modeller

Il modellatore UML Umbrello può generare codice sorgente per vari linguaggi di programmazione in base al modello UML scelto per aiutarti a iniziare l'implementazione del tuo progetto. Il codice generato consiste di dichiarazioni di classi, con i loro metodi e attributi in modo che tu possa «riempire gli spazi vuoti» fornendo la funzionalità delle operazioni delle tue classi.

6.6.1.1 Linguaggio

Scegli il linguaggio di programmazione da usare per i progetti. Le scelte disponibili sono ActionScript, Ada, C++, C#, D, IDL, Java, JavaScript, MYSQL, Pascal, Perl, PHP, PHP5, PastgreSQL, Python, Ruby, SQL, Tcl, Vala e XMLSchema

6.6.1.2 Cartelle

Scrivi tutti i file generati nella cartella: ha un campo modificabile per il percorso desiderato per i file generati o, opzionalmente, un pulsante sfoglia per selezionare il percorso.

Includi i file di intestazione dalla cartella: se selezionata, l'utente può indicare un percorso tramite un campo modificabile o sceglierlo con il pulsante sfoglia.

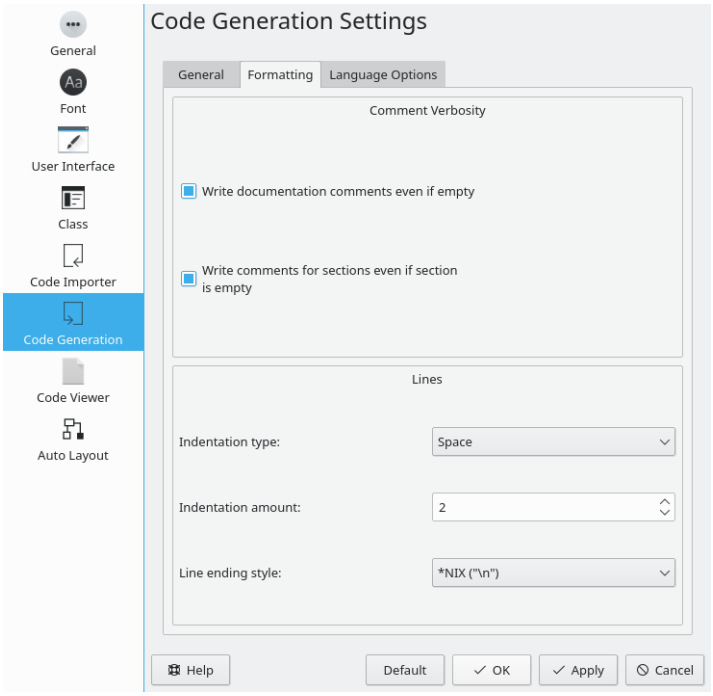
6.6.1.3 Linee guida di sovrascrittura

Quando il codice viene generato nella cartella specificata, questa opzione determina cosa succede se viene trovato un file con lo stesso nome.

- **Sovrascrivi** fà sì che il file sia sovrascritto senza avvisi o opzioni.

- **Chiedi** fa sì che viene chiesto se sovrascrivere il file o rinominarlo.
- **Usa un nome diverso** fa sì che il file sia rinominato aggiungendo un suffisso se esiste già.

6.6.2 Scheda formattazione delle impostazioni per la generazione del codice



Opzioni per le impostazioni di formattazione per la generazione del codice in Umbrello UML Modeller

6.6.2.1 Verbosità dei commenti

Scrivi commenti di documentazione anche se vuoti Genera commenti per la documentazione di classi e funzioni anche se sono vuote.

Scrivi commenti di sezione: permette agli utenti di scegliere se scrivere i commenti di sezione. I valori possibili sono: No, Quando la sezione non è vuota, Sempre.

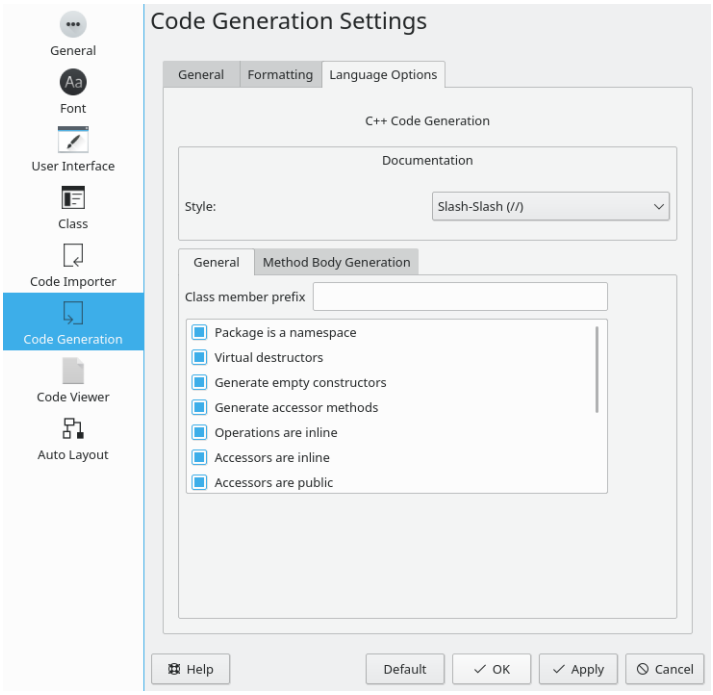
6.6.2.2 Linee

Tipo di rientro: offre una scelta tra nessuna indentazione, tabulatori o spazi.

Quantità di rientro: permette all'utente di specificare il numero di spazio per le scelte di indentazione con tabulatori o spazi.

Stile di fine riga: è la scelta tra gli stile di fine riga di *NIX, Windows e Mac.

6.6.3 Opzioni di linguaggio



Opzioni per le impostazioni del linguaggio per la generazione del codice in Umbrello UML Modeller

Questa pagina cambia per ogni linguaggio di programmazione selezionato nella scheda Generale. Attualmente le uniche opzioni disponibili sono quelle per il linguaggio C++.

6.6.3.1 Generazione di codice C++

6.6.3.1.1 Documentazione

Stile: permette di scegliere se usare «/** */» or «//» come stile di documentazione

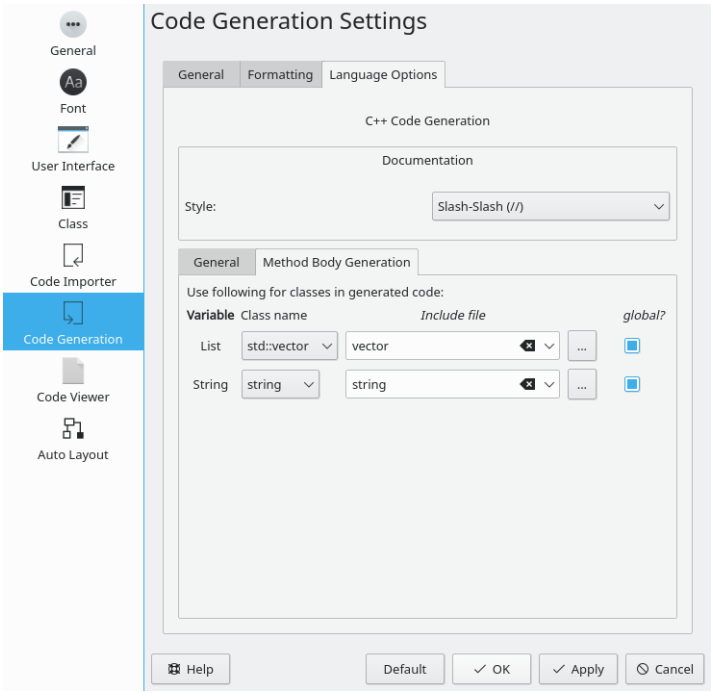
6.6.3.1.2 Generale

Nella scheda **Generale** della scheda **Opzioni di linguaggio** sono mostrate diverse opzioni relativa alla generazione del codice.

- **Prefisso dei membri di classe**
  Un'opzione che permette di aggiungere ai membri della classe, quando il codice viene generato, un prefisso determinato dall'utente.
- **Il pacchetto è uno spazio di nomi**
  Gli spazi dei nomi (namespace) forniscono un modo per prevenire i conflitti di nomi in progetti grandi. I simboli dichiarati dentro il blocco di uno spazio di nomi sono inseriti in un ambito con nome che evita che vengono scambiati per simboli con lo stesso nome in altri ambiti.
- **Distruttori virtuali**
  Anche se i distruttori non sono ereditati, se una classe base dichiara il proprio distruttore come virtuale, il distruttore derivato lo sovrascriverà sempre. Questo rende possibile eliminare gli oggetti allocati dinamicamente di tipo polimorfici attraverso puntatori alla base.

- **Genera costruttori vuoti**
  Questo genererà costruttori che hanno parentesi vuote.
- **Genera metodi delle funzioni d'accesso**
  Genererà i metodi per accedere ai tipi di dati.
- **Le operazioni sono in linea**
  Genera i metodi come in linea, ma i compilatori sono liberi di scegliere di non rendere il metodo in linea.
- **Le funzioni d'accesso sono in linea**
  I metodi che effettuano l'accesso ai dati della classe saranno generati con l'attributo inline, ma i compilatori sono liberi di scegliere di non rendere il metodo in linea.
- **Le funzioni d'accesso sono pubbliche**
  I metodi che sono generati con visibilità pubblica saranno disponibile ogni volta che si crea un'istanza della classe.
- **Crea metodi getter con il prefisso «get»**
  Questo aggiungerà il prefisso «get» ai metodi che ottengono/restituiscono i dati della classe.
- **Rimuovi il prefisso '[a-zA-Z]_' dai nomi dei metodi accessori**
  Se è stato specificato un prefisso in **Prefisso dei membri di classe**, questo lo rimuoverà.
- **I metodi accessori iniziano con lettere maiuscole**
  Rende maiuscola la prima lettera del nome del metodo.
- **Usa «\» come indicatore di documentazione invece di «@»**
  Quali indicatori usare quando si documentano i parametri di un metodo.

6.6.3.1.3 Metodo di generazione del corpo



Opzioni per la impostazioni di linguaggio per generazione del codice del corpo dei metodi in Umbrello UML Modeller

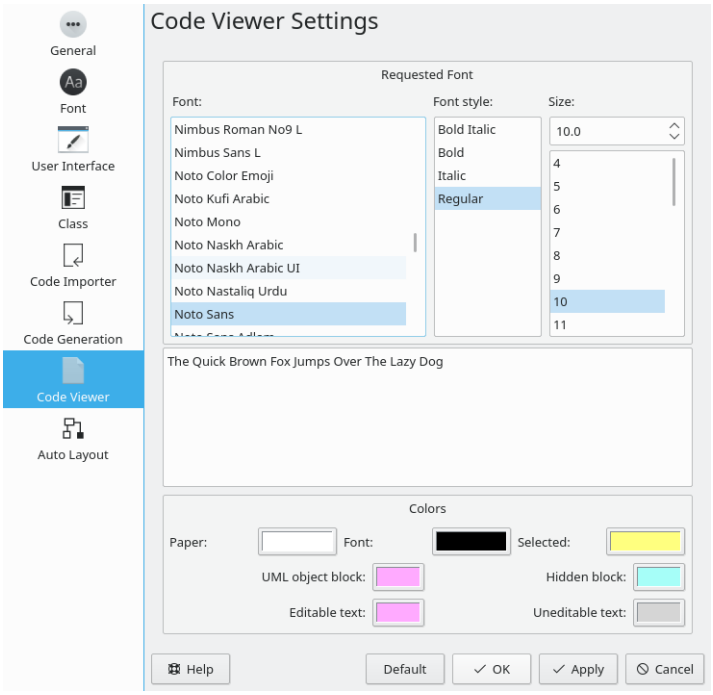
Lista

Ha come opzioni QPtrList, vector e std::vector per il tipo della lista. Segue un campo modificabile o selezionabile per specificare il file di inclusione, assieme ad un pulsante sfoglia per trovare e selezionare il file di inclusione. C'è anche un'opzione per rendere globale la lista.

Stringa

Opzioni di string o QString per il tipo stringa. Segue un campo modificabile o selezionabile per specificare il file di inclusione, assieme ad un pulsante sfoglia per trovare e selezionare il file di inclusione. C'è anche un'opzione per rendere globale la stringa.

6.7 Impostazioni per la visualizzazione del codice

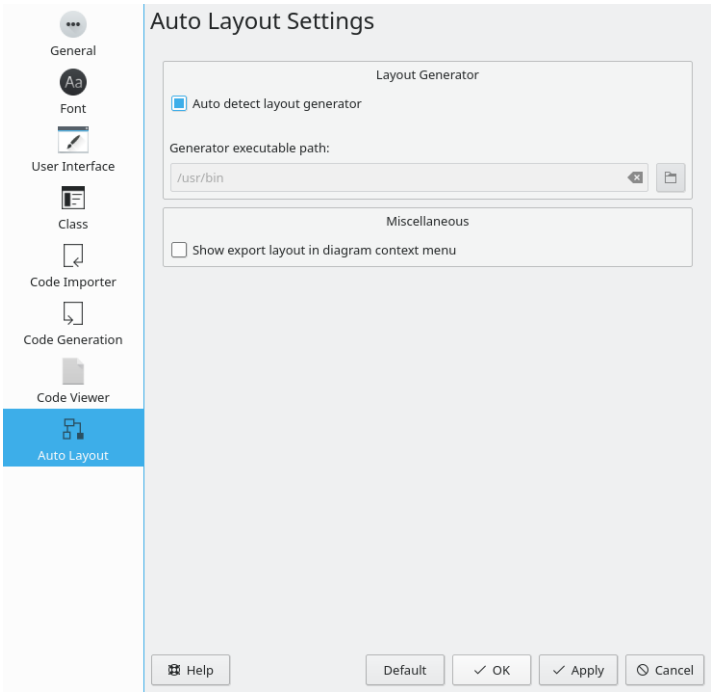


Opzioni per le impostazioni per la visualizzazione del codice in Umbrello UML Modeller

Permette la personalizzazione del visore del codice. La sezione **Carattere richiesto** permette la selezione del carattere, e dello stile e della dimensione del carattere. Una rappresentazione della scelta effettuata viene mostrato sotto le possibili scelte.

Nella sezione **Colori**, le modifiche possono essere effettuate alle seguenti entità: carta, carattere, selezionato, blocco oggetto UML, blocco nascosto, testo modificabile e testo non modificabile. Modifiche ai colori possono essere effettuate facendo clic nelle rispettive etichette sulla casella del colore.

6.8 Impostazioni della disposizione automatica



Opzioni per le impostazioni della disposizione automatica in Umbrello UML Modeller

Rileva automaticamente generatore di disposizioni

La funzionalità di disposizione automatica dipende dai generatori di disposizioni presenti nel pacchetto GraphViz, che è solitamente installato assieme a Umbrello dal gestore dei pacchetti. Umbrello rileva da solo la presenza di questi generatori di disposizioni. Nel caso che questa dipendenza non sia disponibile o non sia quella giusta, è possibile selezionare un altro percorso di installazione.

Mostra disposizione di esportazione nel menu contestuale del diagramma

L'esportazione dei file dot è effettuata usando la disposizione di esportazione. Quando questa opzione è segnata, la disposizione di esportazione viene aggiunta all'elenco di disposizioni di diagrammi disponibili e abilita una rapida anteprima dell'esportazione a dot.

Capitolo 7

Autori e storia

Il progetto è stato iniziato da Paul Hengsen come uno dei suoi progetti universitari. Il nome originale dell'applicazione era UML Modeller. Paul fece tutto lo sviluppo fino alla fine del 2001, quando il programma raggiunse la versione 1.0.

La versione 1.0 offriva già molte funzionalità, ma dopo che il progetto è stato valutato all'università di Paul, altri sviluppatori hanno potuto unirsi dando contributi di valore a UML Modeller, come passare da un file binario a un file XML, supporto per più tipi di diagrammi UML, generazione e importazione di codice solo per menzionarne alcuni.

Paul si è dovuto ritirare dalla squadra di sviluppo nell'estate 2002, ma, come software libero, il programma continua a migliorare e evolversi, ed è mantenuto da un gruppo di sviluppatori da diverse parti del mondo. Nel settembre 2002 il progetto ha cambiato il suo nome da UML Modeller in Umbrello UML Modeller. Ci sono diverse ragioni per il cambio di nome, la più importante è che solo 'uml', come era comunemente noto, era un nome troppo generico e causava problemi con alcune distribuzioni. L'altra ragione importante è che gli sviluppatori pensano che Umbrello sia un nome molto più bello.

Lo sviluppo di Umbrello UML Modeller e le discussioni sugli sviluppi futuri del programma sono aperti e si trovano su Internet. Se vuoi contribuire al progetto, non esitare a contattare gli sviluppatori. Ci sono molti modi in cui puoi aiutare Umbrello UML Modeller:

- Fare rapporti di errori o suggerire miglioramenti
- Correggere errori o aggiungere funzionalità
- Scrivere della buona documentazione o tradurla in altre lingue
- E ovviamente... scrivere codice con noi!

Come vedi, ci sono molti modi in cui puoi contribuire. Tutti sono molto importanti e tutti sono i benvenuti a partecipare.

Gli sviluppatori di Umbrello UML Modeller possono essere raggiunti all'indirizzo umbrello-devel@kde.org.

Capitolo 8

Copyright

Copyright 2001 di Paul Hensgen

Copyright 2002-2020 degli autori di Umbrello UML Modeller.

Questa documentazione è concessa in licenza sotto i termini della GNU Free Documentation License.

Questo programma è concesso in licenza sotto i termini della GNU General Public License.