

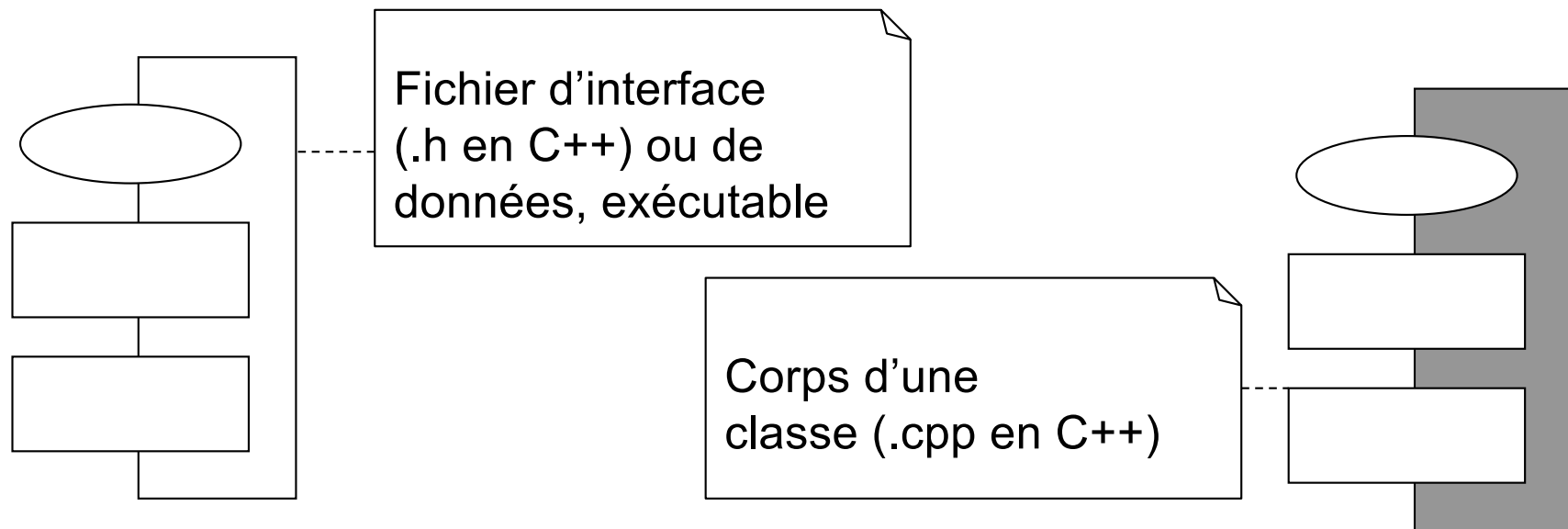
SysML/UML

Diagrammes d'architecture

Diagrammes de composants

2.

Utilisés essentiellement pour les applications de taille très importante. Permettent de préciser l'architecture de l'application en modules : fichiers d'entête, fichiers sources, exécutables, bibliothèques etc. en précisant les relations de dépendance. Notation des modules :

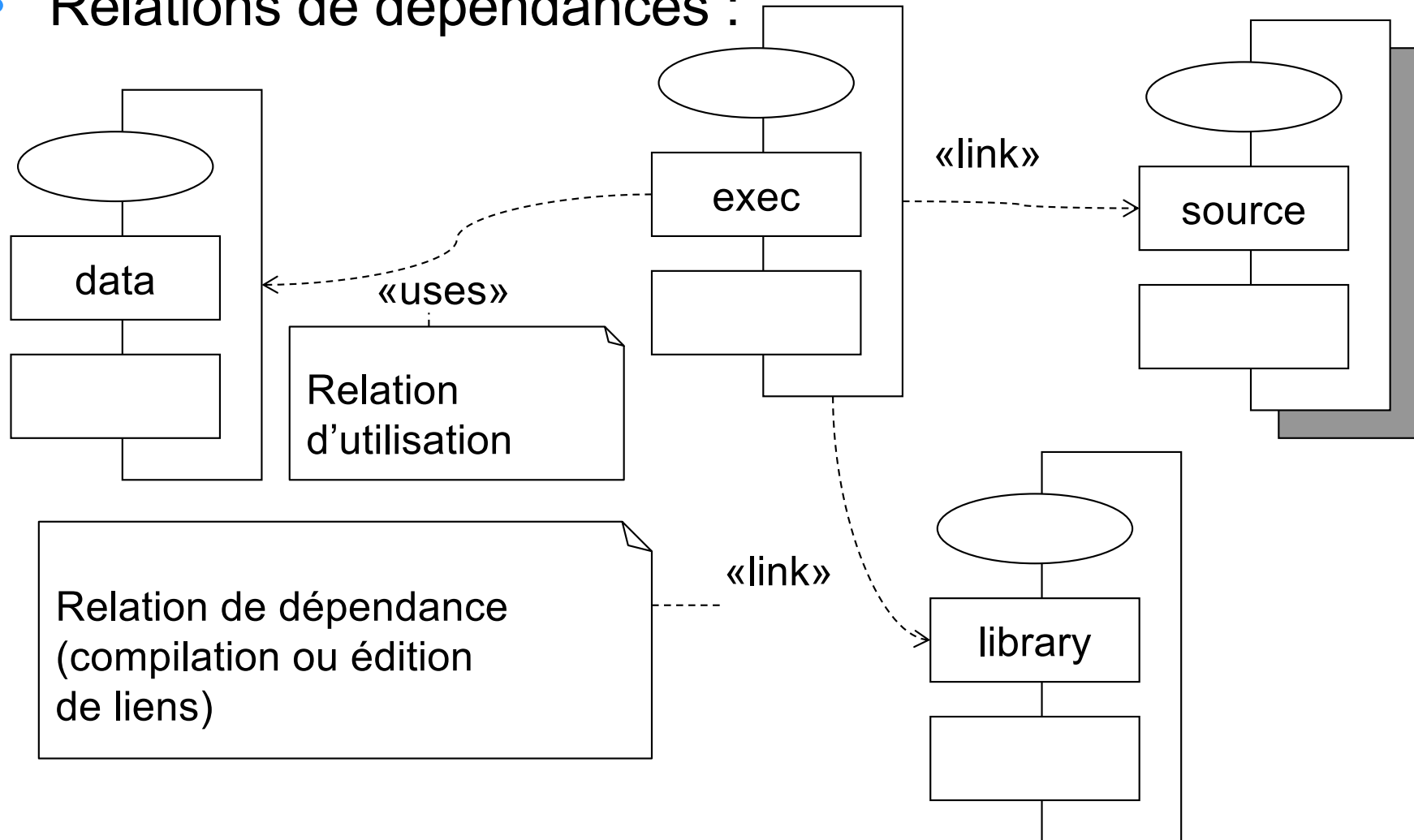


- SysML n'a pas repris les diagrammes de composants

Diagrammes de composants

3

- Relations de dépendances :



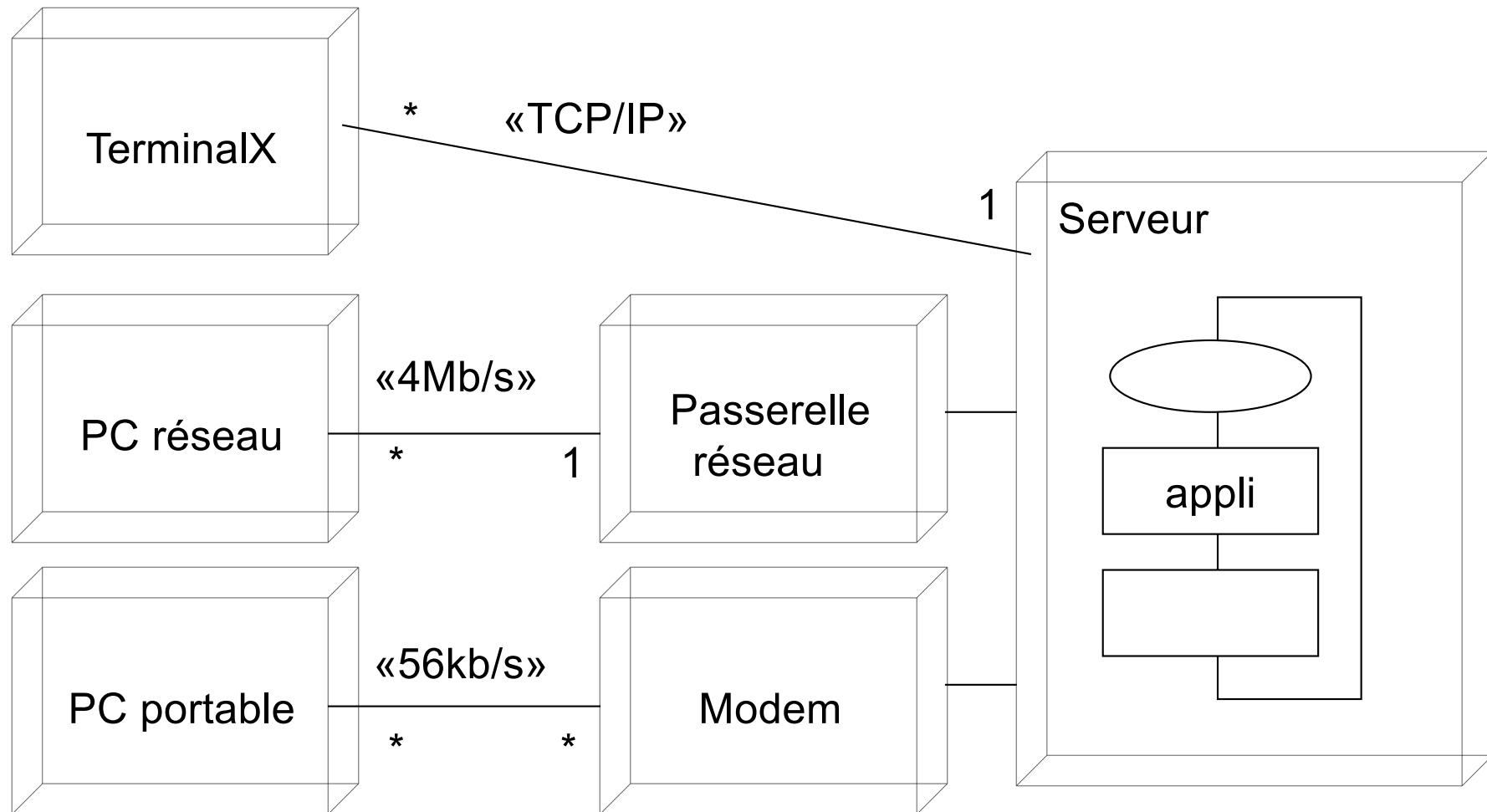
Diagrammes de déploiement

4

- Peu utilisés en dehors des très grosses applications réparties, ils montrent :
 - Les ressources matérielles ou nœuds (stations, serveurs, terminaux...) sur lesquels se déroule l'application
 - Les liens entre nœuds avec précision possible des caractéristiques des communications
 - Eventuellement l'implantation des composants sur les nœuds
- Peuvent montrer des classes de matériels (diagrammes type diagrammes de classes avec indication de multiplicité) ou des instances (diagrammes type diagrammes d'objets)
- **SysML n'a pas repris les diagrammes de déploiement**

Diagrammes de déploiement

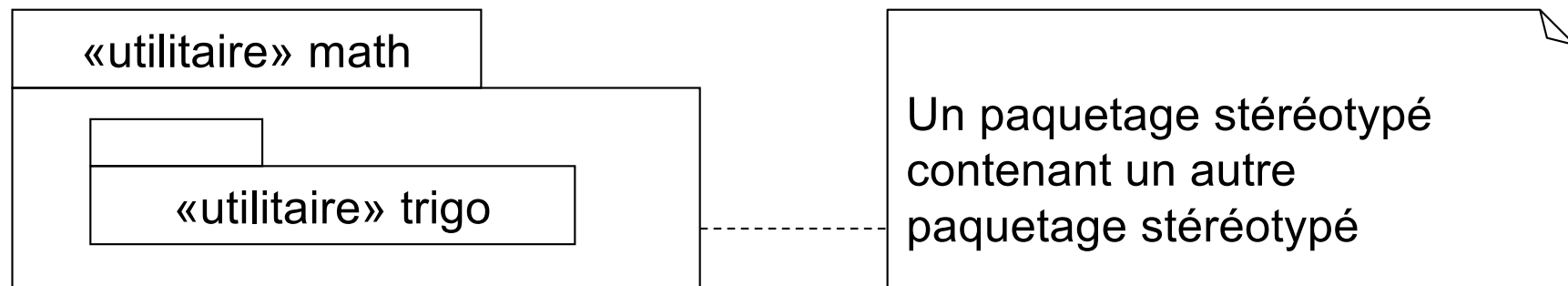
5



Diagrammes de paquetages (« packages »)

6

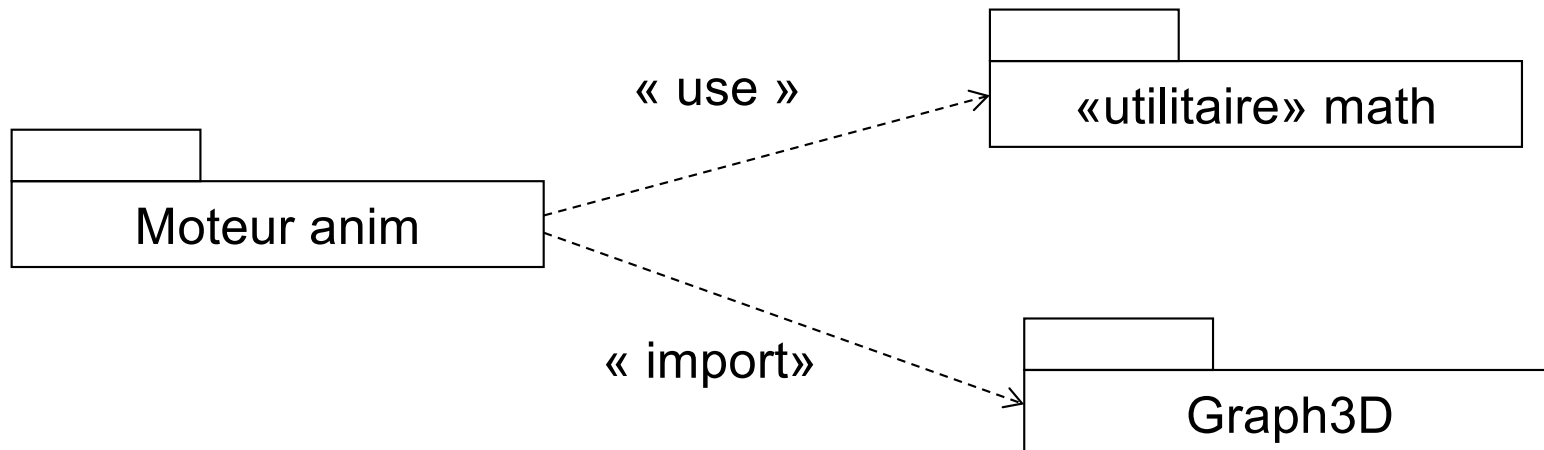
- UML permet d'organiser un modèle complexe en différents paquetages ou «packages».
- Représentés graphiquement comme des dossiers, les paquetages peuvent contenir tout type d'élément de modélisation UML (y compris d'autres paquetages sans limite de niveaux d'emboîtement).
- Des relations de dépendances ou de hiérarchie entre paquetages peuvent également être définies



Diagrammes de paquetages


7

- Les diagrammes de paquetages sont devenus des diagrammes standard UML 2.0
- Visent essentiellement à illustrer la structure (hiérarchie) des packages et les dépendances éventuelles (en les caractérisant et en cherchant à les minimiser). Ici des dépendances stéréotypées :



Diagrammes de paquetages particularités SysML

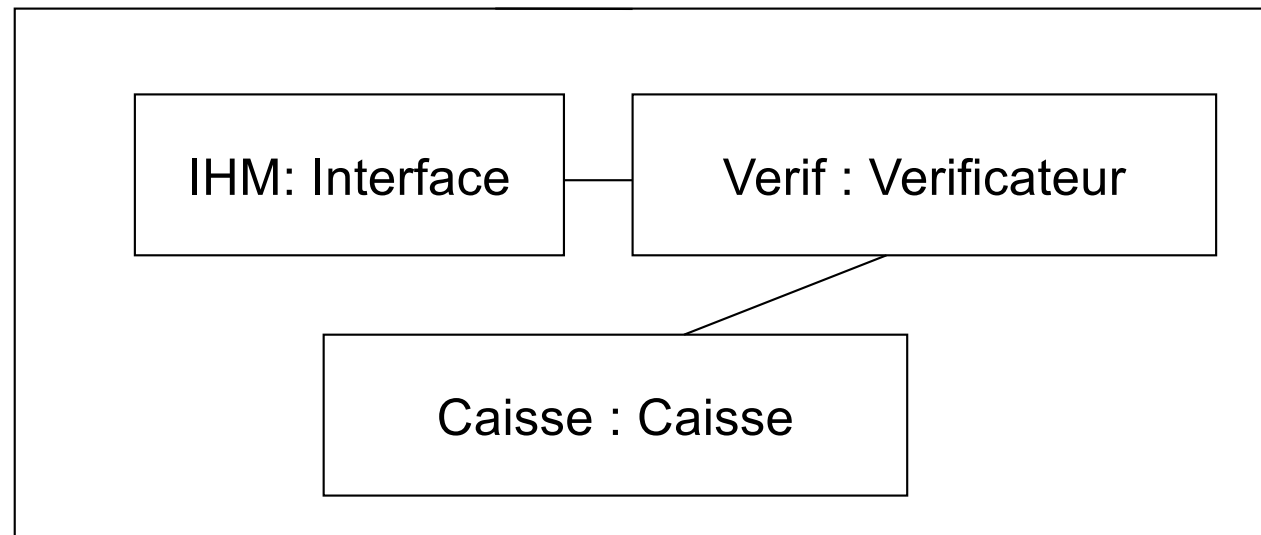
8

- Pratiquement aucune :
 - Possibilité de visualiser la hiérarchie d'inclusion par le même graphisme que dans les diagrammes d'exigences
(liens avec croix entourée coté conteneur) : 
 - Définitions de certains stéréotypes de packages :
<<model>> : racine d'une hiérarchie, <<view>> : vue particulière d'un système (sécurité, performances...),
<<viewpoint>> : contenu d'une vue selon différents critères...

Diagrammes de structure composite

9

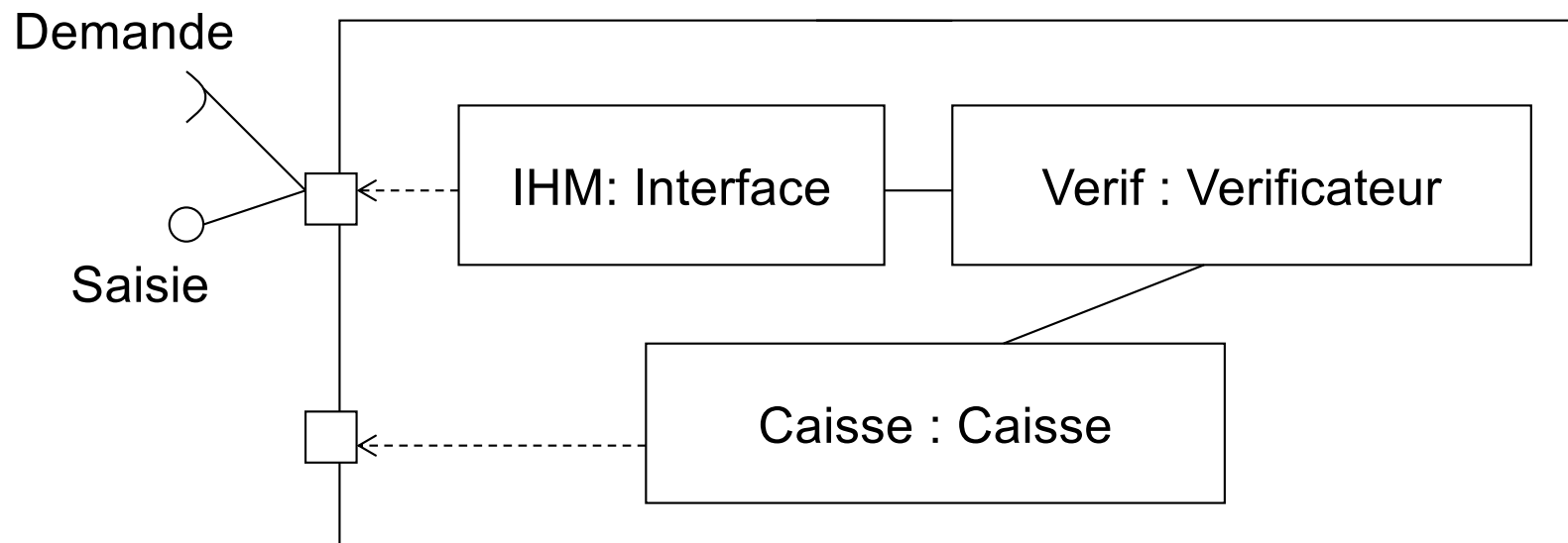
- Offrent la possibilité de décomposer une classe en une structure interne : attention ces parties ne sont pas des instances (donc nom non souligné)
- Il est possible de préciser une multiplicité pour chaque partie (exemple deux IHM en redondance)



Diagrammes de structure composite

10

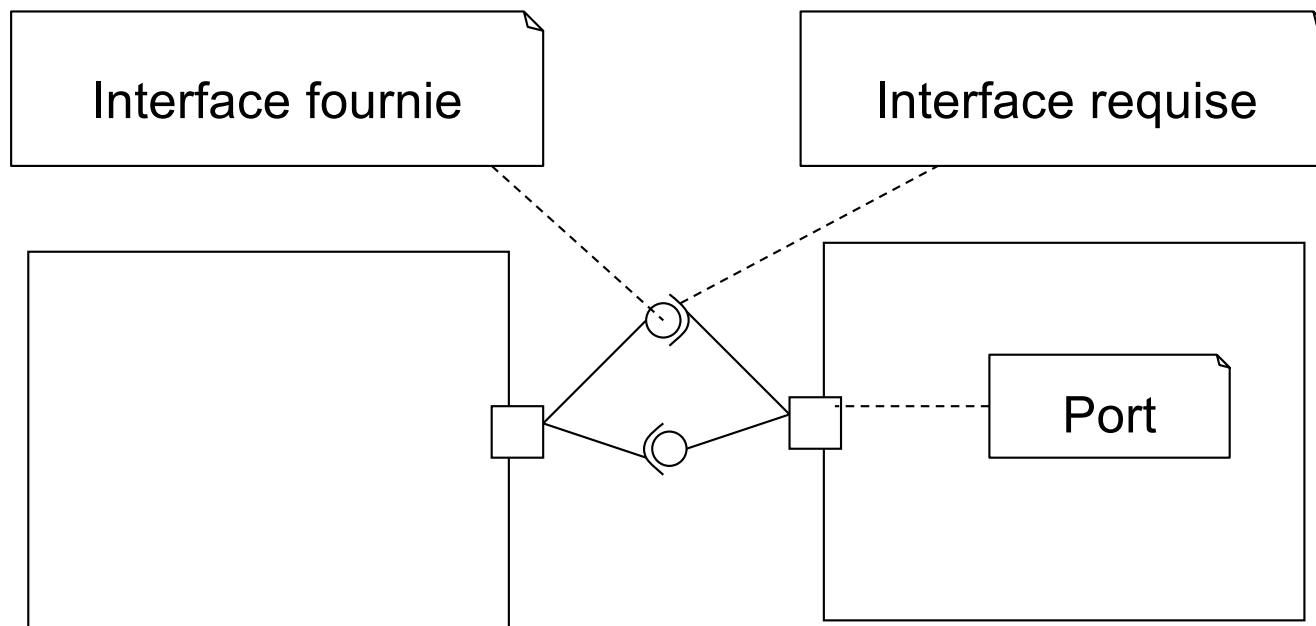
- Incluent également la notion de port qui regroupent les interfaces réalisées et requises vis à vis de l'extérieur :



Diagrammes de structure composite

11

- Cette notation des interfaces (l'une des nouveautés majeures d'UML 2.0) permet de mettre en évidence les connexions d'interface entre classes



Diagrammes de bloc interne SysML

12

- Le **diagramme de bloc interne** (Internal Block Diagram : IBD) reprend le diagramme de structure composite pour les blocs SysML.
- Les composants à l'intérieur d'un bloc s'appellent ses **parties** (parts)
- Les liens entre parties (donc internes à un bloc) s'appellent **connecteurs**
- Les notions de **ports** et **d'interfaces** (donc externes à un bloc) sont reprises telles quelles.
- Les **ports** peuvent être de type **standard** ou **flux** (flux physique : matière, énergie...).
- Un port d'entrée seulement ou de sortie seulement peut être représenté avec seulement une flèche

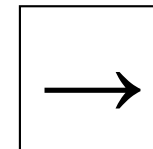


Diagramme Paramétrique SysML

13

- SysML inclut une spécialisation du diagramme de bloc interne pour en faire le **diagramme paramétrique**.
- Le diagramme paramétrique comprend des blocs stéréotypés <<Constraint>> représentant les contraintes.
- Le bloc Constraint contient la contrainte (souvent une loi physique exprimée entre accolades comme en UML) et les paramètres de cette loi.

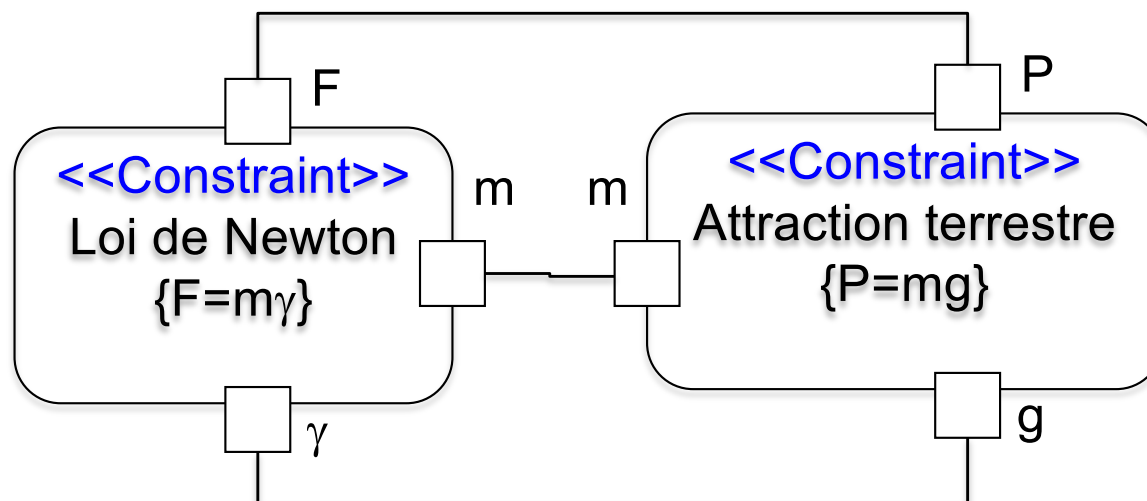
<<Constraint>> Loi de Newton
constraints {F=m γ }
Parameters F force : N m masse: kg γ accélération : m/s ²

<<Constraint>> Attraction terrestre
constraints {P=mg}
parameters P Poids : N m masse : kg g accélération : m/s ²

Diagramme Paramétrique SysML

14

- Ces blocs contraintes sont ensuite « **instanciés** », les paramètres figurant comme ports sur les « instances » qui sont figurés par des rectangles aux coins arrondis.
- Les paramètres sont reliés entre eux (**value binding**) ce qui permet (paraît-il...) de supporter des analyses Simulink etc.



Moralité : la plume et l'enclume tombent avec la même accélération constante...