



**Programma operativo Regione Lombardia/Ministero del
Lavoro/Fondo Sociale Europeo, Obiettivo 3 Misura C3**

**Progetto ID 24063 “Moduli e contenuti professionalizzanti inseriti nei corsi di
laurea e diplomi universitari dell’Università degli Studi di Brescia”**

Azione ID 41654

**“Formazione teorico–pratica nell’ambito di moduli professionalizzanti
orientati all’Informatica del terzo anno di corso
della Laurea in Ingegneria dell’Informazione (nuovo ordinamento)”**

Insegnamento a supporto del quale si svolge l’azione formativa integrativa:

Ingegneria del Software A

Modulo n.2 “ Strumenti CASE e specifiche OMG”



Diagrammi delle classi

- Tecnica centrale di modellazione OO
- Descrizione statica degli oggetti che compongono il sistema (comprensiva di attributi e operazioni) e delle loro relazioni (restrizioni incluse)
- Descrizione dei vincoli
- Sono simili ai modelli per i dati → è facile basare erroneamente il loro sviluppo sui dati piuttosto che sulle responsabilità

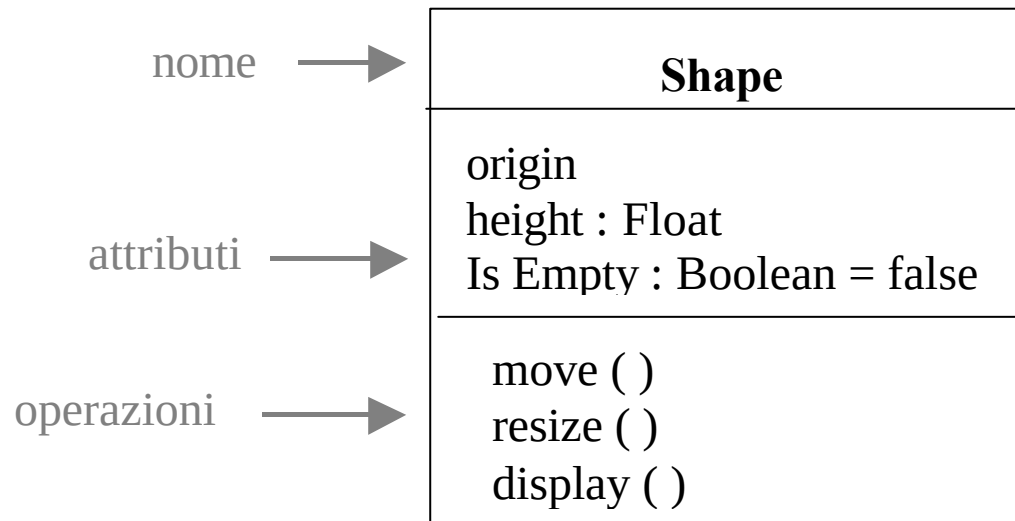
Prospettive nella realizzazione di un (qualsiasi) diagramma

Prospettiva	Confusione fra prospettive
<i>Concettuale</i> : presentazione di concetti del dominio	È possibile e non grave se avviene fra queste due prospettive
<i>Specifica</i> : presentazione dell'interfaccia del sw	
<i>Implementazione</i> : presentazione dell'implementazione	Da evitarsi quando coinvolge questa prospettiva

Le prospettive non appartengono a UML, però il significato di ogni elemento di un diagramma dipende dalla prospettiva adottata

Diagrammi delle classi: concetti essenziali

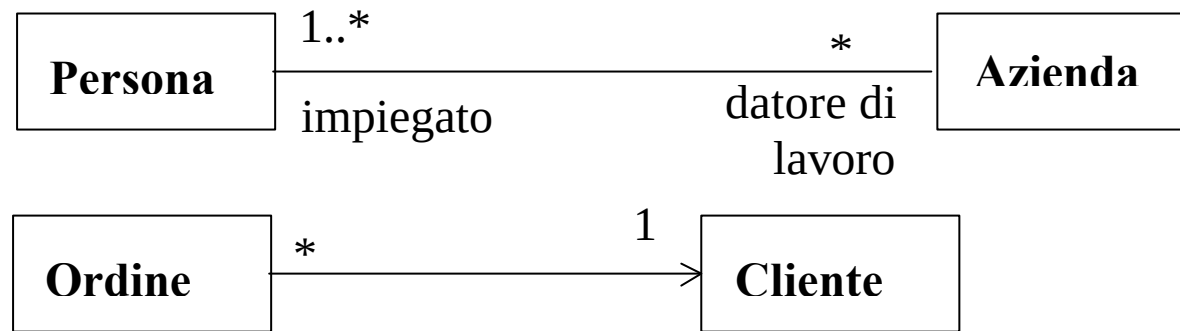
Elementi	Sintassi	Semantica
Classe	Scatola suddivisa in tre parti orizzontali, contenenti rispettivamente <i>nome</i> , <i>attributi</i> (lo stato) e <i>operazioni</i> (il comportamento); <i>nome</i> è una stringa in grassetto che identifica univocamente la classe e può essere prefissata dal nome del package che contiene la classe (es. java::awt::Rectangle)	Insieme di oggetti che condividono gli stessi attributi, operazioni, relazioni e semantica



Diagrammi delle classi: concetti essenziali (cont.)

Elementi	Sintassi	Semantica
Associazione (o ruolo)	Linea continua che unisce due classi; un capo della linea è etichettato con: • <i>molteplicità</i> (es. 1, *, 0..1) • <i>nome del ruolo</i> (stringa opzionale, da specificare solo quando fa aumentare la comprensibilità, altrimenti il nome implicito del ruolo è quello della classe attaccata al capo) e può terminare con una punta biforcuta, che indica la <i>navigabilità</i>	• Relazione statica fra le istanze di due classi; • la <i>molteplicità</i> è il numero di oggetti che prendono parte a tale relazione; • la <i>navigabilità</i> è il verso di percorribilità del collegamento; • l'assenza dell'indicazione di navigabilità si interpreta come mancanza di informazioni circa la stessa

Diagrammi delle classi: concetti essenziali (cont.)



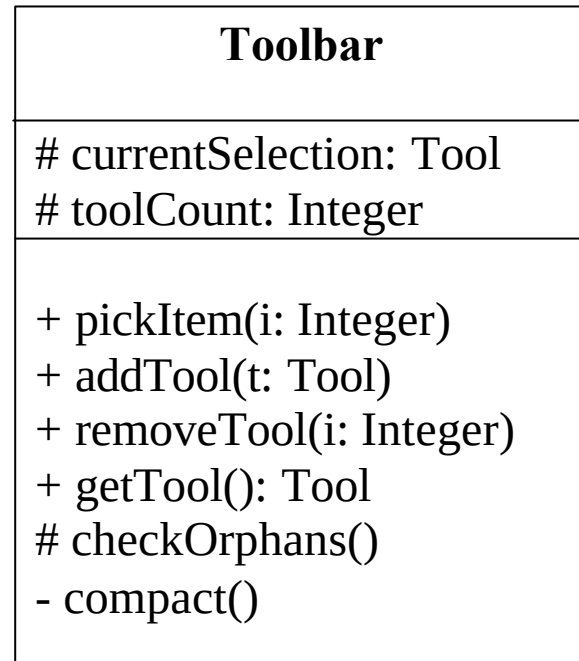
Diagrammi delle classi: concetti essenziali (cont.)

Elementi	Sintassi	Semantica
Attributo	<i>visibilità nome [molteplicità]:</i> <i>tipo = valore-di-default</i> dove solo <i>nome</i> , che è una stringa, è obbligatorio	Proprietà della classe
Visibilità (di attributi e operazioni)	+ (pubblica): è il default # (protected) - (privata)	• Gli elementi pubblici possono essere usati da un'altra classe, quelli privati invece sono riservati all'uso interno • Per la semantica precisa dei tre identificatori bisogna fare riferimento alle regole del linguaggio di programmazione adottato

Diagrammi delle classi: concetti essenziali (cont.)

Elementi	Sintassi	Semantica
Operazione	<i>visibilità nome (lista-parametri): tipi-ritornati {stringa-proprietà}</i> dove • <i>visibilità</i> è opzionale, • <i>nome</i> è una stringa, • <i>lista-parametri</i> è una serie di parametri, ciascuno specificato mediante un elemento direzione <i>nome: tipo = valore-di-default</i> e separato dal successivo mediante una virgola (la direzione può assumere i valori <i>in</i>, <i>out</i>, <i>inout</i>, se manca significa che è <i>in</i>), • <i>stringa-proprietà</i> è opzionale e indica i valori delle proprietà dell'operazione, cioè <i>query</i>, <i>modificatore</i>, <i>get</i> e <i>set</i>	• Processo che una classe sa come effettuare (metodo); • se un'operazione <i>i</i>) non modifica lo stato del sistema e <i>ii</i>) ritorna dei valori è una <i>query</i>, • se un'operazione <i>i</i>) modifica lo stato e <i>ii</i>) non ritorna valori è un <i>modificatore</i>; • un metodo <i>get</i> è una particolare <i>query</i> che restituisce il valore di un attributo e non fa nient'altro; • un metodo <i>set</i> è un particolare <i>modificatore</i> che assegna il valore a esso passato a un attributo e non fa nient'altro

Diagrammi delle classi: concetti essenziali (cont.)

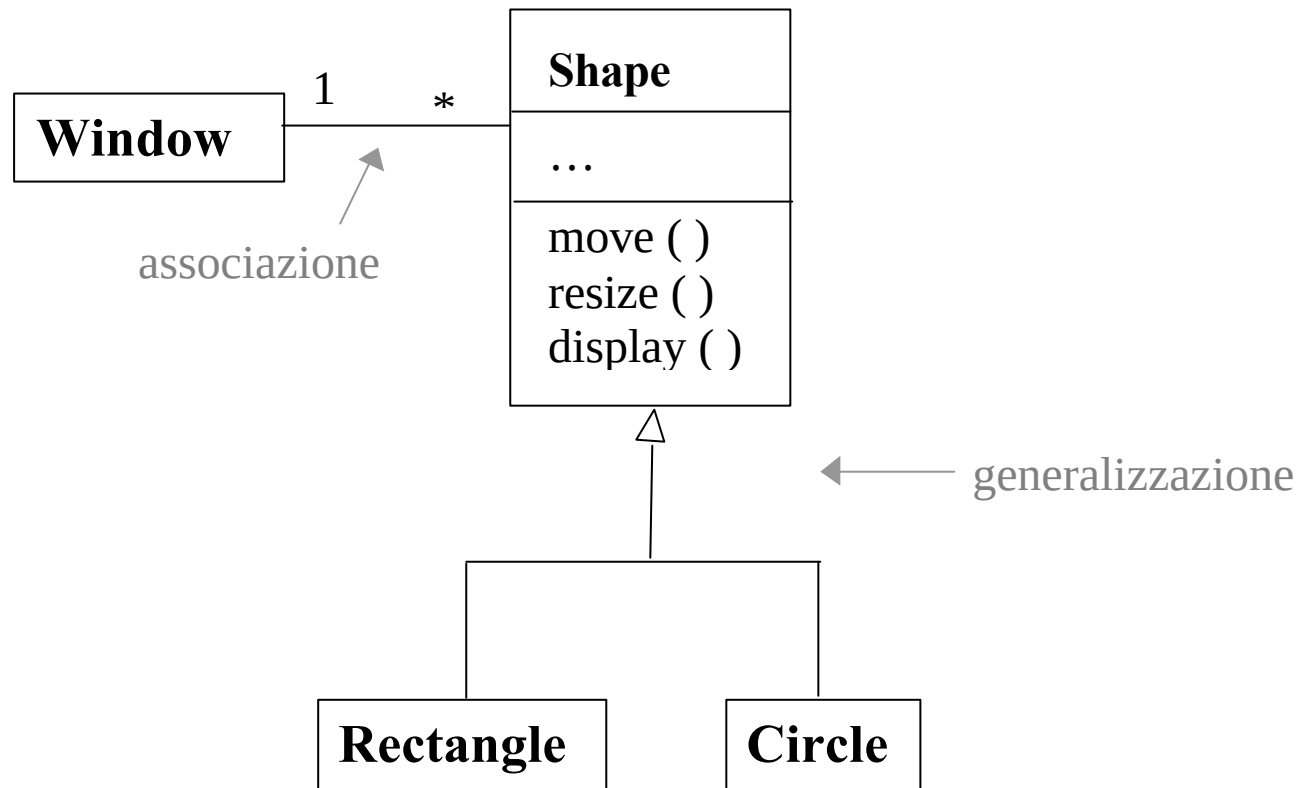


Diagrammi delle classi: concetti essenziali (cont.)

Elementi	Sintassi	Semantica
Generalizzazione	Freccia con testa vuota e uno o più capi: la testa termina nella superclasse, i capi si dipartano ciascuno da una sottoclasse	Se due o più classi sono diverse ma mostrano anche molte somiglianze, queste si possono raccogliere in una superclasse (relazione IS-A)
Vincolo	Descrizione del vincolo in linguaggio naturale racchiusa fra parentesi graffe, oppure descrizione in OCL (Object Constraint Language)	

Responsabilità di una classe = obbligo che la classe si assume nello svolgere un servizio. Fa parte del contratto che essa stipula con le altre classi

Diagrammi delle classi: concetti essenziali (cont.)



Prospettive e diagrammi delle classi

Prospettiva	Interpretazione
<i>Concettuale</i> (da adottarsi per realizzare il modello di dominio nella fase di elaborazione del RUP)	classe = concetto nella mente del cliente (→ il diagramma diventa il linguaggio dell'interlocutore) associazione = relazione fra concetti (la navigabilità non è utile) attributo = proprietà degli oggetti della classe, notazione alternativa rispetto all'associazione operazione = una delle responsabilità principali della classe generalizzazione: qualsiasi cosa si possa dire del supertipo (in termini di associazioni, attributi e operazioni), si può dire anche dei sottotipi

Prospettive e diagrammi delle classi (cont.)

Prospettiva	Interpretazione
<i>Specifica</i>	classe = tipo (un tipo è l'interfaccia – o parte di essa - di una o più classi “fisiche”, una classe “fisica” può implementare più tipi) associazione = responsabilità (indipendenti dalla struttura dati) ovvero, per ogni relazione R fra due classi C1 e C2 con navigabilità C1→C2, esistenza di un'interfaccia (implicita, che, a livello implementativo può essere realizzata in vario modo) che • dato un elemento di C1, determina tutti gli elementi di C2 che partecipano a R • effettua l'aggiornamento della relazione

Prospettive e diagrammi delle classi (cont.)

Prospettiva	Interpretazione
<i>Specifica (cont.)</i>	attributo = responsabilità: esiste un'interfaccia (implicita) che • consente di impostare il valore dell'attributo di un oggetto • restituisce il valore dell'attributo operazione = metodo pubblico generalizzazione = creazione di sottotipi (o ereditarietà di interfacce): l'interfaccia del sottotipo deve includere tutti gli elementi dell'interfaccia del supertipo (si dice che deve essere conforme all'interfaccia del supertipo); deve essere possibile sostituire un'istanza della sottoclasse all'interno di qualsiasi pezzo di codice che preveda l'esistenza di un'istanza della superclasse e tutto deve continuare a funzionare N.B. la creazione di sottoclassi “fisiche” è solo uno dei modi per implementare la creazione di sottotipi, un altro è la delega

Prospettive e diagrammi delle classi (cont.)

Prospettiva	Interpretazione
<i>Implementazione</i>	classe = classe “fisica” associazione $R (C1 \rightarrow C2)$ = dato un qualsiasi oggetto di C1 esistono specifici meccanismi di collegamento verso gli oggetti corrispondenti di C2 la navigabilità può essere diversa da quella di specifica attributo = campo generalizzazione: la sottoclasse eredita tutti i metodi e i campi della superclasse e può ridefinirne le funzioni (eredità di implementazione)

Diagramma delle classi: benefici

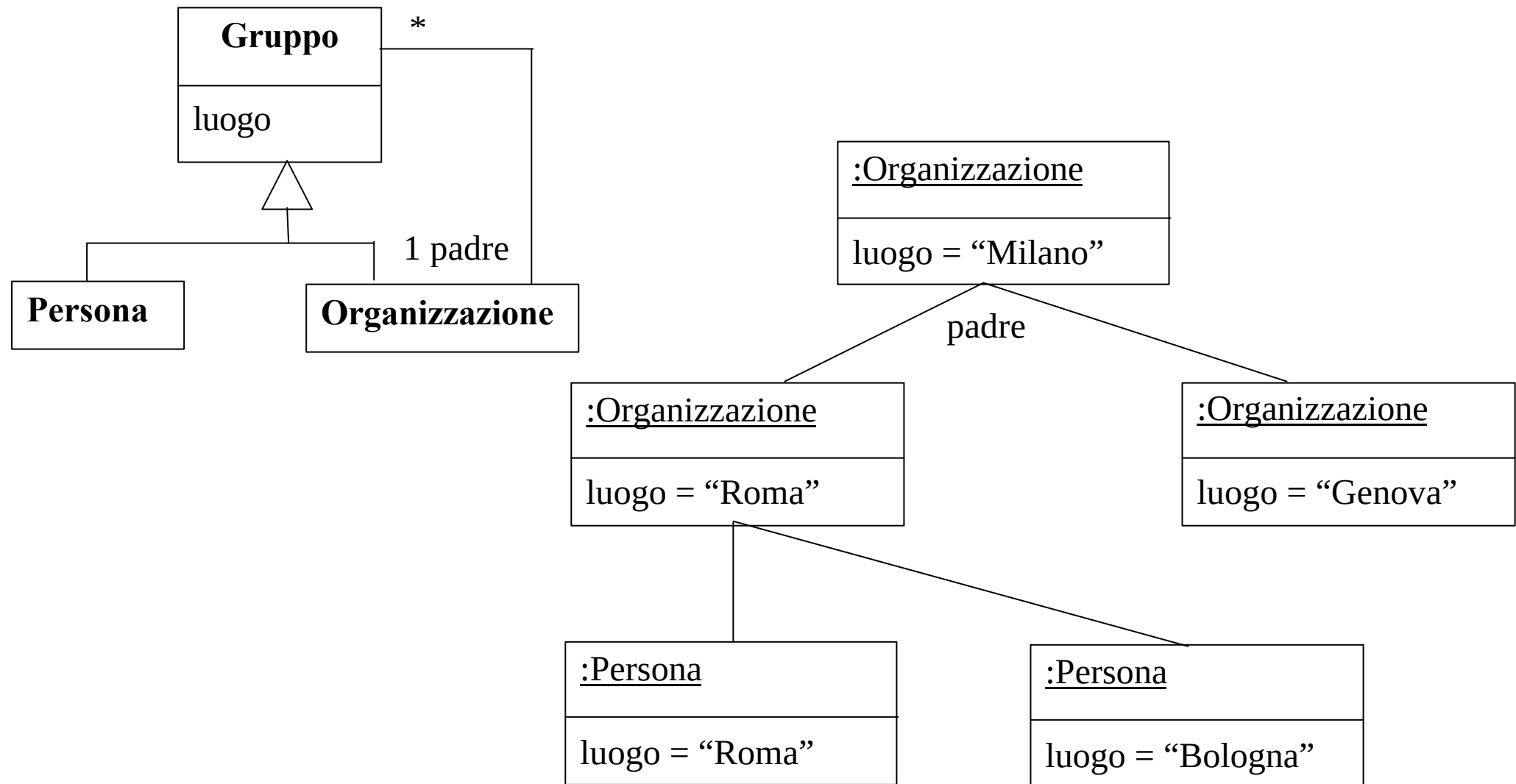
- Cattura il vocabolario del sistema che si sta sviluppando
- Il suo sviluppo deve essere basato sulle responsabilità
- Mediante l'organizzazione in classi consente di distribuire le responsabilità in maniera bilanciata, che è la chiave per un buon progetto
- Consente di comprendere le astrazioni presenti nel sistema

Diagramma degli oggetti (o delle istanze)

- È la fotografia degli oggetti che compongono un sistema sw in un dato momento
- È utile quando le connessioni fra oggetti sono complicate

Elementi	Sintassi	Semantica
Oggetto	Scatola suddivisa in due parti, entrambe dal contenuto opzionale, la prima contenente <i>nome-istanza :nome-classe</i> , la seconda zero, uno o più <i>nome-attributo = “valore”</i>	Istanza di una classe
Collegamento	Linea continua fra due oggetti, eventualmente etichettata con <i>nome-associazione</i>	Istanza di relazione fra due classi

Diagramma degli oggetti (cont.)



Stereotipi

Concetto UML	Uso
Stereotipo	Meccanismo di estensione di UML: rappresenta un costrutto non compreso in UML ma simile a un altro che vi è compreso

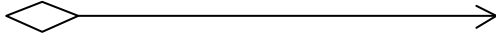
Diagrammi delle classi: concetti avanzati (opzionali)

Elementi	Sintassi	Semantica
Stereotipo di • classe • associazione • generalizzazione	<< <i>nome-stereotipo</i> >> Es. <<interfaccia>>	Uno stereotipo è un sottotipo di un tipo definito nel metamodello UML
Operazione e attributo con campo d'azione di classe	Come operazione e attributo d'istanza ma sottolineato	Metodi e attributi statici

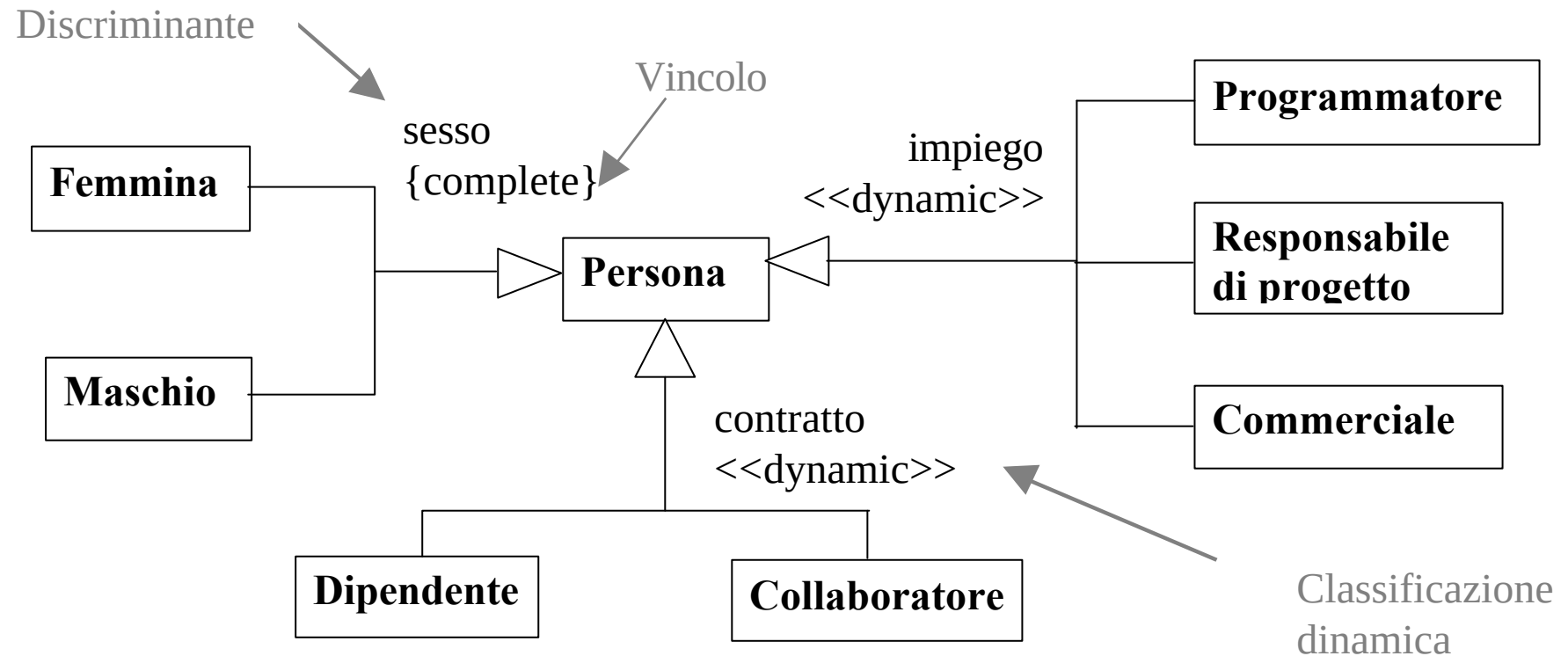
Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Classificazione multipla (subtyping)	• Come la generalizzazione, eventualmente etichettata con un <i>discriminante</i>, che indica la base della classificazione; • il discriminante è accompagnato dal <i>vincolo</i> {complete} se ogni istanza della superclasse deve essere anche un'istanza di uno dei sottotipi del gruppo così contraddistinto (superclasse astratta); • ogni combinazione fra sottotipi aventi discriminante {complete} distinto deve essere legale; • tutti i sottotipi aventi lo stesso discriminante (gruppo di sottotipi) devono essere disgiunti	• Associa a una classe (la superclasse) più sottotipi ortogonali (le sottoclassi) • utile a livello di modellazione concettuale

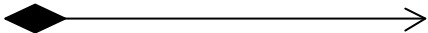
Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Classificazione dinamica	Come la classificazione multipla ma il discriminante è accompagnato dallo stereotipo <<dynamic>>	• Consente agli oggetti di cambiare tipo all'interno di una struttura di sottotipi • utile a livello di modellazione concettuale
Aggregazione	Freccia che termina sulla classe “parte” con una punta biforcuta; all'altra estremità si diparte dalla classe “tutto” con un rombo vuoto; su entrambe le estremità è obbligatoria la molteplicità 	Generica relazione “parte-di” (o HAS-A)

Diagrammi delle classi: concetti avanzati (cont.)



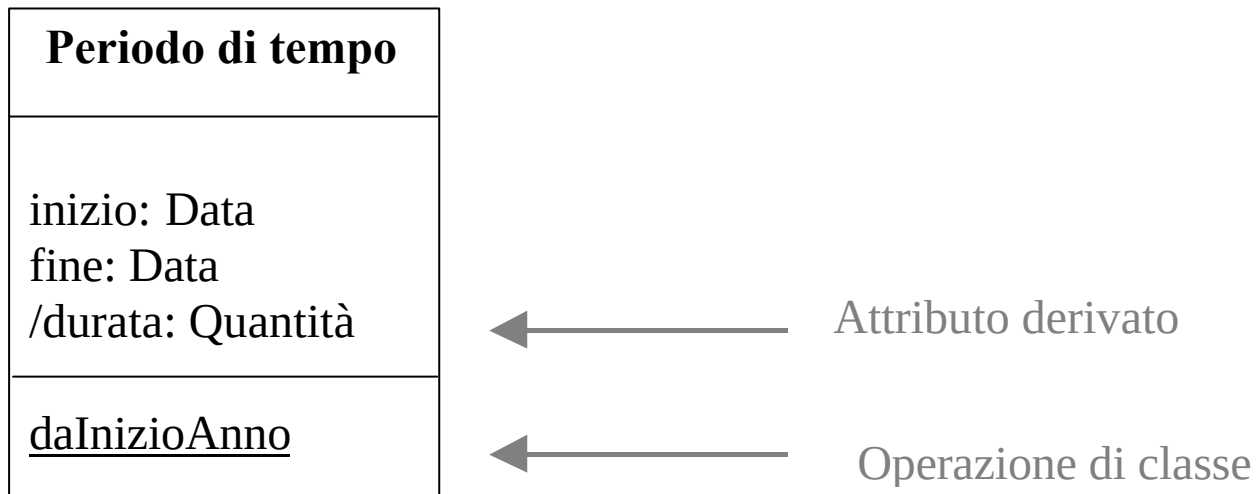
Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Composizione	Come nell'aggregazione ma il rombo è pieno 	• Varietà più forte dell'aggregazione: l'oggetto componente può appartenere a un solo oggetto composto • la cancellazione dell'oggetto composto si propaga a tutti gli oggetti componenti

Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Associazione e attributo derivati	Il nome dell'associazione o dell'attributo è preceduto da una barra (/)	• È un attributo/associazione rappresentato esplicitamente anche se può essere derivato a partire da altri attributi/associazioni • dal punto di vista della specifica, indica un vincolo tra valori (non cosa viene direttamente memorizzato e cosa viene derivato) • dal punto di vista implementativo indica i campi usati come cache per motivi di efficienza • dal punto di vista concettuale, può essere usato per ricordarsi di chiedere conferma agli esperti circa una presunta derivabilità

Diagrammi delle classi: concetti avanzati (cont.)



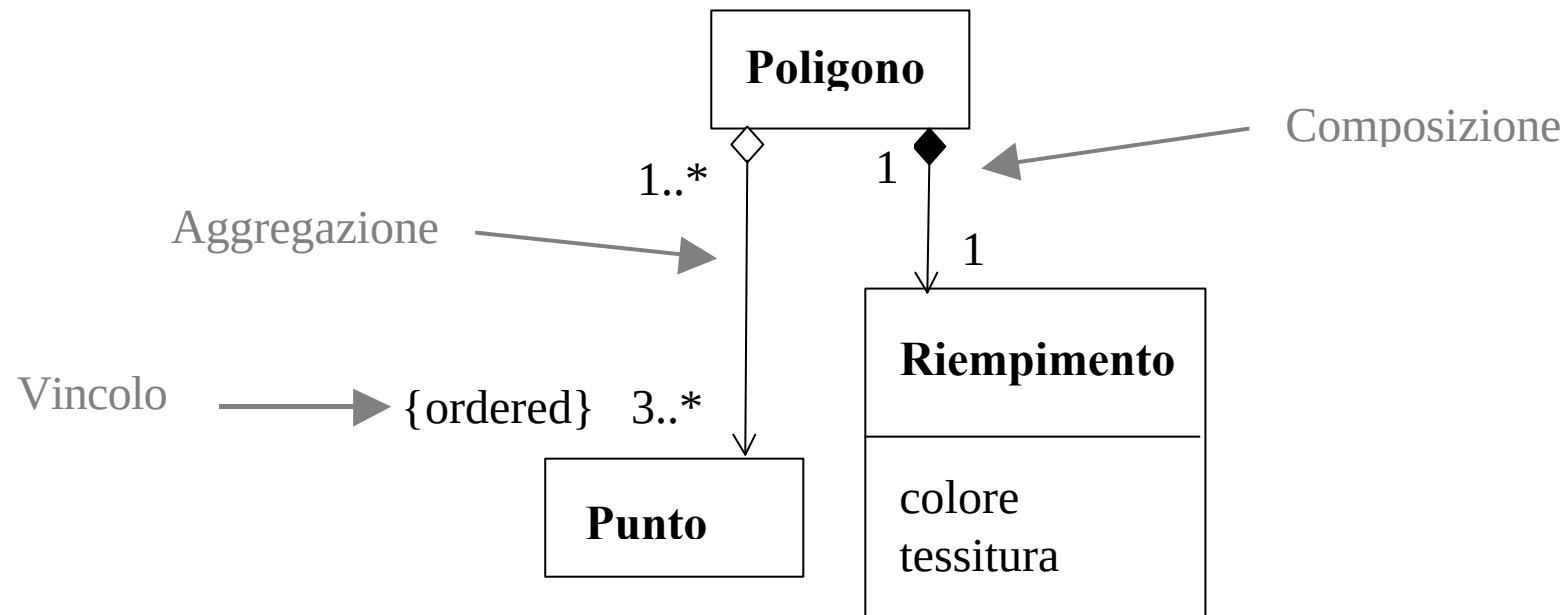
Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Classe e metodo astratti	Nome in corsivo + eventuale <i>vincolo</i> {abstract}	
Interfaccia	Uso dello stereotipo <<interface>> entro la scatola della classe, dove il nome è in corsivo	
Realizzazione	Freccia con linea tratteggiata e punta vuota diretta dalla classe all'interfaccia	• Relazione fra una classe che implementa un'interfaccia e l'interfaccia stessa • a livello di specifica, realizzazione e subtyping sono indistinguibili

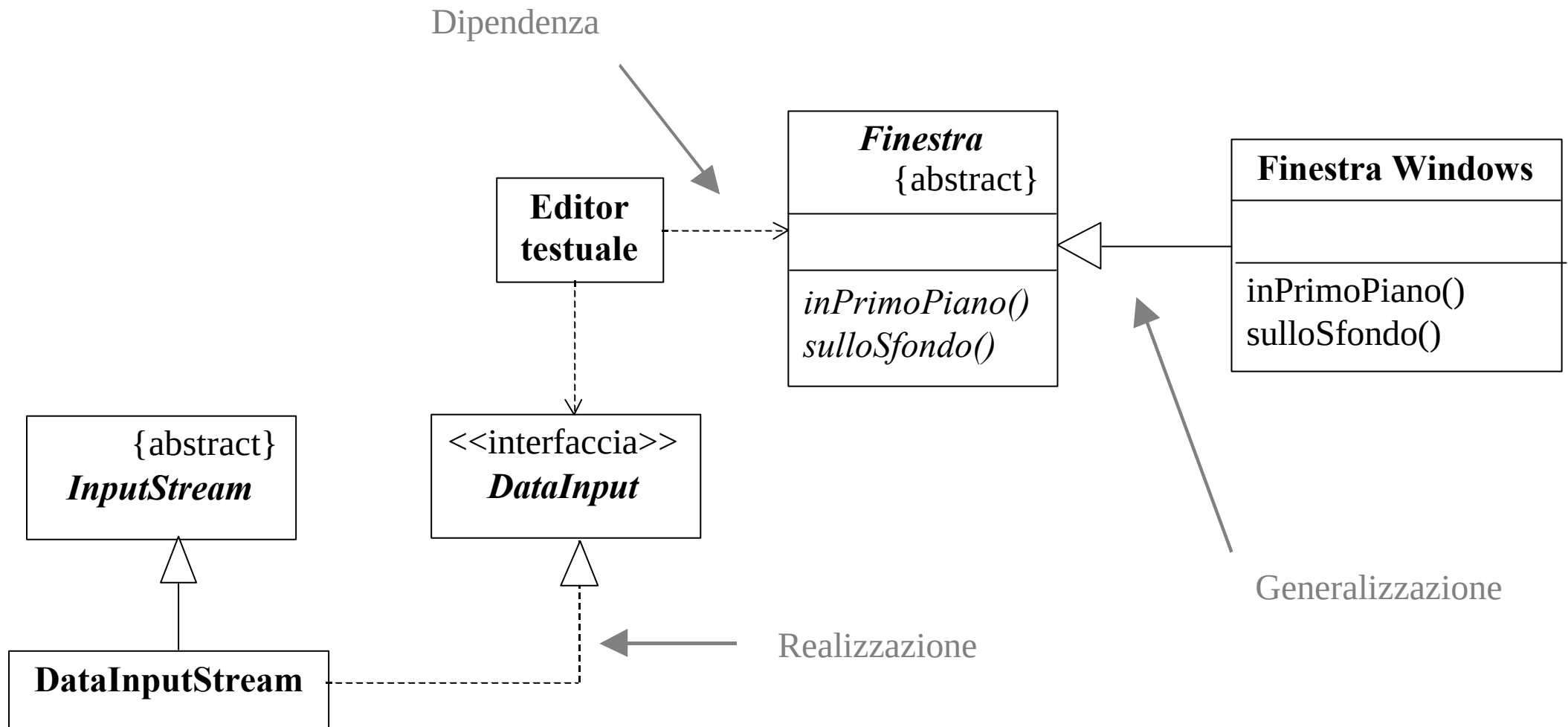
Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Dipendenza	Freccia con linea tratteggiata e punta biforcuta diretta dalla classe all'interfaccia (o alla classe astratta)	Relazione fra una classe e un'interfaccia (o una classe astratta), dove se l'interfaccia (o la classe astratta) dovesse cambiare, anche la classe potrebbe cambiare
Vincoli circa i capi di associazioni a più valori	<code>{ordered}</code> <code>{bag}</code> <code>{hierarchy}</code> <code>{dag}</code>	• <code>{ordered}</code>: esiste un ordinamento fra gli oggetti destinazione dell'associazione • <code>{bag}</code>: gli oggetti destinazione possono comparire più volte • <code>{hierarchy}</code>: gli oggetti destinazione formano una gerarchia • <code>{dag}</code>: gli oggetti destinazione formano un DAG

Diagrammi delle classi: concetti avanzati (cont.)



Diagrammi delle classi: concetti avanzati (cont.)



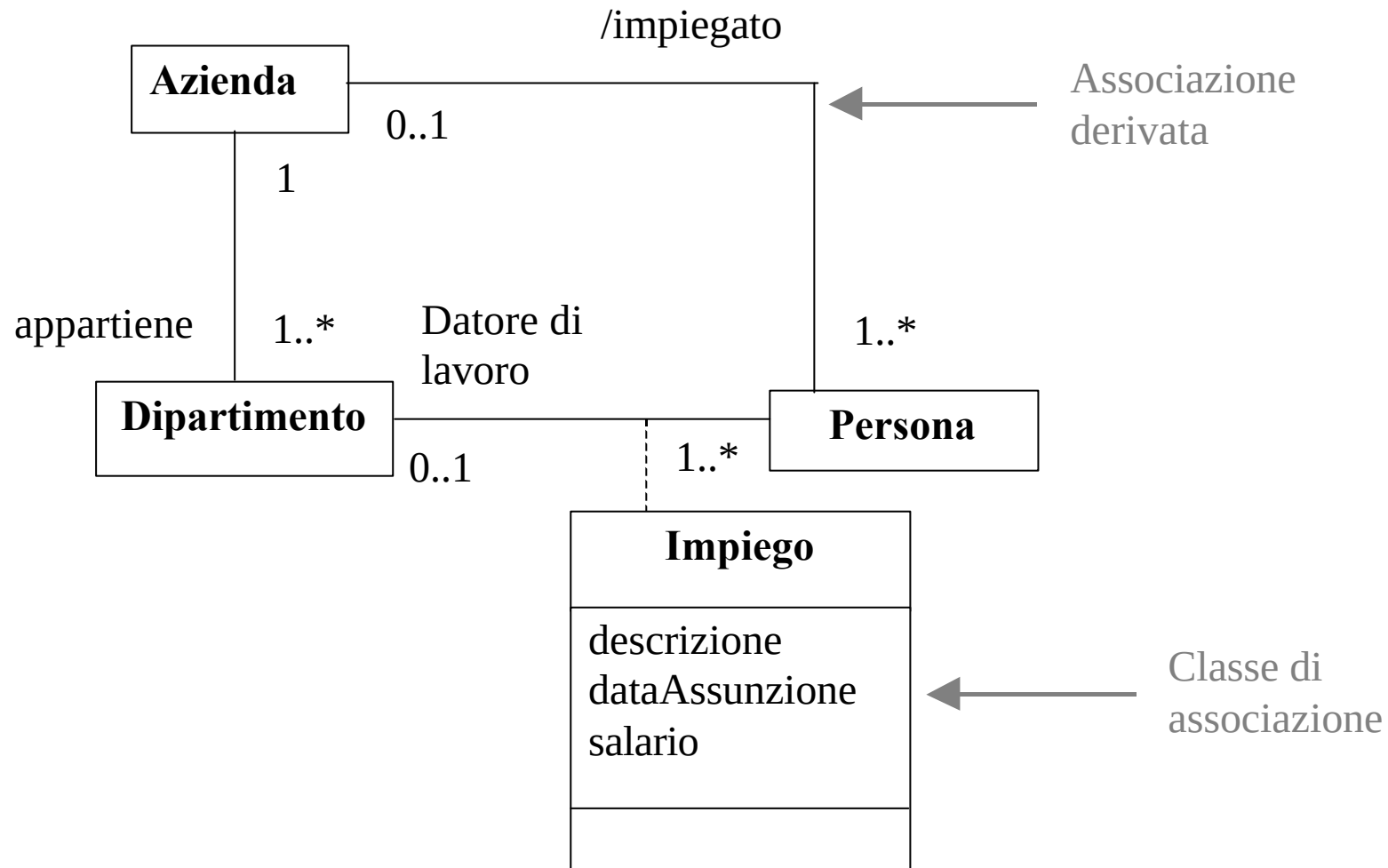
Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Vincolo frozen	{frozen} scritto a fianco di un attributo, o del nome di una classe, o a un capo di un'associazione	• Il valore dell'attributo o del capo dell'associazione, inizializzato al momento della creazione dell'oggetto, non può cambiare per tutta la vita dell'oggetto • Se applicato a una classe, indica che tutti i capi di associazione collegati alla classe sono frozen • Dal punto di vista implementativo il valore dell'attributo frozen viene passato al costruttore dell'oggetto e non esistono operazioni per aggiornarlo

Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Classe di associazione	Classe collegata a una linea di associazione mediante una linea tratteggiata	Aggiunge un vincolo extra: ci può essere solo un'istanza della classe di associazione tra ogni coppia di oggetti associati
Classe parametrica (o template)	Classe con un riquadro tratteggiato sull'angolo superiore destro, contenente i parametri (tipi) della classe, che possono essere più d'uno	• Usata per lo più per definire collezioni • Raramente utile in fase di modellazione concettuale; da usare in fase di specifica e implementazione solo se effettivamente supportata dal linguaggio di programmazione

Diagrammi delle classi: concetti avanzati (cont.)



Diagrammi delle classi: concetti avanzati (cont.)

Elementi	Sintassi	Semantica
Stereotipo <<bind>>	Stereotipo sulla relazione di raffinamento che congiunge la classe elemento legato con la classe template, seguito dal nome del tipo usato come parametro fra parentesi angolari (<>)	L'elemento legato non è un sottotipo: infatti non è lecito aggiungergli caratteristiche rispetto a quelle del template

Diagrammi delle classi: concetti avanzati (cont.)

