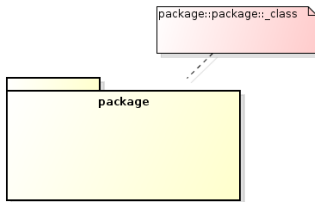


## Diagrammi dei package(4)

Sono usati spesso nella parte architetturale, ci permettono di tenere sotto controllo la complessità del sistema che si misura con le dipendenze. Un package non è altro che un raggruppamento di elementi UML. In effetti andremo a raggruppare solo classi. I rapporti tra package UML e package di linguaggio di programmazione è molto stretto, quasi 1:1. Il package si presenta come una grande cartella che ha un titolo e contiene al suo interno classi o altri package. Una classe appartiene ad un solo package. Il package individua un *namespace*, ogni elemento deve avere un nome distinto all'interno dello spazio dei nomi. In UML per riferirsi ad un nome qualificato si usa la notazione dei :: .

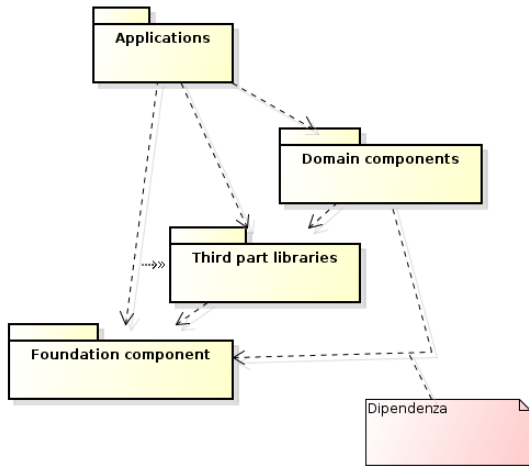


L'interfaccia di un package è l'insieme delle classi che hanno visibilità pubblica di un package.

Principi di progettazione:

- **Common Closure Principle**, classi dello stesso package condividono la stessa causa di cambiamento;
- **Common Reuse Principle**, classi dello stesso package dovrebbero sempre essere riusate insieme.

I package possono avere **dipendenze** tra loro:



Caratteristiche:

- Tutte le dipendenze dovrebbero seguire la stessa direzione, a meno di isolamento voluto da sottostrutture;
- Evitare le dipendenze circolari;
- Relazioni di dipendenza non soltanto transitive, se modifico *aaa* non necessariamente modifico *ccc*;
- Più dipendenze entranti, più il *package* dovrebbe essere stabile.

**Esempio:**

*Il cliente sfoglia il catalogo ed aggiunge i prodotti desiderati al carrello della spesa. Quando il cliente termina l'acquisto e deve pagare, lo stesso fornisce le informazioni sulla consegna dei prodotti e sulla carta di credito.*

*Il sistema verifica l'autorizzazione al pagamento con carta di credito e conferma l'acquisto immediatamente e mediante una successiva mail.*

