

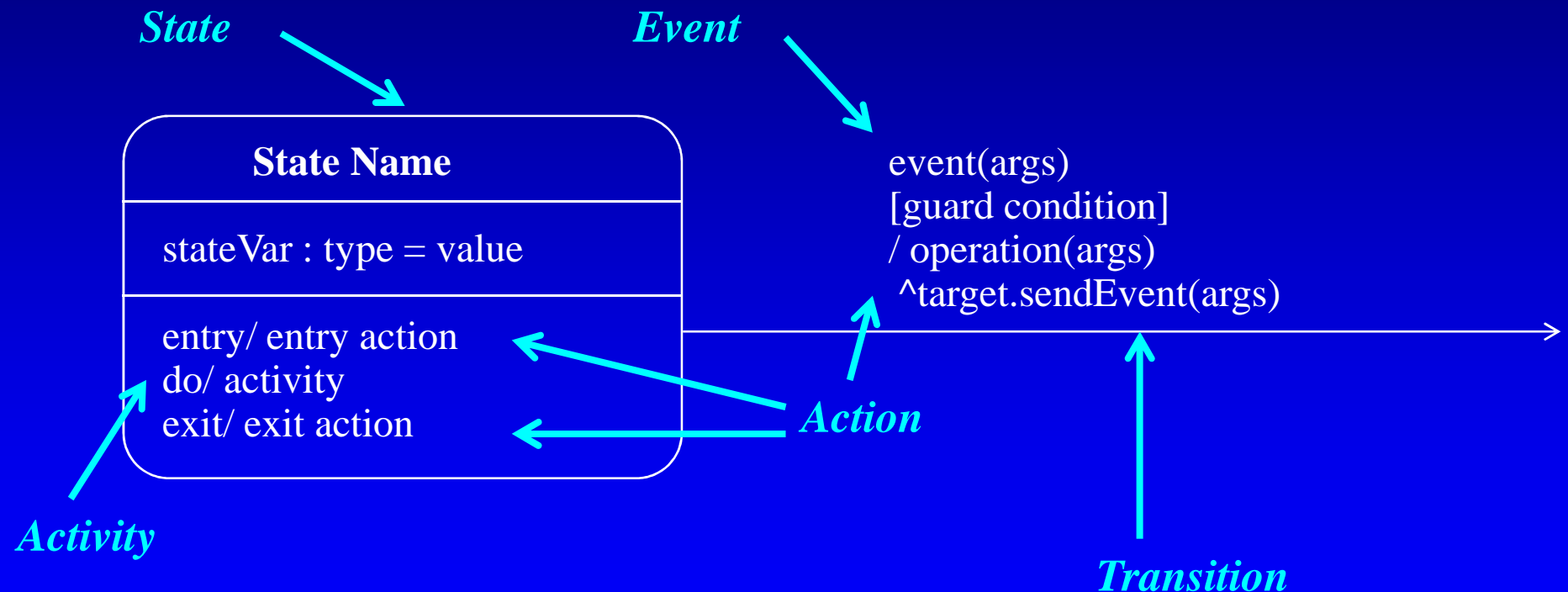
# Định nghĩa các trạng thái

---

- ◆ Mục đích
  - Thiết kế ảnh hưởng của trạng thái đối tượng lên hành vi của nó
  - Phát triển statecharts để mô hình các hành vi này
- ◆ Những gì cần xem xét:
  - Những object nào có trạng thái đáng kể?
  - Cách xác định các trạng thái của một object?
  - Cách ánh xạ statechart với phần còn lại của mô hình?

# Statechart là gì?

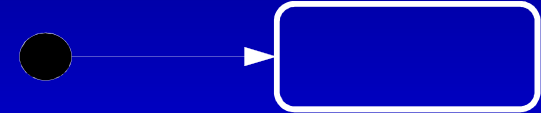
- ♦ Là 1 đồ thị có hướng với các node là các trạng thái nối với nhau bởi các transition
- ♦ Mô tả lịch sử đời sống của đối tượng



# Các trạng thái đặc biệt

- ♦ Trạng thái bắt đầu (Initial state)
  - Là trạng thái khi mới được khởi tạo của object
  - Mang tính bắt buộc
  - Chỉ có thể có 1 initial state
- ♦ Trạng thái kết thúc (Final state)
  - Chỉ vị trí kết thúc đời sống của object
  - Optional
  - Có thể có nhiều

Initial state



Final state



# Qui trình suy dẫn ra Statecharts

---

- ◆ Xác định và định nghĩa các trạng thái
- ◆ Xác định các event
- ◆ Xác định các transition (hồi đáp lại các event)
- ◆ Thêm các activity và các action

# Xác định và định nghĩa các trạng thái

- ♦ Significant, dynamic attributes

Số sinh viên tối đa trong 1 lớp là 100

`numStudents < 100`

Open

`numStudents >= 100`

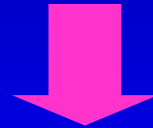
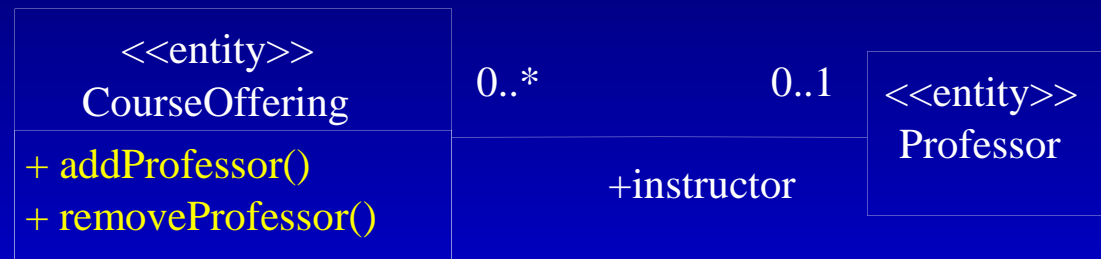
Closed

- ♦ Sự tồn tại và không tồn tại của các link



# Xác định các Event

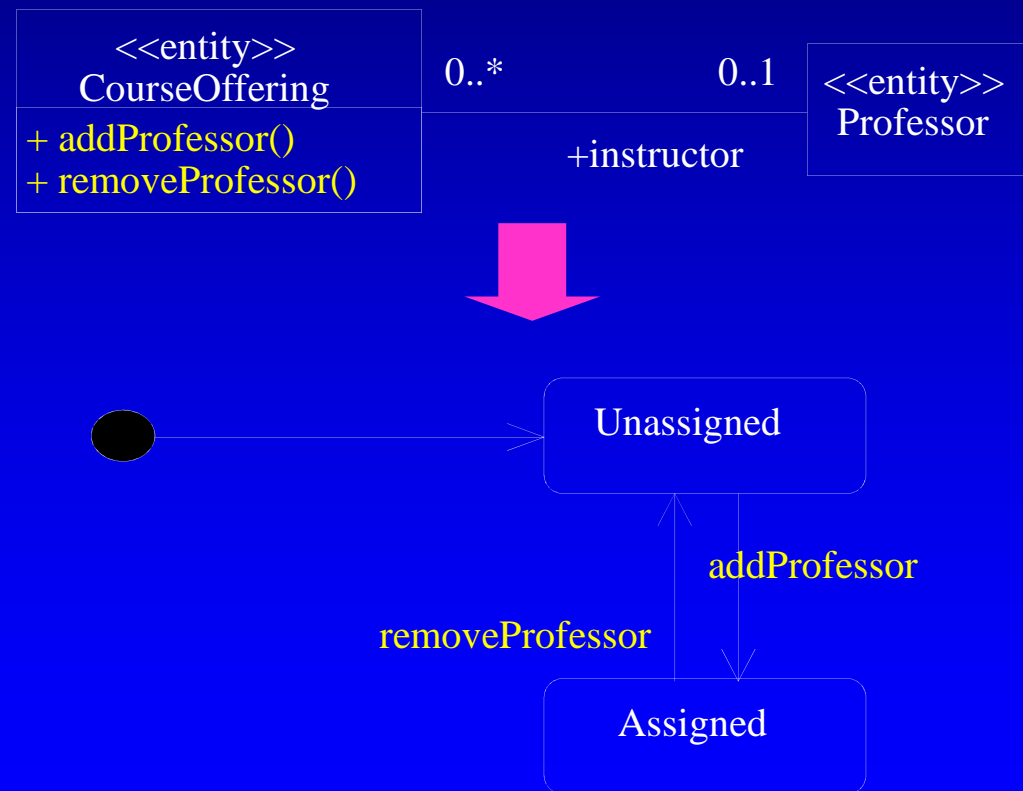
- ♦ Xem xét các class interface operation



Events: addProfessor, removeProfessor

# Xác định các Transition

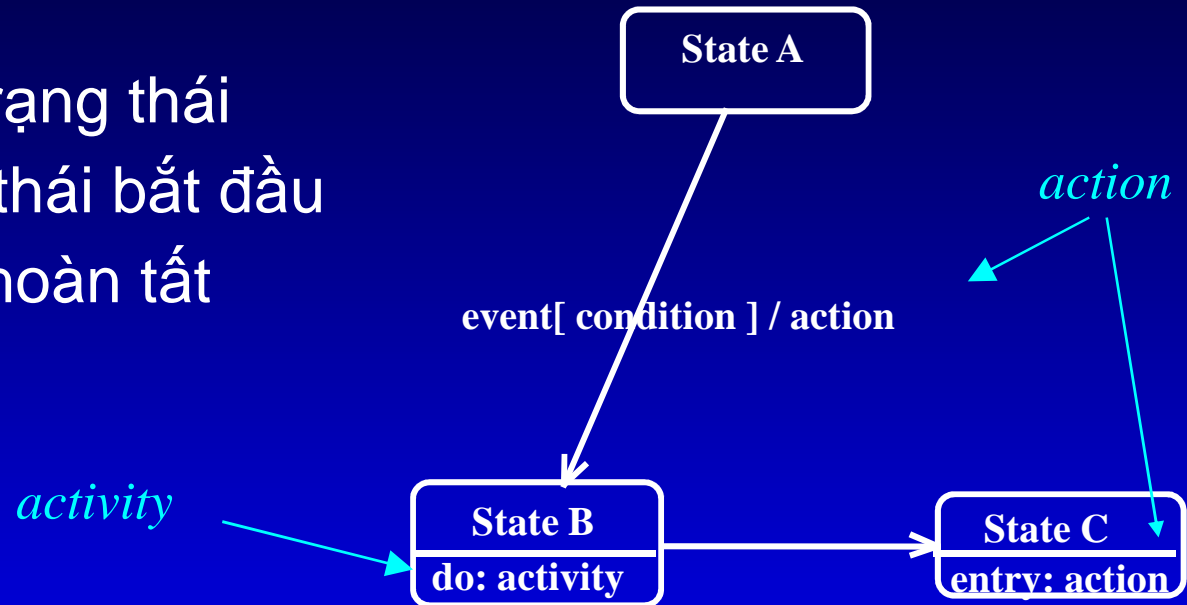
- ♦ Với mỗi trạng thái, xác định events nào gây ra transitions đến trạng thái nào, bao gồm các điều kiện kiểm soát, nếu cần
- ♦ Transitions mô tả điều gì xảy ra khi đối tượng hồi đáp lại một event nhận được



# Thêm Activities và Actions

## ◆ Activities

- Kết hợp với một trạng thái
- Bắt đầu khi trạng thái bắt đầu
- Cần thời gian để hoàn tất
- Có thể ngắt



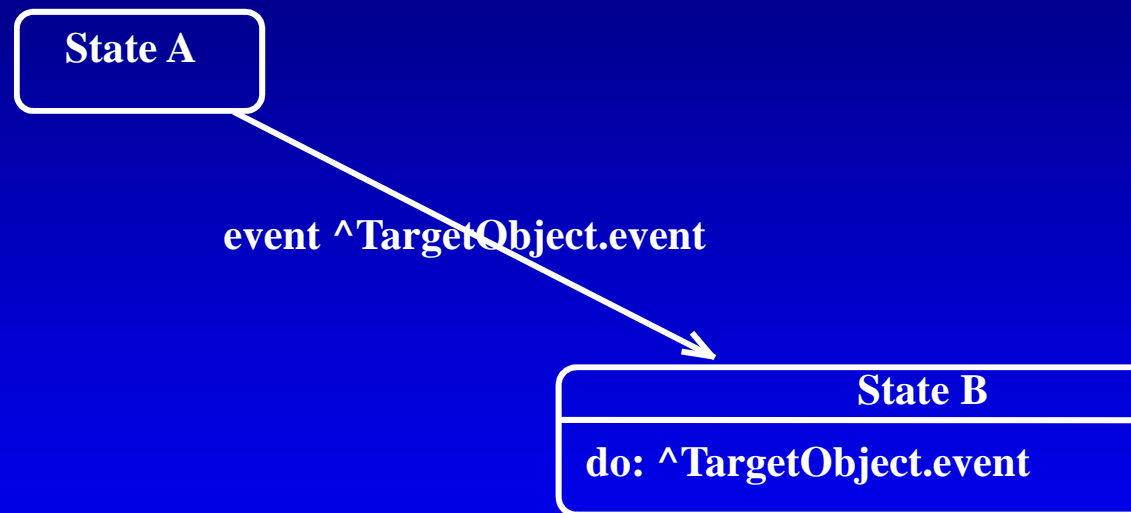
## ◆ Actions

- Kết hợp với 1 transition
- Cần thời gian không đáng kể để hoàn tất
- Không thể ngắt ngang



# Gửi Events

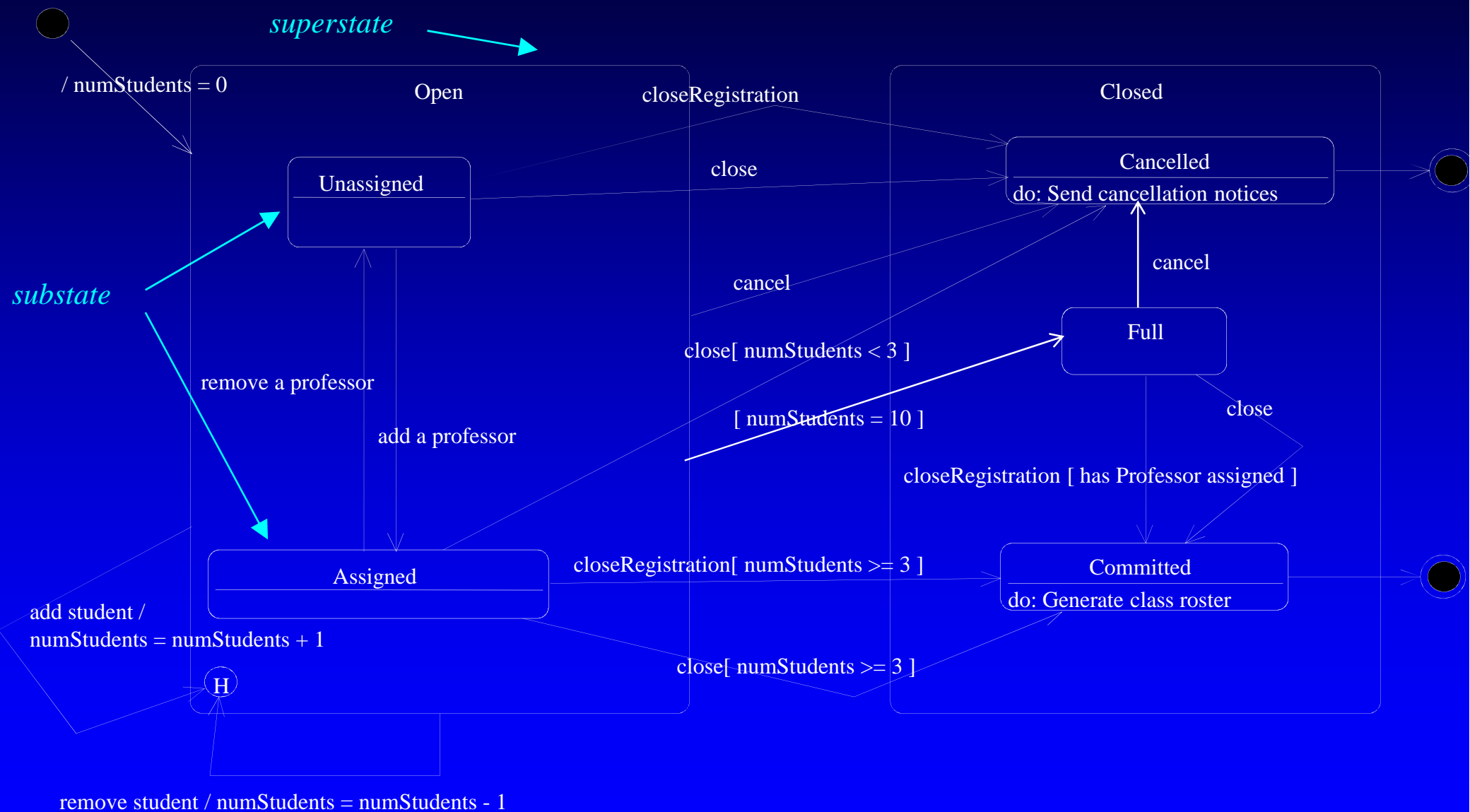
- ♦ Một event có thể gây ra việc gửi một event khác
- ♦ Một activity cùng có thể gửi event đến object khác



# Ví dụ: Statechart



# Ví dụ: Statechart với các trạng thái lồng nhau



# Những Object có Significant State?

---

- ◆ Các Object có vai trò thể hiện rõ bởi state transitions
- ◆ Các use case phức tạp là state-controlled
- ◆ Không cần mô hình hóa tất cả các object
  - Các Object dễ dàng cài đặt
  - Các Object không thuộc loại state-controlled
  - Các Object chỉ với một trạng thái

# Cách Statecharts gắn với phần còn lại?

- ◆ Các Event biến thành các operation
- ◆ Các Method phải được cập nhật với các thông tin đặc thù cho các trạng thái
- ◆ Các trạng thái được biểu diễn bởi attributes
  - Chúng là input cho bước định nghĩa Attribute

