

# Diagrammi di interazione

- Diagrammi di sequenza
- Diagrammi di comunicazione (ex collaborazione)
- Diagrammi di interazione generale
- Diagrammi di temporizzazione

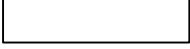


Descrivono la collaborazione di un gruppo di oggetti per implementare collettivamente un comportamento

# Diagrammi di sequenza

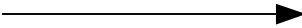
Ogni diagramma

- illustra tipicamente il comportamento dinamico del sistema in corrispondenza (di un singolo scenario) di un singolo caso d'uso
- descrive in un'unica vista il modo in cui gruppi di oggetti possono collaborare, scambiandosi messaggi (cioè invocando operazioni), nello svolgimento del caso d'uso
- enfatizza l'ordinamento temporale delle chiamate di metodi
- rende esplicita la struttura dei messaggi → evidenzia l'eventuale eccessiva centralizzazione di progetti dove un solo oggetto compie tutto il lavoro
- non è adeguato per la rappresentazione della logica di controllo (mostra dei limiti quando si devono rappresentare processi con molti cicli o condizioni)
- non definisce con precisione il comportamento degli oggetti, cioè non spiega i dettagli degli algoritmi
- presenta limitazioni nel rappresentare il comportamento polimorfo

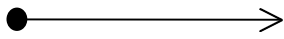
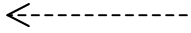
## Diagramma di sequenza

Elementi	Sintassi	Semantica
Partecipante (termine che però non fa parte di UML)	Scatola, posta in cima a una linea di vita, contenente <i>nome : classe</i> dove sia <i>nome</i> , sia <i>: classe</i> sono opzionali ma non possono mancare entrambi 	Solitamente (ma non necessariamente) indica un'istanza di una classe. Di seguito <u>assumiamo si tratti sempre di un oggetto</u>
Linea di vita	Linea tratteggiata verticale Il verso del tempo è dall'alto verso il basso 	Vita dell'oggetto (formalismo introdotto da Jacobson)
Barra di attivazione	Elemento opzionale: sottile scatola verticale sovrapposta alla linea di vita di un oggetto 	Intervallo in cui l'oggetto è attivo nell'interazione (ovvero un suo metodo è in esecuzione)


## Diagramma di sequenza (cont.)

Elementi	Sintassi	Semantica
Messaggio di invocazione	<p>Freccia con punta piena e linea continua, posta tra due linee di vita (non necessariamente distinte) ed etichettata col nome del messaggio</p>  <p>Il nome del messaggio può essere seguito da un elenco di parametri fra parentesi tonde, dove ogni parametro può essere accompagnato dal suo tipo (introdotto da :) ed è separato dal successivo mediante una virgola</p> <p>L'ordine temporale di scambio dei messaggi è dall'alto verso il basso</p>	<ul style="list-style-type: none"><li>• Invocazione da parte dell'oggetto dalla cui linea di vita si diparte la freccia di un metodo dell'oggetto verso la cui linea di vita la freccia è diretta</li><li>• Se le linee di vita sorgente e destinazione coincidono, si tratta di una <i>chiamata interna</i></li></ul>

## Diagramma di sequenza (cont.)

Elementi	Sintassi	Semantica
Messaggio trovato ( <i>found message</i> )	Come il messaggio di invocazione ma l'origine della freccia è una pallina nera piena 	Messaggio che proviene da una fonte esterna e innesca l'interazione visualizzata
Messaggio di ritorno	Elemento opzionale: freccia con punta aperta e linea tratteggiata, posta tra due linee di vita distinte: la linea destinazione aveva precedentemente inviato un msg alla linea sorgente 	Indica il valore ritornato da un'invocazione  L'uso è consigliato solo se aggiunge effettivamente info

## Diagramma di sequenza (cont.)

<b>Elementi</b>	<b>Sintassi</b>	<b>Semantica</b>
Messaggio asincrono	Freccia in linea continua a punta aperta 	Messaggio che non blocca l'esecuzione del chiamante, il quale può proseguire con la sua elaborazione  Pratica comune nelle applicazioni multithread, dove serve, ad es. per creare un thread o per comunicare con un thread già in esecuzione

## Diagramma di sequenza (cont.)

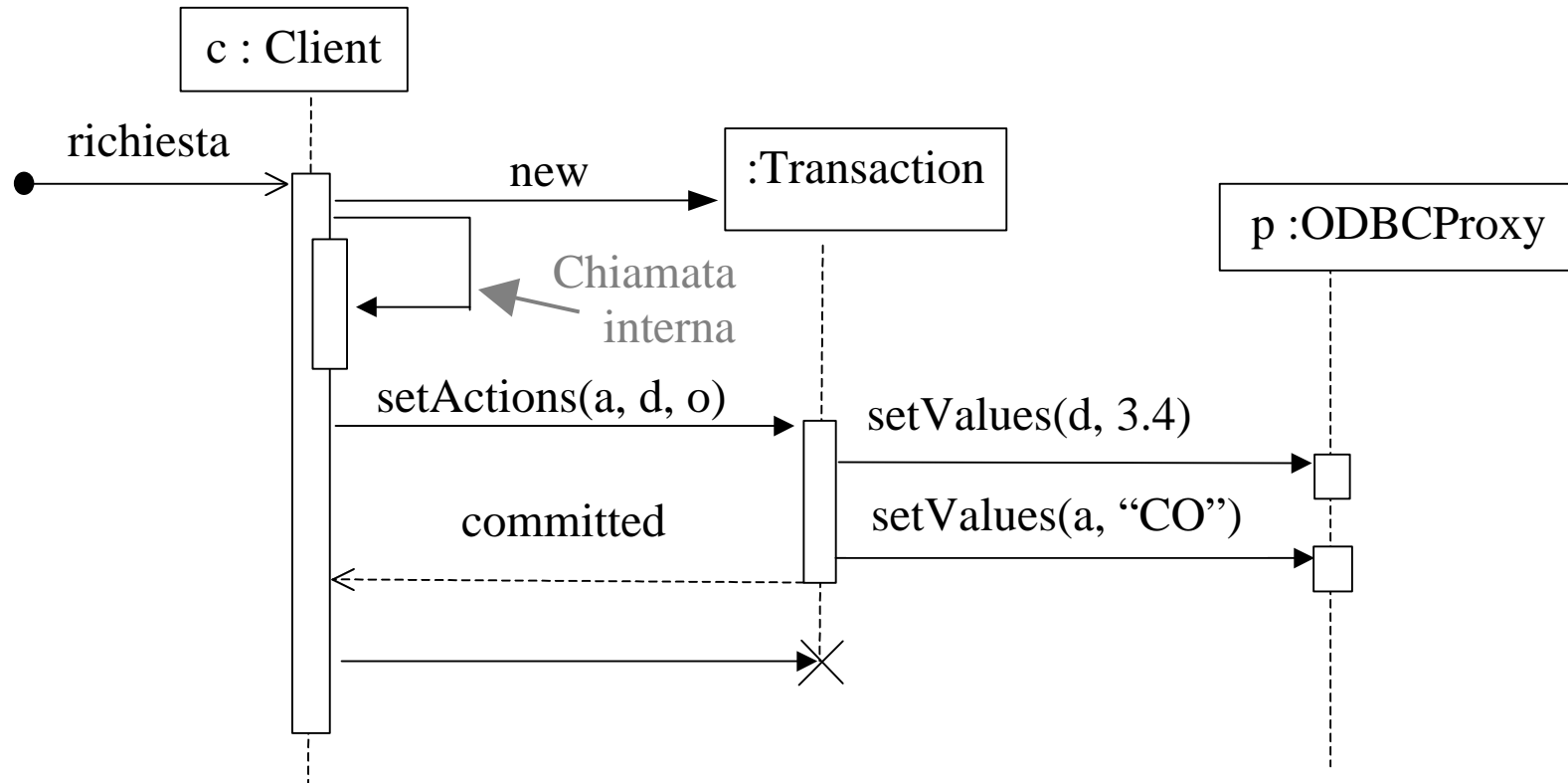
Elementi	Sintassi	Semantica
Messaggio di creazione di un oggetto	È come un messaggio di invocazione ma termina contro la scatola che rappresenta l'oggetto Se il nuovo oggetto fa qualcosa immediatamente dopo la creazione, la sua barra di attivazione può essere direttamente attaccata alla scatola dell'oggetto stesso	Creazione di un'istanza di una classe  Si può etichettare la freccia con la stringa <i>new</i> (che però non fa parte di UML)

## Diagramma di sequenza (cont.)

<b>Elementi</b>	<b>Sintassi</b>	<b>Semantica</b>
Messaggio di distruzione di un oggetto	È come un messaggio di invocazione ma termina in una grande X posta sotto la barra di attivazione dell'oggetto che viene distrutto	Distruzione dell'istanza di un oggetto In un ambiente dotato di garbage collection gli oggetti non sono distrutti esplicitamente ma è bene usare la X per indicare quando un oggetto non serve più ed è pronto per essere cancellato automaticamente
Autodistruzione di un oggetto	Grande X posta sotto la barra di attivazione dell'oggetto	



# Esempio



## Diagramma di sequenza (cont.)

<b>Elementi</b>	<b>Sintassi</b>	<b>Semantica</b>
Frame di interazione	Riquadro, dotato di operatore, che seleziona uno o più frammenti del diagramma, ciascuno dei quali può essere dotato di guardia L'operatore è scritto nell'angolo superiore sx del frame	Novità di UML 2 Serve per rappresentare la logica di controllo
Guardia	Espressione condizionale contenuta fra parentesi quadre	Il messaggio associato viene inviato solamente se la condizione è verificata

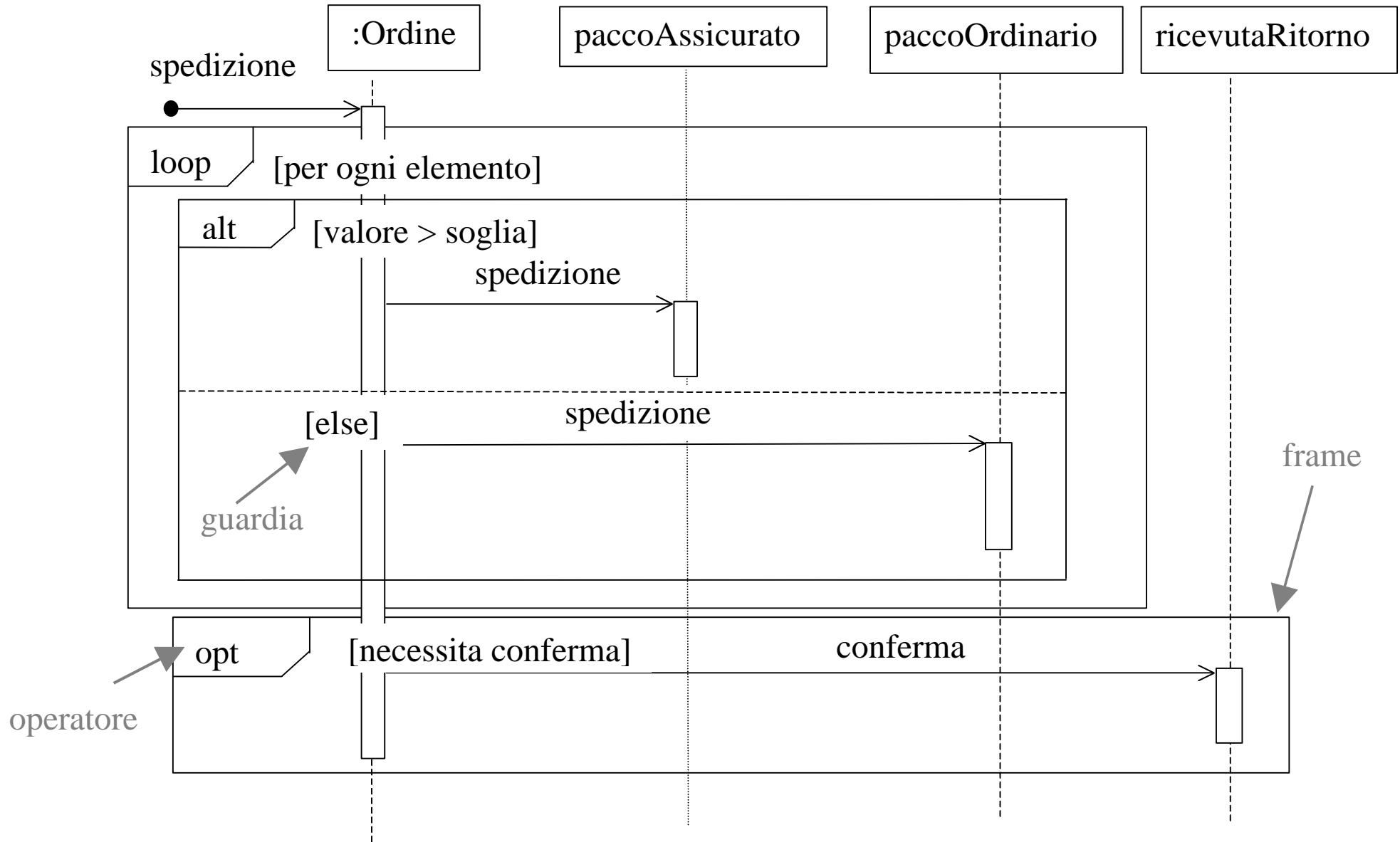
## Alcuni operatori

<b>Operatore</b>	<b>Sintassi</b>	<b>Semantica</b>
<code>alt</code>	Il frame evidenzia più frammenti contenuti in <i>tracce</i> , cioè reciprocamente separati mediante una linea orizzontale tratteggiata, ciascuno dotato di guardia che esprime una condizione mutuamente esclusiva rispetto a quella degli altri frammenti	Rappresenta una alternativa: viene eseguito solo il frammento la cui condizione è vera
<code>opt</code>	Il frame evidenzia un singolo frammento dotato di guardia (equivale ad <code>alt</code> con una sola traccia)	Il frammento viene eseguito solo se la condizione è vera
<code>par</code>	Il frame evidenzia più frammenti contenuti in tracce	Ogni frammento è eseguito in parallelo

## Alcuni operatori (cont.)

<b>Operatore</b>	<b>Sintassi</b>	<b>Semantica</b>
region	Il frame evidenzia un singolo frammento privo di guardia	Rappresenta una regione critica: il frammento è eseguibile da un solo thread per volta
loop	Il frame evidenzia un singolo frammento dotato di guardia	Il frammento può essere eseguito più volte; la base dell'iterazione è indicata dalla guardia
neg	Il frame evidenzia un singolo frammento privo di guardia	Il frammento mostra una interazione non valida
ref	Il frame <ul style="list-style-type: none"><li>• ricopre le linee di vita coinvolte nell'interazione a cui si riferisce,</li><li>• indica il nome di tale interazione ed eventuali parametri,</li><li>• può avere un valore di ritorno</li></ul>	Il frame si riferisce a un'interazione descritta da un altro diagramma

# Esempio con operatori



## Schede CRC (Classe-Responsabilità-Collaborazione)

- Non appartengono a UML
- Si usano per esplorare alternative di comportamento e di interazione delle classi con cui si intende implementare un caso d'uso (o i suoi scenari), prima di effettuare una scelta che sarà poi documentata coi diagrammi di interazione (usare questi ultimi per tale esplorazione sarebbe troppo pesante)

Scheda = cartoncino 10 x 15 cm

Nome_Classe	
Responsabilità	Collaborazione

## Responsabilità di una classe

- È la descrizione di alto livello degli scopi (da uno a tre) per cui è stata creata la classe e dei doveri della stessa nei confronti delle altre e del sistema
- Permette di superare la visione della classe come semplice contenitore di dati
- Può corrispondere a un'operazione, a un attributo, o a un'aggregazione di attributi e operazioni

Collaboratori di una classe = altre classi con cui la classe interagisce