Diagrammi dei package

All'aumentare delle dimensioni dei sistemi sw, diviene difficile comprendere e modificare gli stessi

Package = meccanismo di raggruppamento di più classi (o, in teoria, anche di altri elementi del modello) in unità di livello più alto, al fine di dominare questa complessità

Dipendenza

- Fra due elementi esiste una (relazione di) dipendenza se i cambiamenti apportati alla definizione dell'uno si possono potenzialmente ripercuotere sull'altro
- Idealmente solo i cambiamenti dell'interfaccia di una classe dovrebbero ripercuotersi sulle altre
- Il principio del progetto per il cambiamento richiede di minimizzare le dipendenze, cosicché gli effetti dei cambiamenti siano ridotti e il sistema si possa modificare con minore sforzo
- UML prevede molti tipi di dipendenza, ognuno con la propria semantica e stereotipo (ad es. <<import>> e <<access>>)

Dipendenza fra classi e fra package

- La dipendenza è il metodo euristico maggiormente usato in UML per raggruppare le classi
- Ogni package può raggruppare classi pubbliche e/o private e/o protette (*protected)*
- Fra due package esiste dipendenza se c'è dipendenza fra una classe del primo e un'altra (necessariamente pubblica) del secondo
- Le dipendenze fra package non sono transitive

Interfaccia di un package

- È l'insieme dei metodi pubblici delle classi pubbliche del package
- Per ridurre tale insieme si può dare a tutte le classi del package visibilità privata, aggiungendo poi altre classi pubbliche, denominate facciate (mediante lo stereotipo <<facade>>), per implementare il comportamento esterno
- Le facciate delegano l'esecuzione delle proprie operazioni alle classi nascoste

Diagramma dei package

- Mostra più package di classi con le loro dipendenze
- Aiuta a individuare le dipendenze (al fine di ridurle)
- È uno strumento fondamentale per mantenere il controllo sulla struttura globale del sistema sw

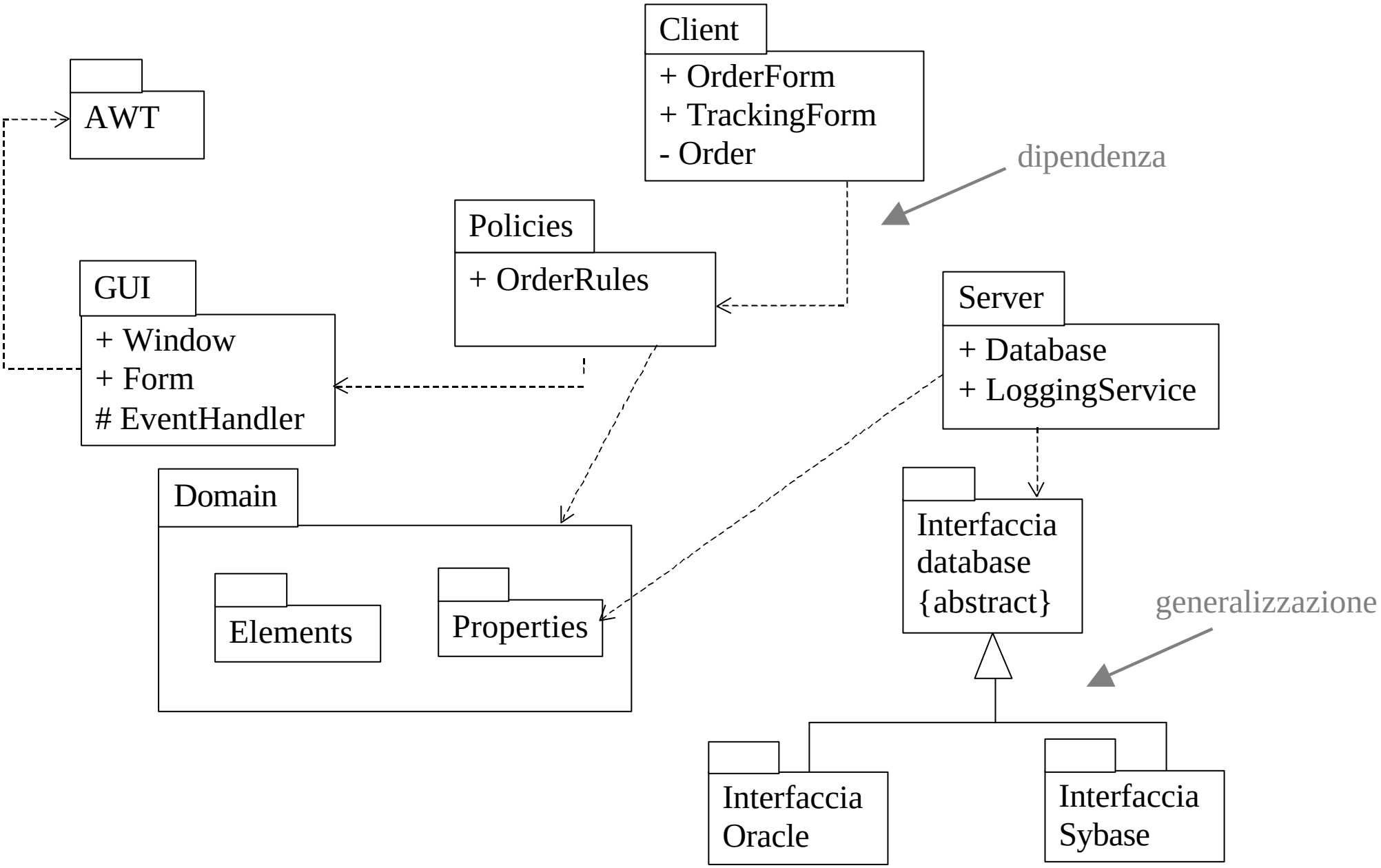
Elementi	Sintassi	Semantica
Package	Scatola con etichetta. L'etichetta può: ▪ Essere vuota, nel qual caso il nome del package è contenuto nella scatola ▪ Contenere il nome del package, nel qual caso la scatola contiene un elenco di classi, oppure un diagramma dei package o delle classi ▪ Contenere lo stereotipo <<global>>, che significa che tutti i package del sistema sono dipendenti dal package considerato	L'aggregazione di più package entro un package consente la modellazione gerarchica di sistemi complessi

Diagramma dei package (cont.)

Elementi	Sintassi	Semantica
Dipendenza	Freccia con linea tratteggiata e punta biforcuta, uscente dal dipendente	Una dipendenza verso un package che a sua volta ne contiene altri è la somma di più dipendenze di livello inferiore, cioè denota che ci sono dipendenze verso qualche elemento interno
Generalizzazione	Freccia, con linea continua e punta triangolare vuota, uscente dal package più specifico	Il package più generale, marcato come {abstract} (perché contiene classi astratte), serve solo per definire un'interfaccia che sarà implementata dal package più specifico

Suggerimenti

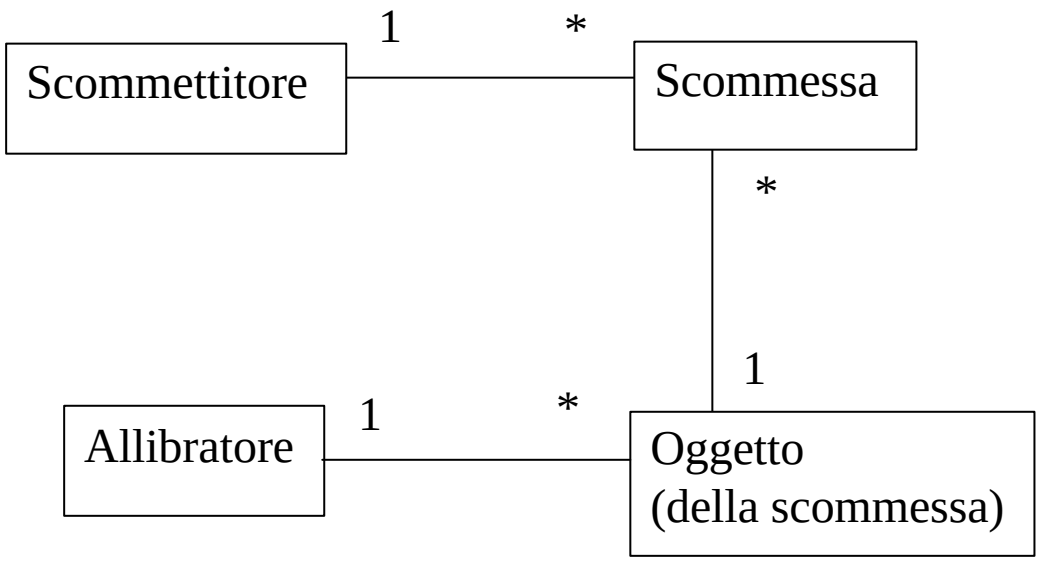
- Minimizzare i cicli nella struttura delle dipendenze
- Se ci sono dei cicli, cercare di contenerli in un package più grande
- Sforzarsi di eliminare cicli nelle dipendenze tra i package del dominio e le interfacce esterne
- In un sistema già implementato, le dipendenze fra package possono essere dedotte da uno strumento automatico analizzando le classi: ciò è utile per migliorare la struttura del sistema attraverso una riduzione delle dipendenze (refactoring)
- Considerare il package come unità di base per il testing (può essere utile aggiungere delle classi che servono solo in supporto al testing delle altre classi nel package)
- Regola empirica: se la stampa del diagramma delle classi dell'intero sistema su un A4 non è leggibile, è meglio scomporlo in package


Client
+ OrderForm
+ TrackingForm
- Order
AWT
dipendenza
Policies
+ OrderRules
GUI
+ Window
+ Form
# EventHandler
Server
+ Database
+ LoggingService
Domain
Elements
Properties
Interfaccia
database
{abstract}
generalizzazione
Interfaccia
Oracle
Interfaccia
Sybase

Collaborazione

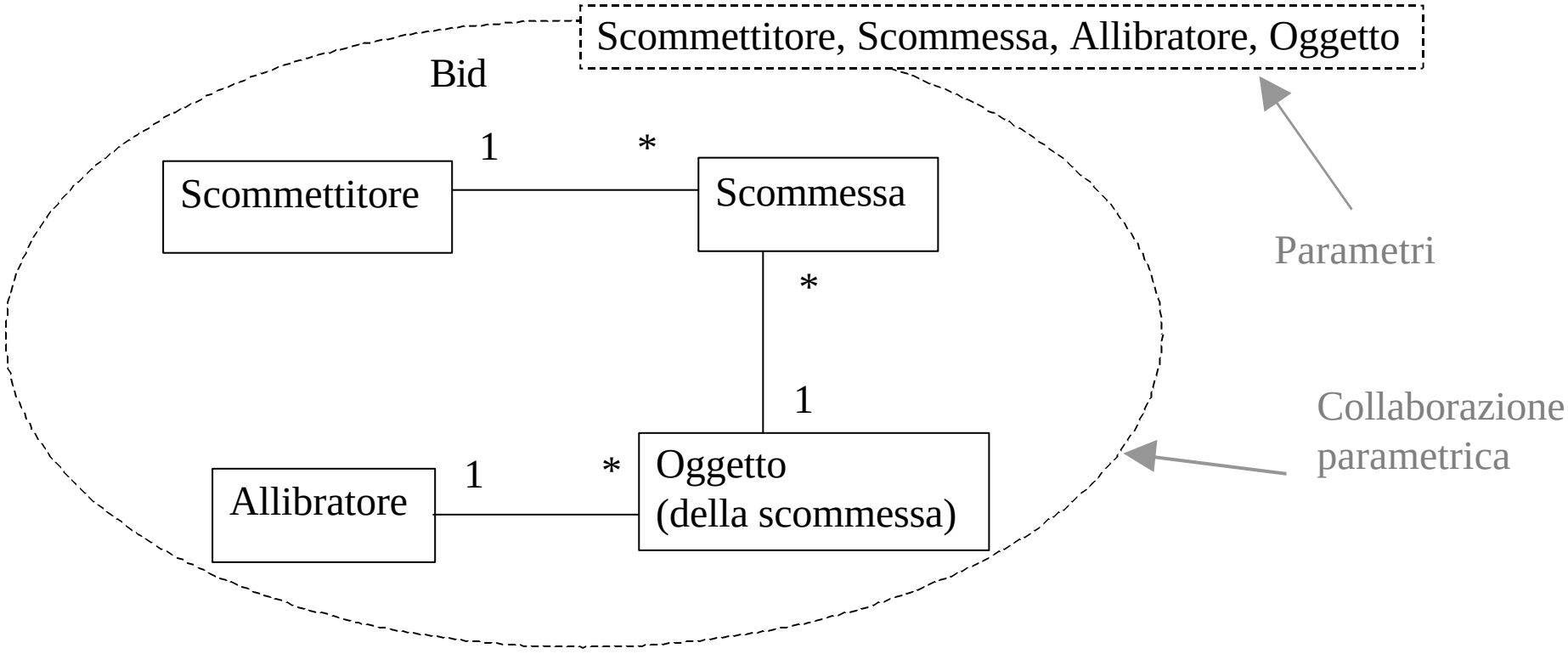
Interazione tra due o più classi, passibile di una duplice rappresentazione:

- il suo comportamento è tipicamente descritto da un diagramma delle interazioni
- un diagrammi delle classi mostra invece le classi che interagiscono



Collaborazione parametrica

- Collaborazione in cui le classi non sono quelle effettive del sistema ma semplici ruoli nella collaborazione stessa
- Serve quando una collaborazione è molto simile per più classi dello stesso sistema ma i nomi delle classi cambiano



Relazione fra package e collaborazioni

- I package entro i diagrammi dei package possono contenere diagrammi delle classi per mostrare le classi che partecipano alle collaborazioni (parametriche e non)
- Una collaborazione parametrica può essere usata per mostrare comportamenti comuni fra package