Métodos de Desenvolvimento de Software (MDS) 2014/2015

Package Diagrams

Package Diagrams: goal

Manage complexity of the diagrams, grouping together elements of those diagrams into ***packages***.

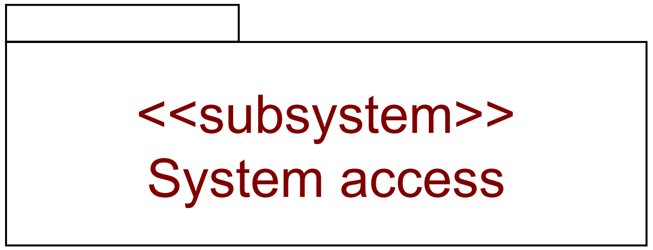
- criar diagramas de alto nível de abstracção (de uma colecção de casos de uso, de classes, etc)
- A *package is a collection of UML elements logicaly related*.
- A package diagram is composeb by packages and their relations.

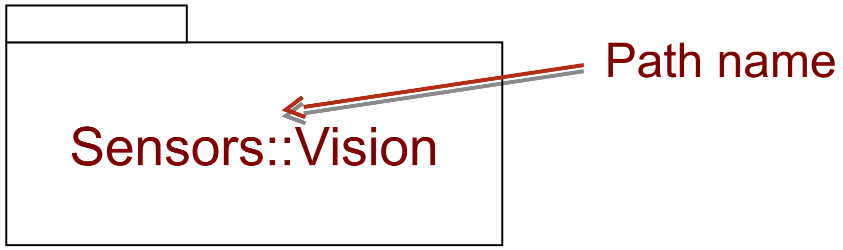
Package Diagrams: introduction

- A Package is a eneral purpose mechanism for organizing group elements
  - Absolutely necessary in big systems to deal with scale
- A package can contain other elements including: classes, interfaces, diagramas, components, nodes, use cases and other
  - Subsystems group together objects (and other subsystems), reducing the complexity of a system
- We should avoid excessive nesting
- To use packages:
  - Is easy for maagement and search of elements in a model
  - Avoida name conflicts
  - Has a visibility mechanism

Packages Notation

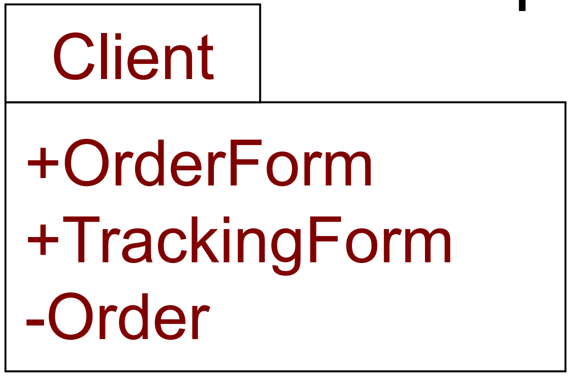
- It is represented by a *tabbed folder*
- All *packages* have a name that distinguishes them from the other packagesEach element can only be part of one *package*
- The *path name* is the name of a package with the prefix of the name of the package of which the packge is in



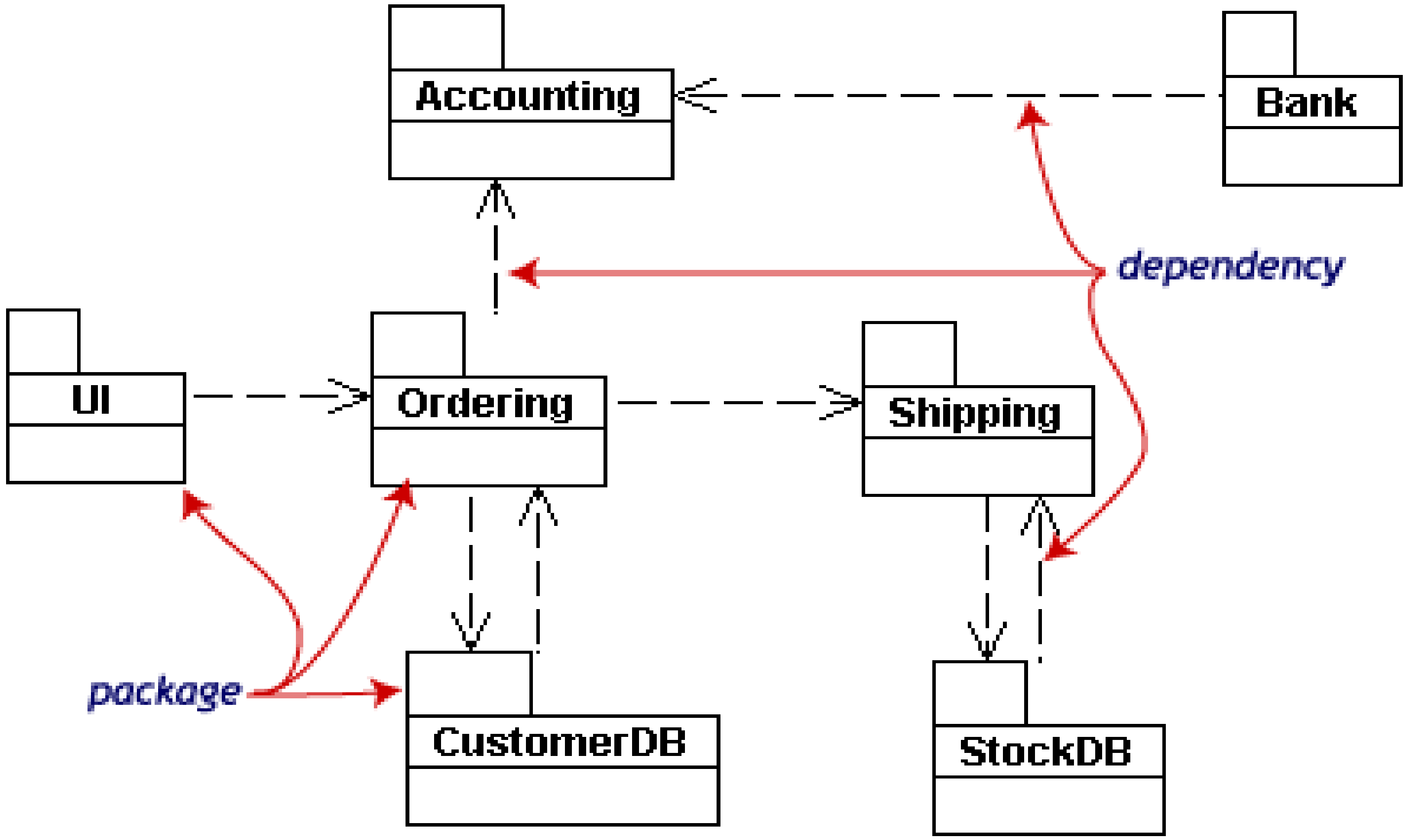


Visibility

- **+ (*public*)**: um elemento de um pacote **X** é público se é visível para os elementos de um pacote **Y** que importa o pacote **X**
- **# (*protected*)**: elementos protegidos só podem ser vistos pelos pacotes filhos
- **- (*private*)**: Elementos privados não podem ser vistos fora do pacote em que estão declarados



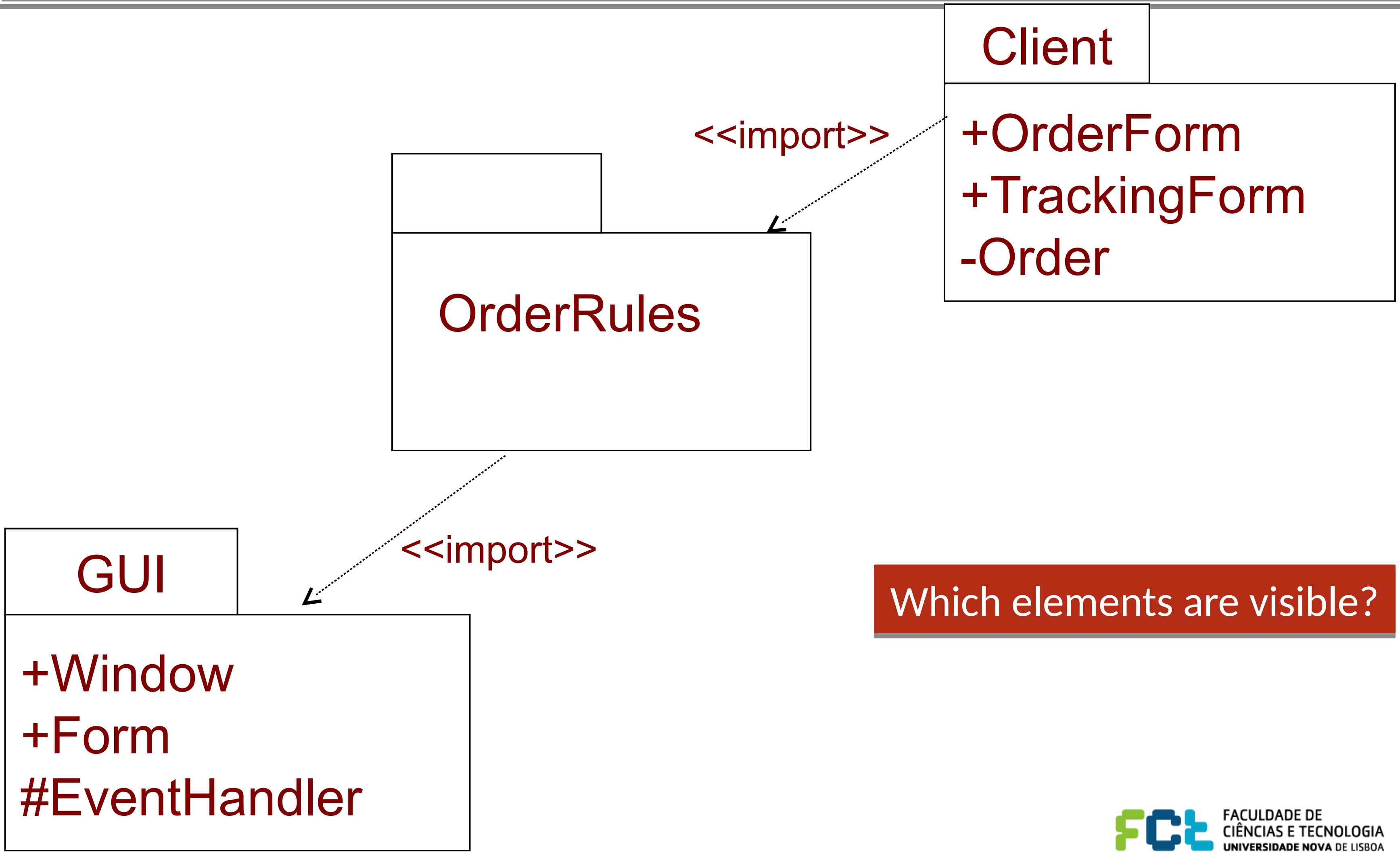
Dependencies among packages



Import and Export

- The public elements of a package are called **exports**
- If **X** **imports** **Y**, a elemento of **X** can see the public elements of Y, but an element of **Y** can not see the elements of **X**.
- The import is represented by a dependency with stereotype **`<<import>>`**
- If an element is visible in **X**, it is visible within all the nested packages of **X**

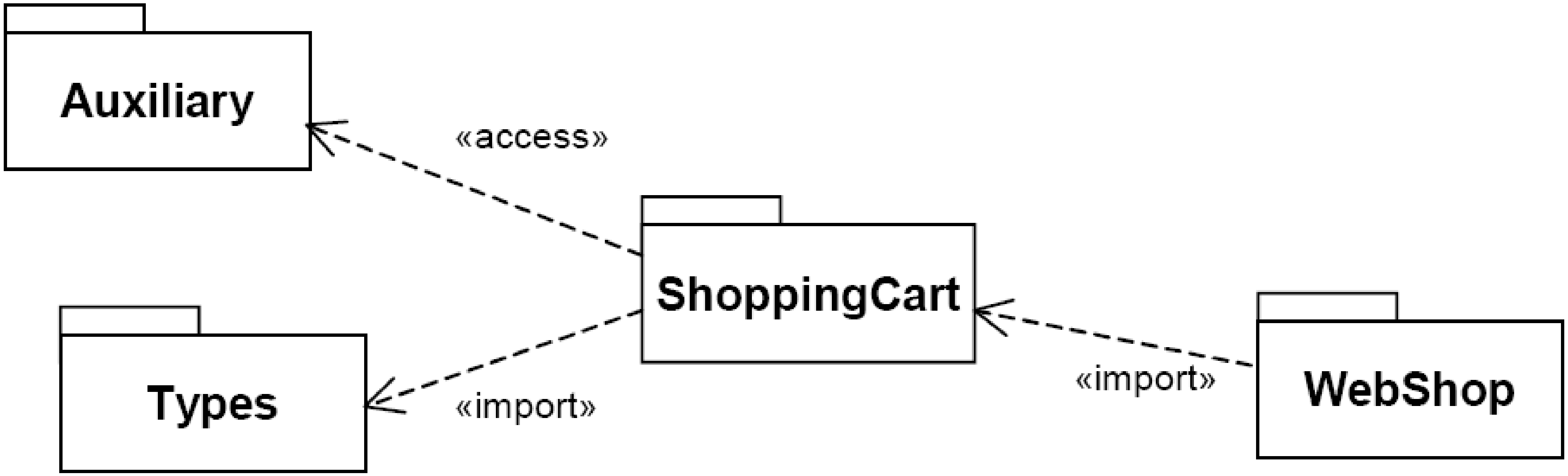
Visibility import/export



Which elements are visible?

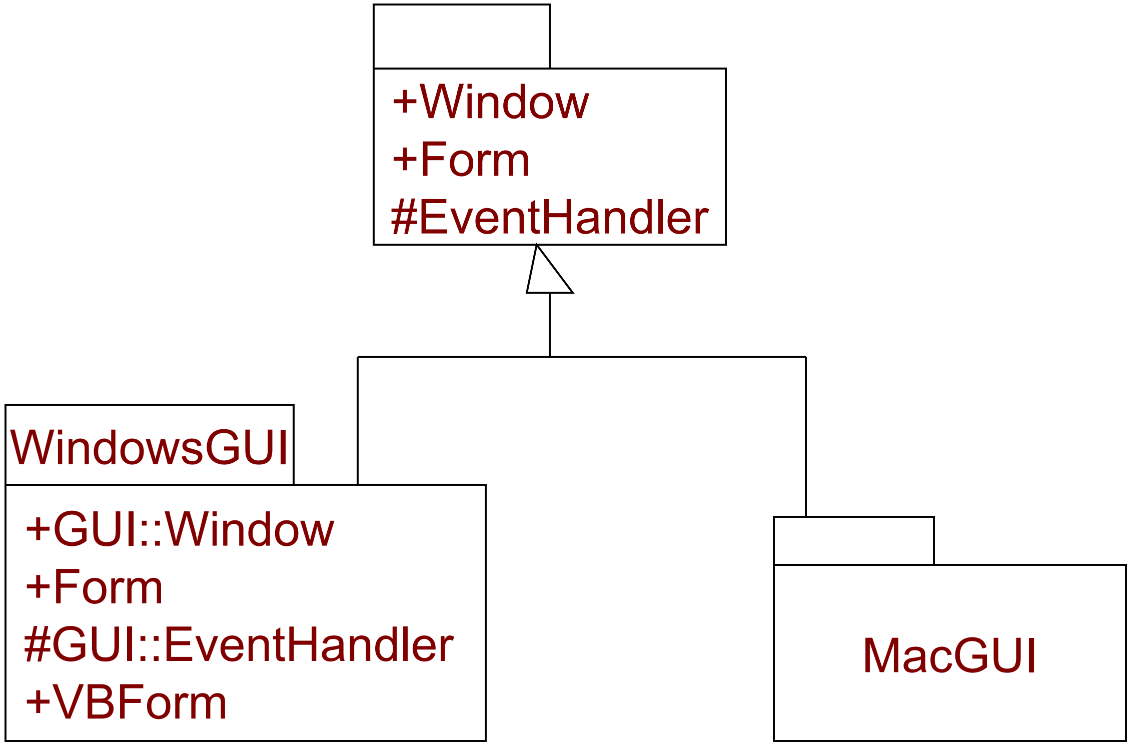
<<access>>

- The elements of `Auxiliary` are only accessed by `ShoppingCart` (this kind of import turns the imported elements into private)

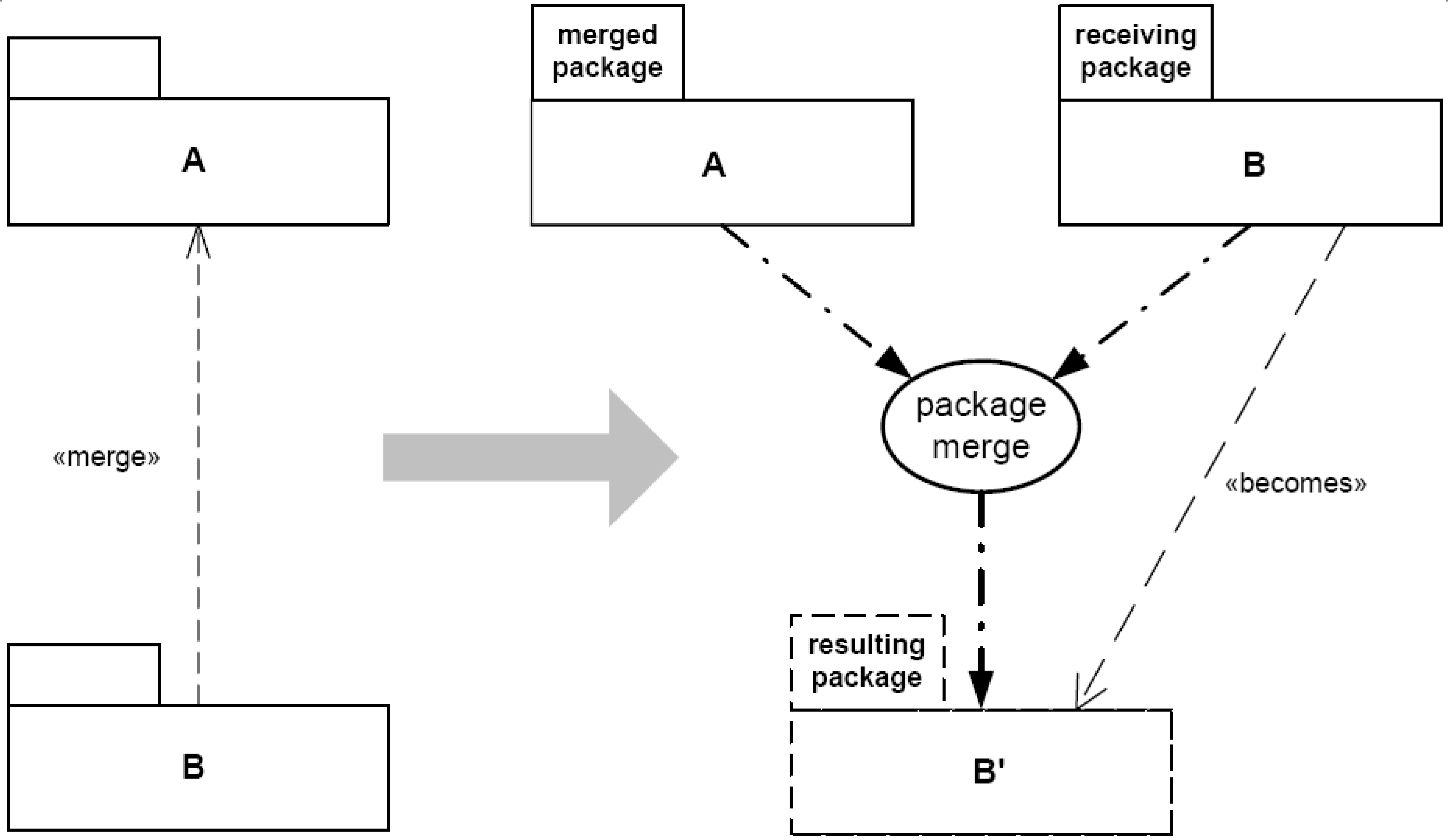


Generalization

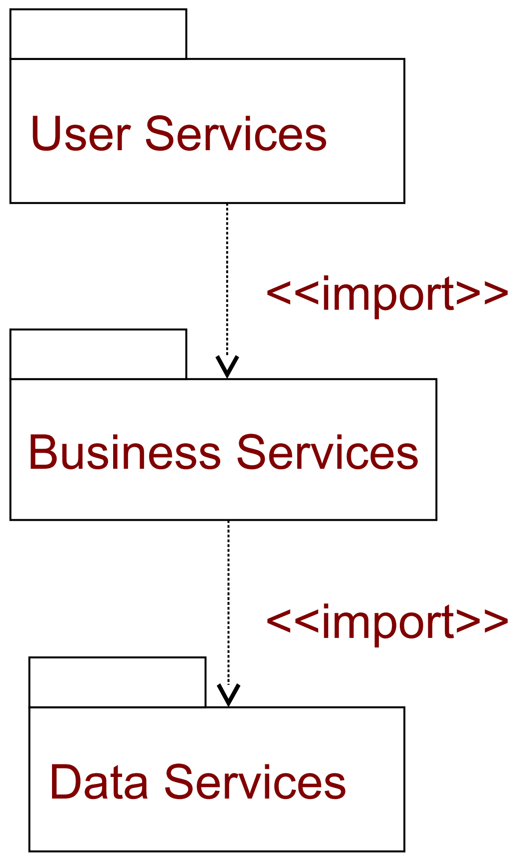
- Specifies a family of packages



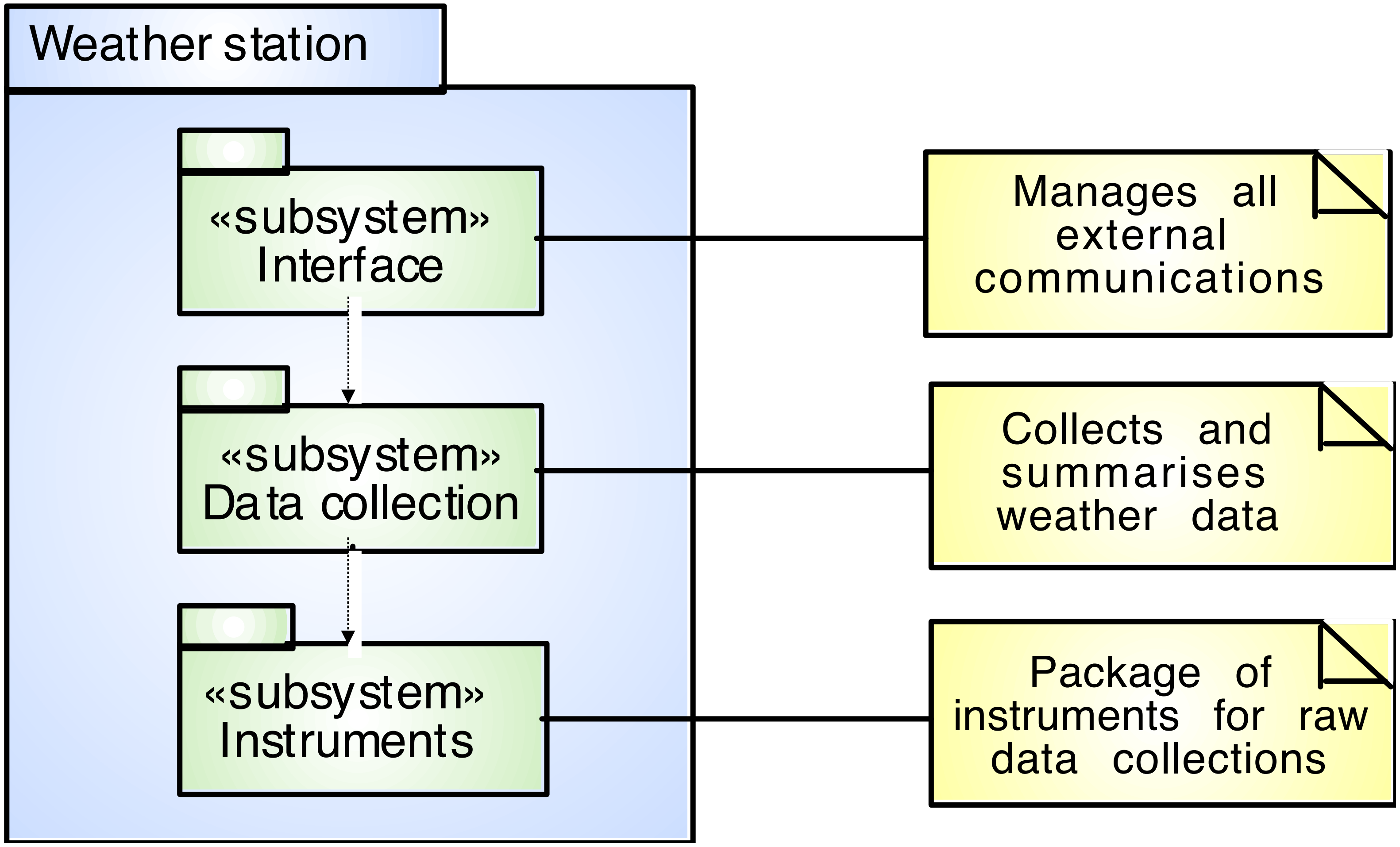
Fusion (*Merge*) of packages



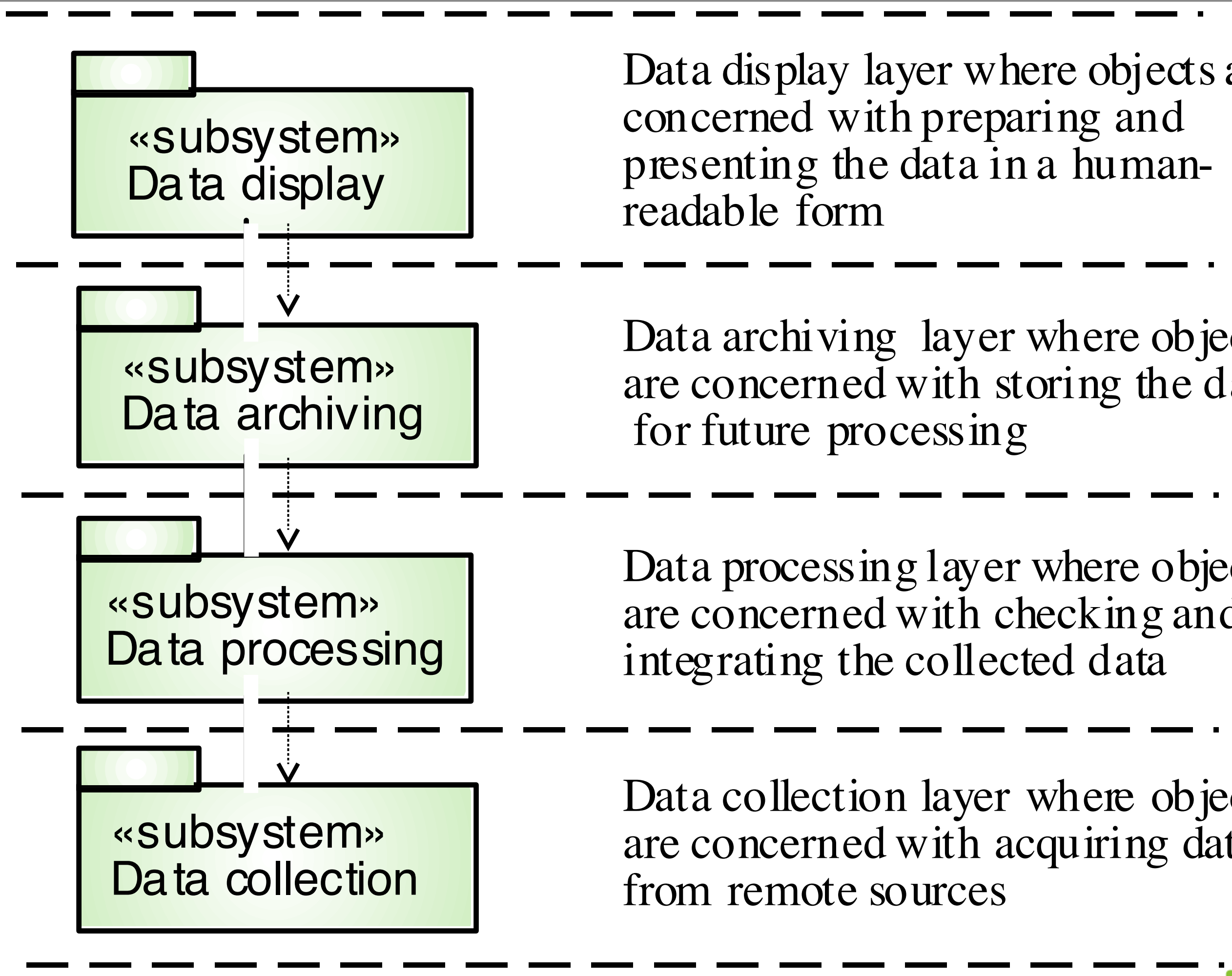
Three-tier architecture



Architecture of a weather station


Weather station
«subsystem» Interface
«subsystem» Data collection
«subsystem» Instruments
Manages all external communications
Collects and summarises weather data
Package of instruments for raw data collections

Layered architecture


«subsystem» Data display
«subsystem» Data archiving
«subsystem» Data processing
«subsystem» Data collection

Data display layer where objects are concerned with preparing and presenting the data in a human-readable form

Data archiving layer where objects are concerned with storing the data for future processing

Data processing layer where objects are concerned with checking and integrating the collected data

Data collection layer where objects are concerned with acquiring data from remote sources

FACULDADE DE CIÊNCIAS E TECNOLOGIA UNIVERSIDADE NOVA DE LISBOA