

Modélisation des SED et UML

Pavol BARGER
SY08 A11 cours 13

Approches SED

- Automates temporisés
 - UPPAAL
- Réseaux de Petri
 - CPN tools
- StateMate
- StateFlow
- UML

2

Objectifs UML

- représenter des systèmes entiers (non seulement le logiciel) par des concepts objet
- établir un couplage explicite entre les concepts et les artefacts exécutables
- prendre en compte les facteurs d'échelle inhérents aux systèmes complexes et critiques
- créer un langage de modélisation utilisable à la fois par les humains et les machines

3

Modèle et metamodelle

- Modèle
 - une description abstraite d'un système ou d'un processus
 - représentation simplifiée
 - pas de solution unique
- Metamodelle
 - décrit les éléments de modélisation, la syntaxe et la sémantique
 - règles de création de modèles

4

Modélisation

- Analyse du système, processus
 - décomposition en éléments simples
- Modélisation des états
- Modélisation des fonctions
- Modélisation des objets/entités
- Chaque modèle a une vue sur la réalité

5

Vues UML

- Statiques
 - diagrammes de classes
 - diagrammes d'objets
 - diagrammes de cas d'utilisation
 - diagrammes de composants
 - diagrammes de déploiement

6

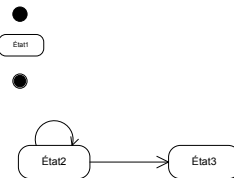
Vues UML

- Dynamiques
 - diagrammes de séquence
 - diagrammes de collaboration
 - diagrammes d'états-transitions
 - diagrammes d'activités

7

Diagrammes d'états-transitions

- Basés sur les automates d'états finis
- 3 types d'état
 - état initial
 - état normal
 - état final
- transitions



8

Transitions, événements

- Un événement peut être associé à une transition
- Un événement n'est pas un état. C'est une information instantanée.
- A l'occurrence d'un événement, l'état peut changer

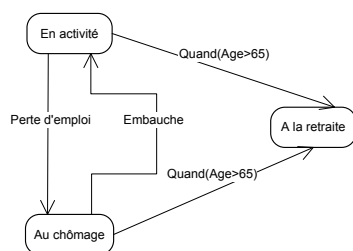
9

Événements

- 4 types d'événements
 - événement signal
 - envoi et réception asynchrones
 - événement appel d'une opération
 - événements création et destruction d'objet
 - création déclanche l'état initial
 - événement temporel
 - à la fin d'une temporisation
 - relatif ou absolu
 - événement modification
 - passage d'une fonction booléenne de faux à vrai
 - événement mémorisé

10

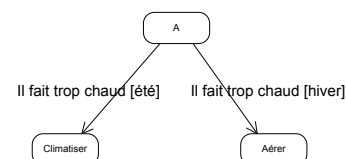
Exemple



11

Gardes

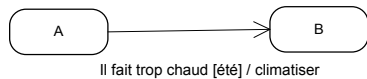
- Une condition de garde
 - booléenne
 - notée entre crochets



12

Actions

- Actions sont associées aux transitions ou aux états
- Chaque action est instantanée et atomique, donc ininterrompible
- Une action correspond à une génération d'un signal



13

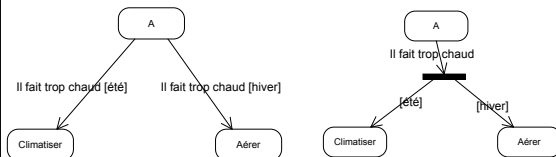
Actions dans l'état

- Une action peut être exécutée
 1. à l'entrée d'un état
 2. à la sortie d'un état
 3. pendant la durée de l'état

14

Transitions composites

- Statique



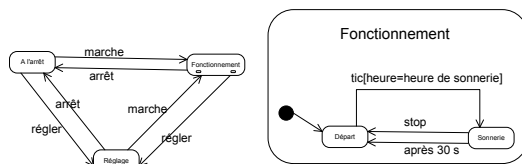
15

États composites

- Un état composite est un état décomposé en sous-états
- Chaque sous-état peut être composite à son tour
- Décomposition hiérarchique
- Exemple :
 - étudiant présent ou absent du cours
 - étudiant présent peut être attentif ou endormi

16

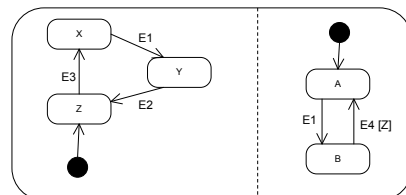
Exemple d'un réveil



17

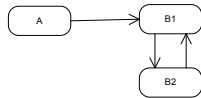
États composites concurrents

- Composition parallèle synchronisée
 - chaque événement peut se produire à tout moment
 - différence avec la composition parallèle des automates



18

États composites

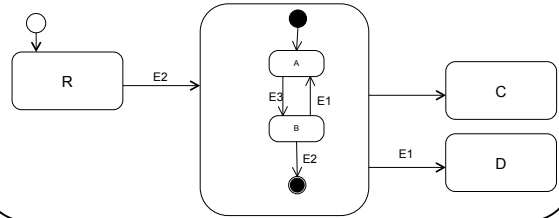


- Mélange de niveaux hiérarchiques
 - A éviter
- Chaque élément doit s'adresser exclusivement aux éléments de son niveau

19

Sortie automatique

- La transition a lieu à l'arrivée dans l'état final



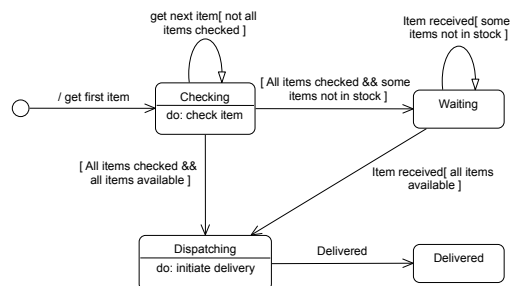
20

Exemple : Annulation d'un ordre

- Modèle de gestion de stock
- Solutions
 - Transitions d'annulation de chaque état
 - Définition d'un macroétat

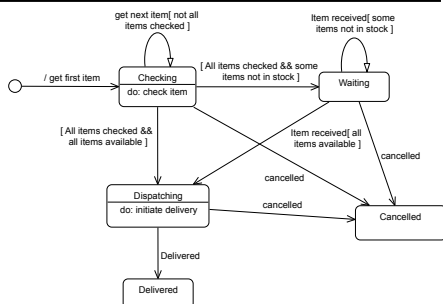
21

Exemple stock



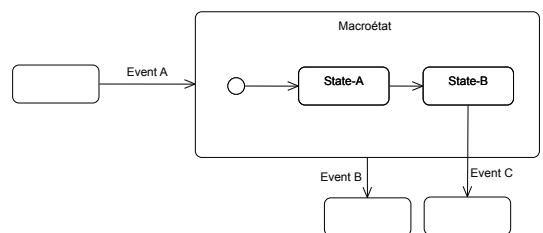
22

Transitions d'annulation



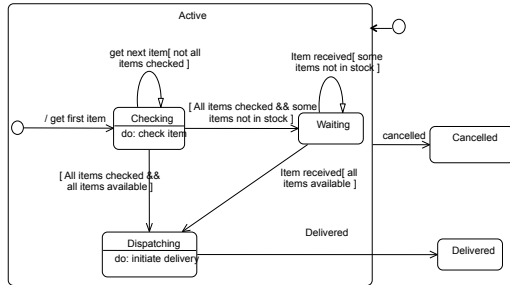
23

Macroétat : définition



24

Exemple : Solution macroétat



25

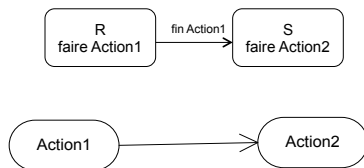
Remarques

- Un seul état initial est permis dans chaque état composite
- État final représente la sortie de l'état composite

26

Diagrammes d'activités

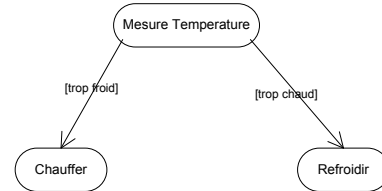
- Inspirés de diagrammes états-transitions
- Centrés sur les activités



27

Diagrammes d'activités

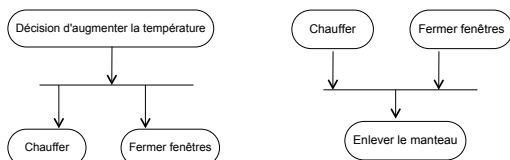
- Transitions se passent le plus souvent à la fin de l'action précédente



28

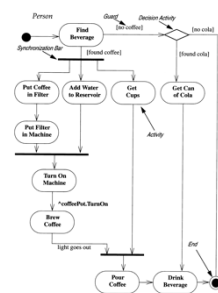
Diagrammes d'activités

- Synchronisation



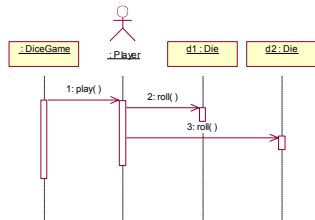
29

Diagrammes d'activités



30

Diagramme de séquences



- modèle de stop and wait

31

UML conclusion

- UML permet d'exprimer les systèmes à événements discrets
- Formalisme répandu
- Néanmoins cette représentation n'est pas suffisamment rigoureuse → formalisme semi-formel

32