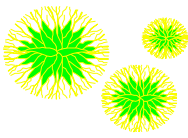


Information Systems Analysis and Design CSC340

XI. State and Activity Diagrams

State Diagrams
 Events and States
 Superstates
 Activity Diagrams
 Petri Nets



© 2002 Jaelson Castro and John Mylopoulos Statechart Diagrams -- 1

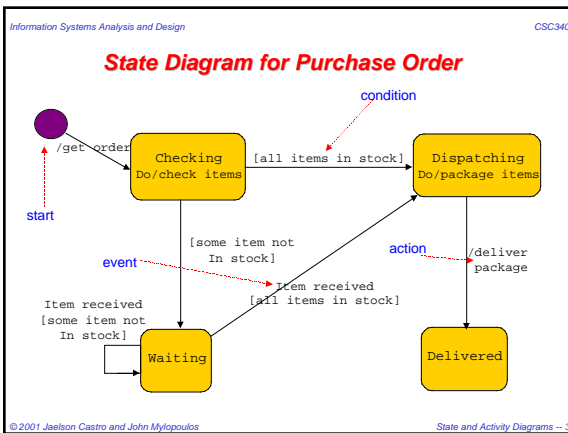
Information Systems Analysis and Design CSC340

State Diagrams (Statecharts)

- These are state transition diagrams (with some interesting additions) which can be used to describe the operating environment of the system, interactions of the system with that environment, also the lifetime of some object (a person, a student,...) within the system.
- Transitions are supposed to represent actions which occur "quickly" and are not interruptible. A transition can have an associated triple

$$\text{Event}[\text{Condition}]/\text{Action}$$
 all parts of this triple are optional.
- States are supposed to represent longer-running activities (or other things). What constitutes "quickly" and "longer-running" depends on the application.

© 2001 Jaelson Castro and John Mylopoulos State and Activity Diagrams -- 2

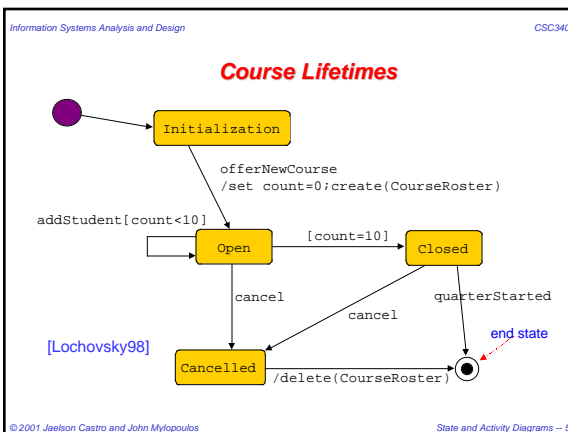


Information Systems Analysis and Design CSC340

Events

- An **event** is a happening that the system needs to know about.
- For example, completing an assignment, failing an exam, or a system crash are all events.
- An event may trigger an action by an actor or the system being developed.
- In UML, there are four types of events:
 - ✓ **Change events** occur when a condition becomes true, e.g.,
 $\text{when}(\text{balance} < 0);$
 - ✓ **Signal events** designate the receipt of an explicit (real-time) signal from one object or actor to another;
 - ✓ **Call events** indicate the receipt of a call for an operation by an object or actor;
 - ✓ **Time events** mark the passage of a designated period of time, e.g., after(10 seconds)

© 2001 Jaelson Castro and John Mylopoulos State and Activity Diagrams -- 4

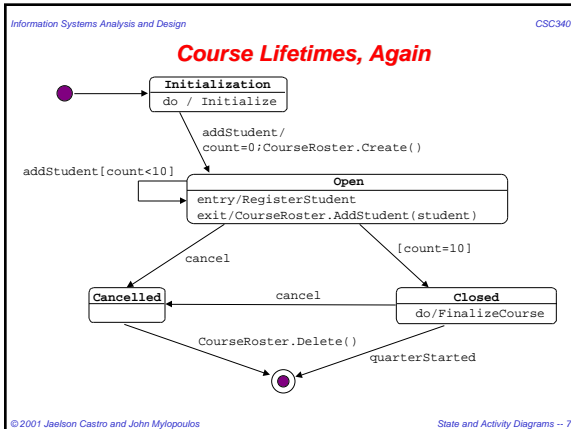


Information Systems Analysis and Design CSC340

States

- A state represents a time period during which
 - ✓ A predicate is true, e.g., budget - expenses > 0,
 - ✓ An action is being performed, e.g., check inventory for order items, or
 - ✓ Someone waits for an event to happen, e.g., arrival of a missing order item.
- A state can be "on" or "off".
- When a state is "on", all its outgoing transitions are eligible to fire. For a transition to fire, its event must occur and its condition must be true. When a transition does fire, its action is carried out.
- States can have associated activities. Special activity constructs include:
 - ✓ **do/stateDiagramName(parameterList)** -- "calls" another state diagram;
 - ✓ **entry/action** -- carry out the action when entering the activity;
 - ✓ **exit/action** -- carry out the action when exiting;

© 2001 Jaelson Castro and John Mylopoulos State and Activity Diagrams -- 6



Information Systems Analysis and Design CSC340

Superstates

- State transition diagrams can be very hard to read once they grow to more than a few dozen states.
- For UML state diagrams, states can be composed into nested states, or **superstates**. Such compositions make it possible to view a state diagram at different levels of abstraction.
- A superstate consists of one or more states.
- When an **AND superstate** is "on", all its states are also "on".
- When an **OR superstate** is "on", one of its states is "on".

© 2001 Jaelson Castro and John Mylopoulos State and Activity Diagrams -- 8

