

MODELLAZIONE CON UML: *PACKAGES*

Prof. Mariacarla Staffa

ORGANIZZAZIONE DEI DIAGRAMMI



I modelli di sistemi complessi diventano velocemente anch'essi complessi quando gli sviluppatori li revisionano e rifiniscono

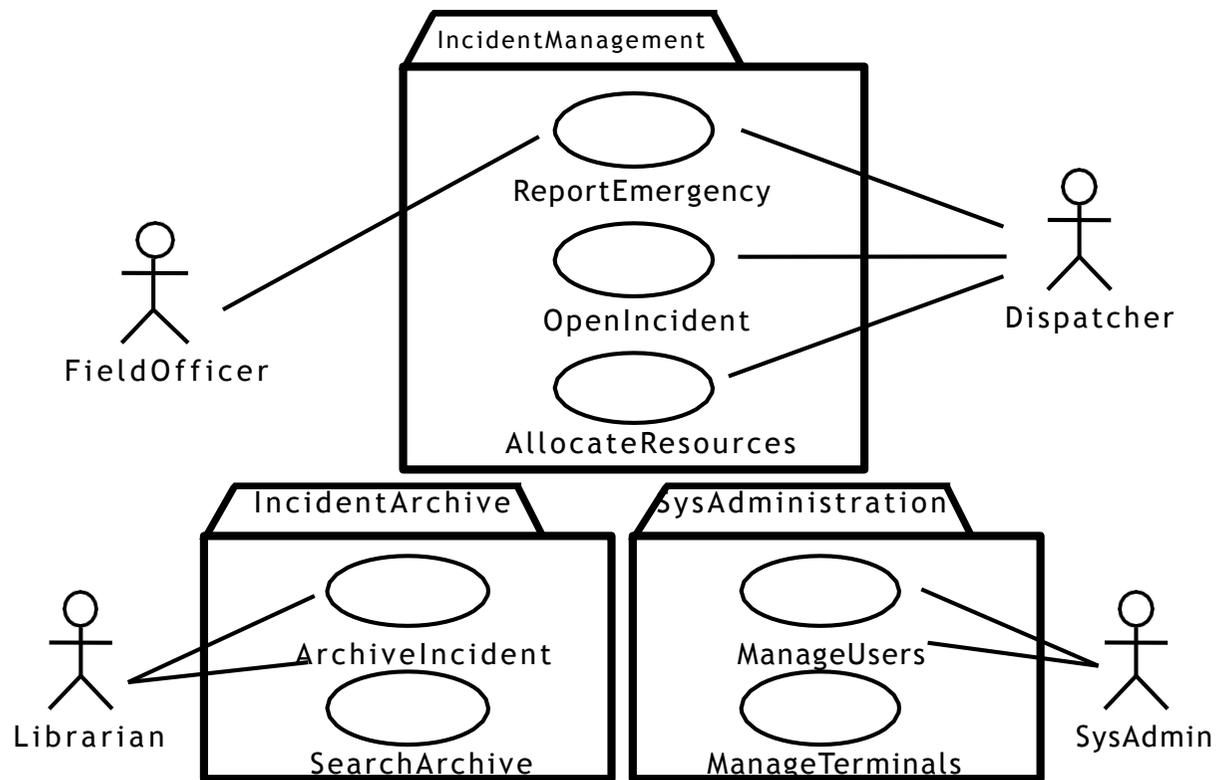


La complessità dei modelli può essere gestita raggruppando elementi in relazione tra loro in **PACKAGE**

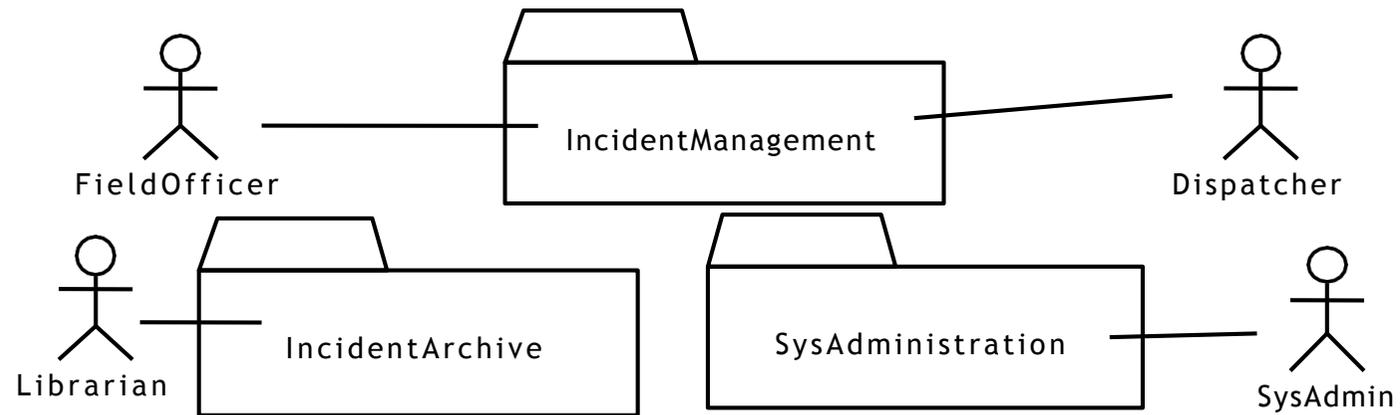


Un package è un raggruppamento di elementi, come casi d'uso, classi, o attività che semplificano la comprensione

ESEMPIO DI PACKAGE: CASI D'USO DI FRIEND ORGANIZZATI PER ATTORI

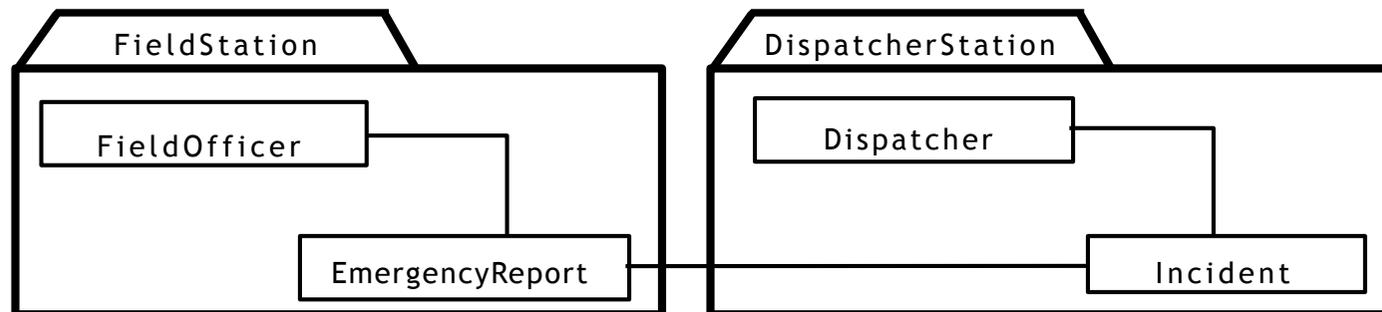


ESEMPIO DI PACKAGE



PACKAGE NEI DIAGRAMMI DELLE CLASSI

- Anche i diagrammi delle classi possono essere organizzati in package
- Le classi del caso d'uso *ReportEmergency* sono organizzate rispetto al punto in cui gli oggetti sono creati
 - *FieldOfficere EmergencyReport* sono parte del package *FieldStation*
 - *Dispatcher e Incident* sono parte di *DispatcherStation*



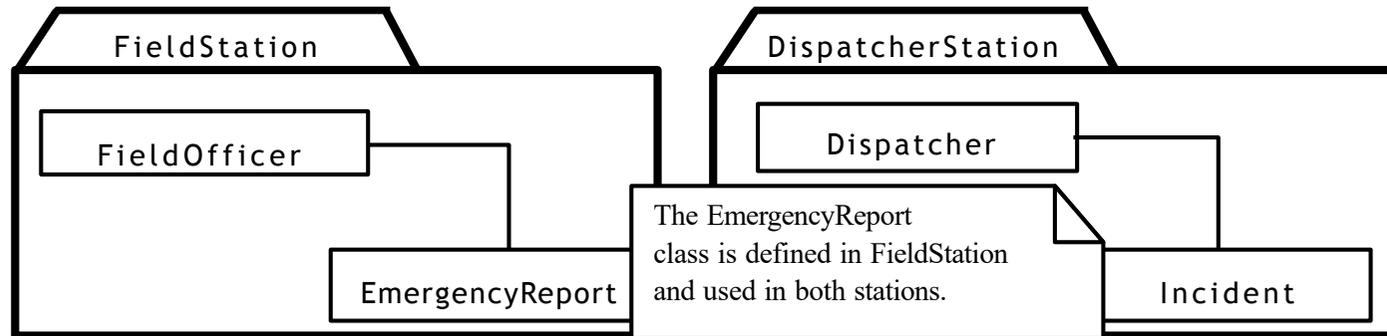
ANCORA SUI PACKAGE

- I package possono essere usati per trattare la complessità alla stregua dell'organizzazione di file e directory di un generico utente
 - La differenza è che i package non sono necessariamente gerarchici:
 - la stessa classe può trovarsi in più di un package
- Per ridurre le inconsistenze, le classi appartengono ad un solo package e gli altri package fanno riferimento ad essa

NOTE

- Una **nota** è un commento allegato al diagramma
- Sono usate dagli sviluppatori per allegare informazioni ai modelli e agli elementi del modello
- E' un meccanismo per ricordare questioni rilevanti di un modello, per chiarire un punto complesso e per tener traccia di un punto da svolgere in futuro

ESEMPIO DI NOTA: LE NOTE POSSONO ESSERE ALLEGATE AD UN ELEMENTO DEL DIAGRAMMA SPECIFICO



ESTENSIONE DEI DIAGRAMMI

- L'obiettivo di UML consiste nel fornire un insieme di notazioni per modellare un'ampia classe di sistemi software
- Tuttavia, un insieme prefissato di notazioni non potrebbe consentire di perseguire un tale scopo
 - **È impossibile anticipare tutte le necessità che si verificano in tutti i domini applicativi**
- UML fornisce un numero di meccanismi che consentono di estendere il linguaggio

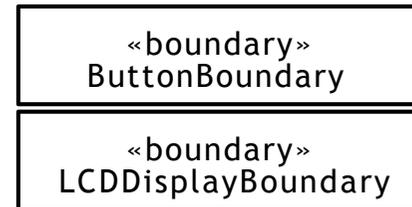
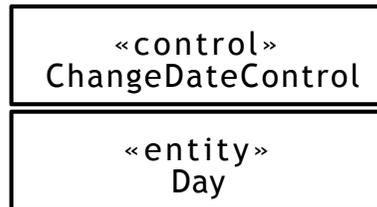
STEREOTIPI

- Uno **stereotipo** è un meccanismo di estensione che consente di classificare gli elementi del modello in UML
- E' rappresentato da una stringa racchiusa tra parentesi angolari <<>> e allegata all'elemento a cui si applica, come una classe o un'associazione
- Formalmente, uno stereotipo corrisponde, semanticamente, a creare una nuova classe nel meta- modello UML (ovvero, il modello che rappresenta i costrutti di UML)

STEREOTIPI

- Esempio:
 - Durante l'analisi, classifichiamo gli oggetti in tre categorie:
 - *entity*,
 - *boundary*
 - *control*
 - Hanno la stessa struttura ma scopi diversi
- Il linguaggio base di UML include solo un tipo di oggetto. Per rappresentare le tre categorie usiamo gli stereotipi <<entity>>, <<boundary>> e <<control>>

ESEMPIO DI STEREOTIPI



VINCOLI

- Un vincolo è una regola che viene allegata ad un elemento restringendone la semantica
- Consente di rappresentare fenomeni che non potrebbero essere espressi altrimenti in UML
- Esempio
 - Un incidente (*Incident*) può essere associato con uno o più *EmergencyReport* prodotti dal luogo dell'evento
 - I *Dispatcher* devono essere in grado di esaminare i rapporti in ordine cronologico
 - L'ordinamento cronologico degli *EmergencyReport a Incident* è espresso dal vincolo *{ordinato in base all'ora di ricezione}*
 - I vincoli possono essere espressi in linguaggio naturale o mediante OCL (Object Constraint Language)

ESEMPIO DI VINCOLO



ATTIVITÀ DI SVILUPPO E RELATIVI PRODOTTI

