



UNIFIED MODELING LANGUAGE™

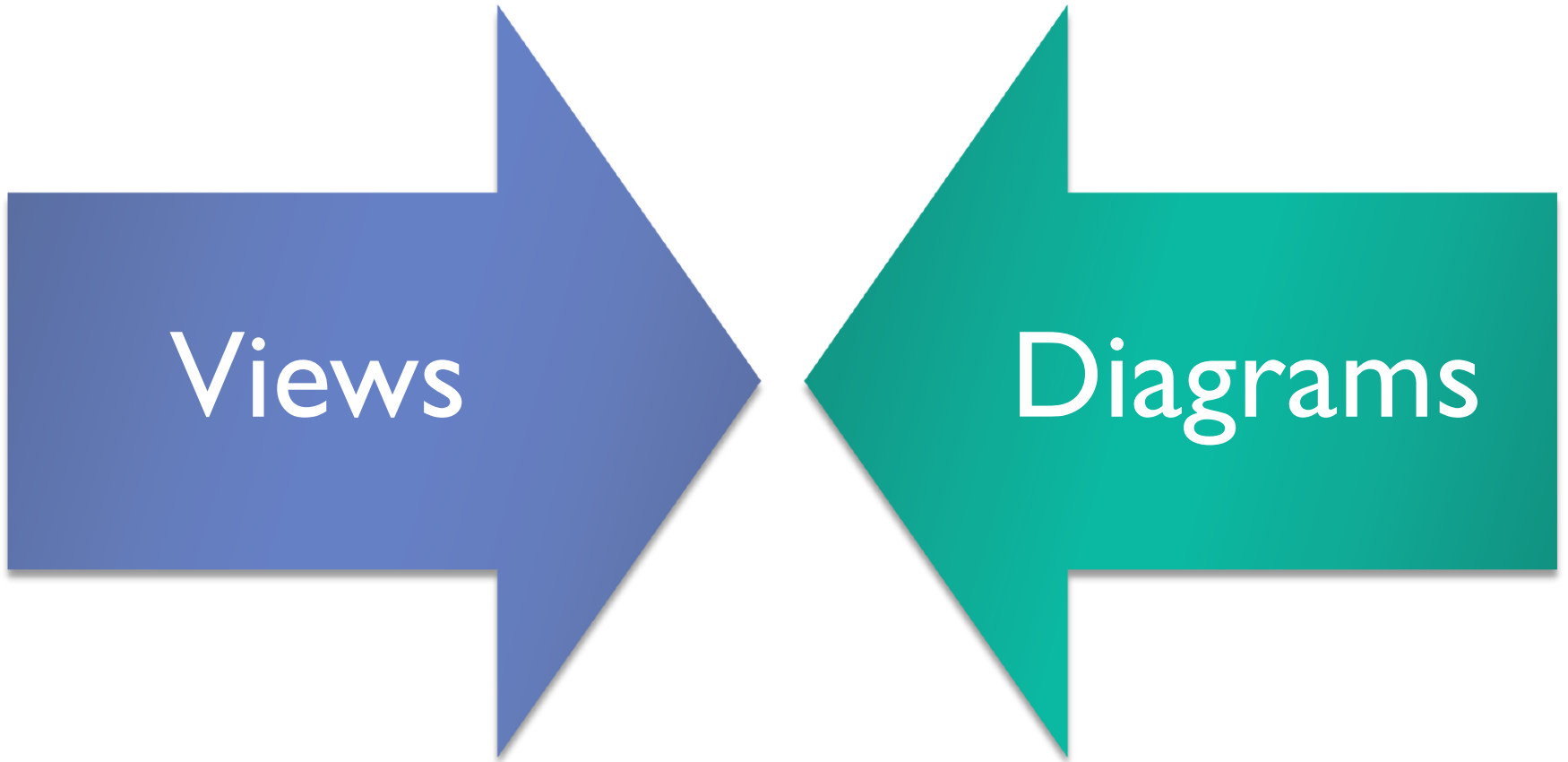


14.

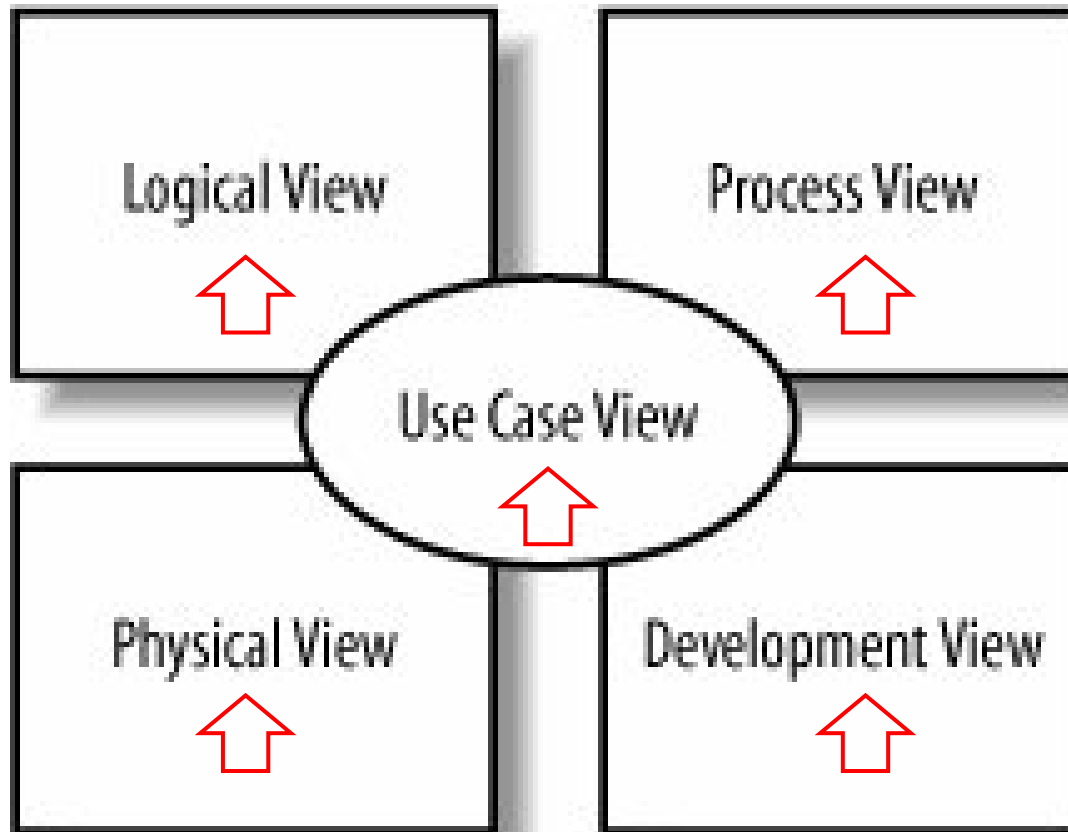
Modeling an Object's State: State Machine Diagrams

Shaoning Zeng, <http://zsn.cc>

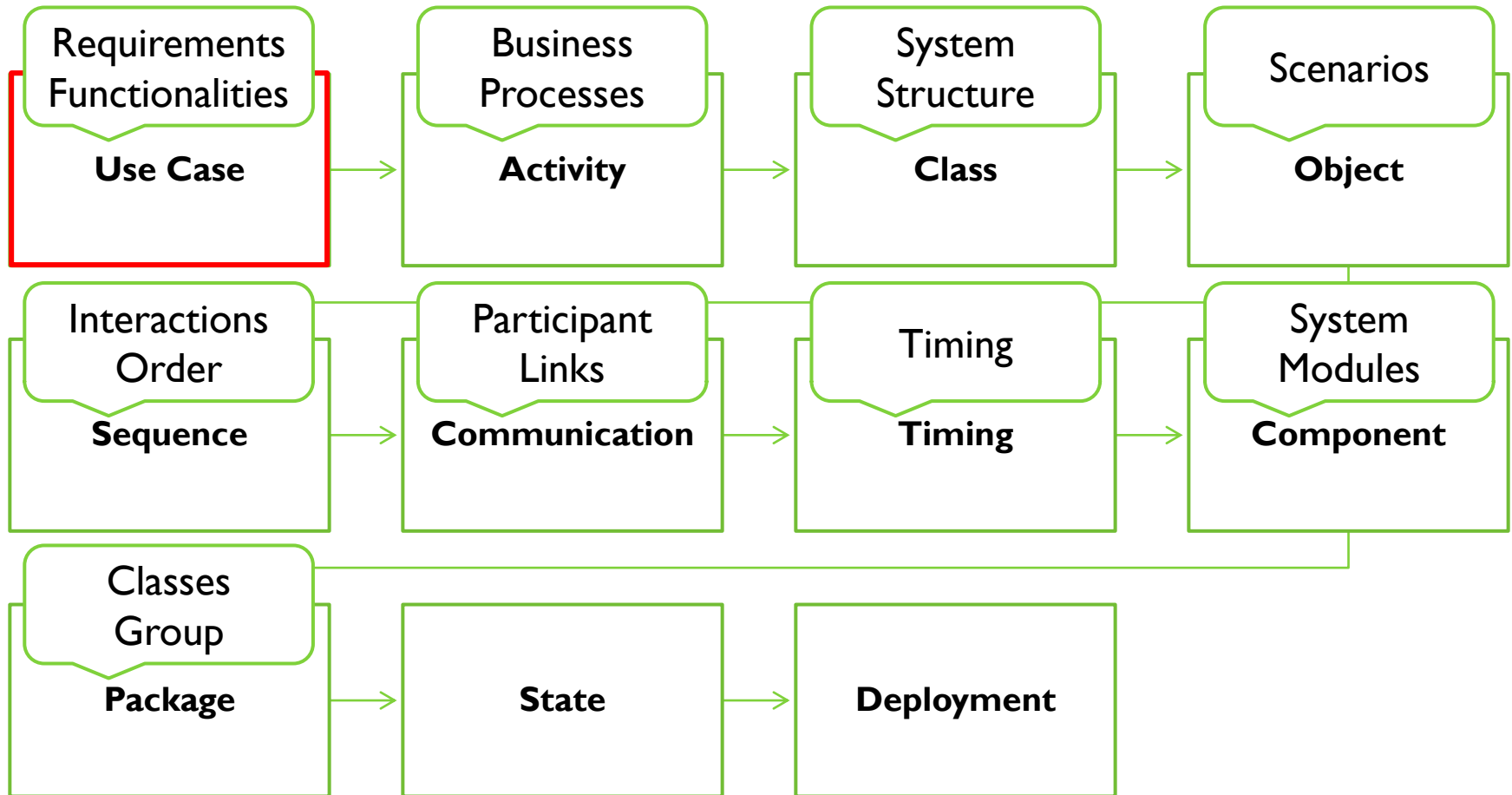
Modeling Language



Views of Your Model



Roadmap



Diagrams in each view

Use case View

- User case diagrams

Process View

- Activity diagrams

Logical View

- Class diagrams
- Object diagrams
- Sequence diagrams
- Communication diagrams
- Timing diagrams
- State machine diagrams

Development View

- Component diagrams
- Package diagrams

Physical View

- Deployment diagrams
-

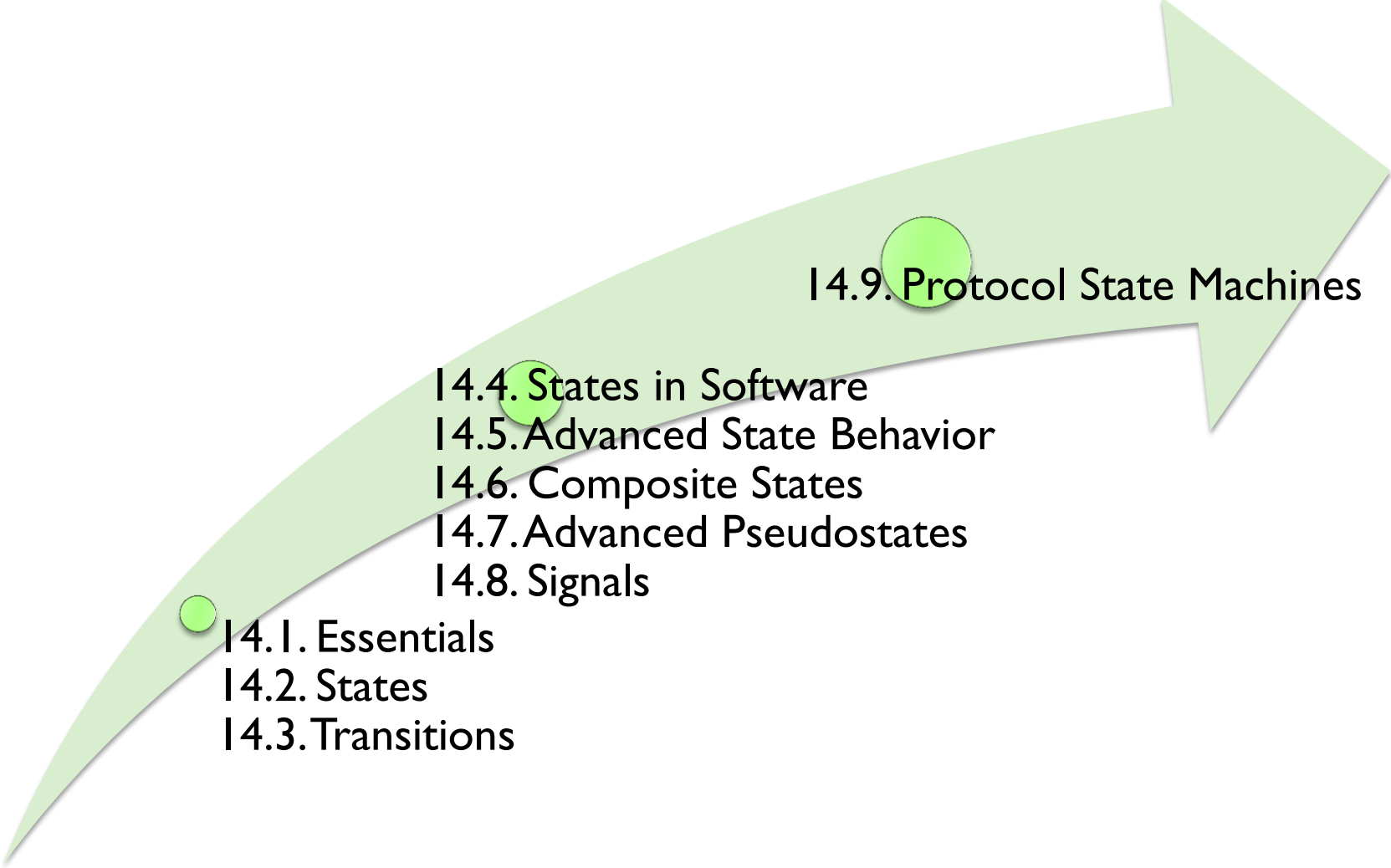


Why using State Machine Diagrams?

- ▶ Sometimes **the state of an object** or system is an important factor in its behavior. 一个对象的状态
 - ▶ AccountApplication object in CMS: approved or rejected
- ▶ Some typical usages: 常见使用情况
 - ▶ Real-time/mission-critical systems, such as heart monitoring software 实时系统，如心跳监控软件
 - ▶ Dedicated devices whose behavior is defined in terms of state, such as ATMs 专用设备，如ATM
 - ▶ First-person shooter games, such as Doom or Half-Life 第一人称射击游戏



14. Modeling an Object's State: State Machine Diagrams 状态图



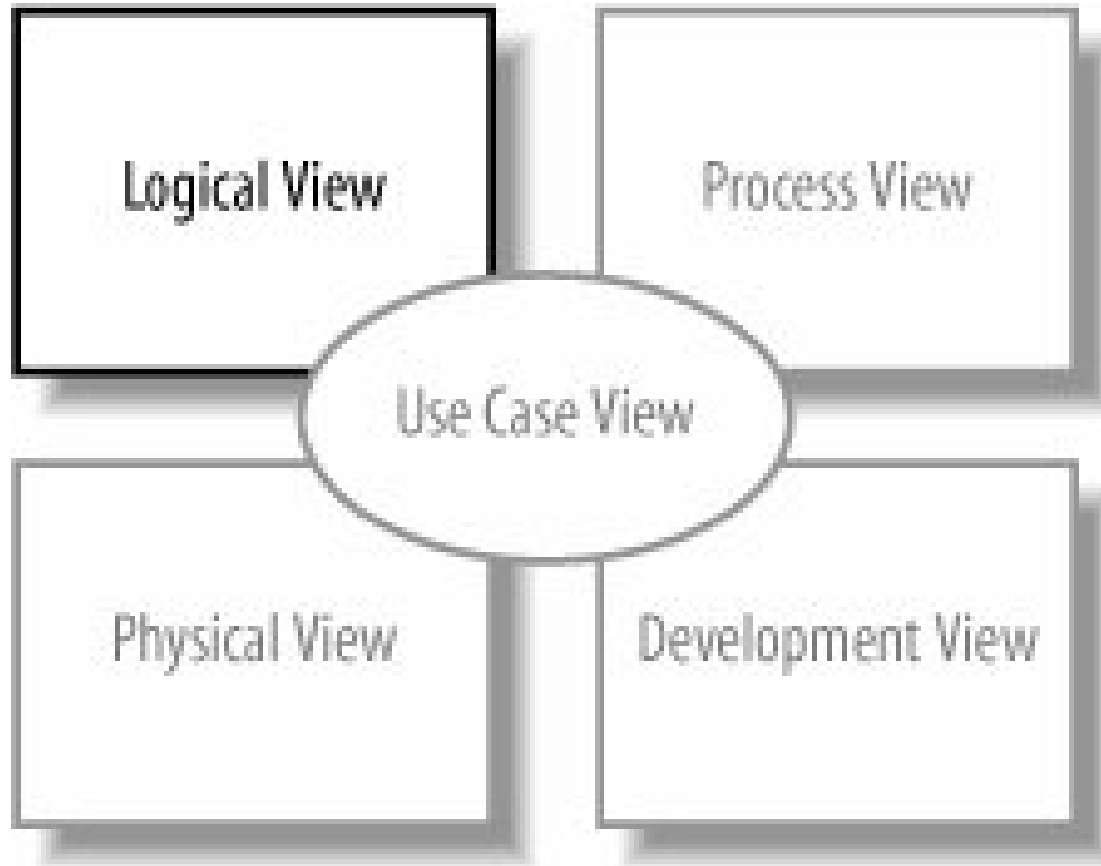
14.9. Protocol State Machines

14.4. States in Software
14.5. Advanced State Behavior
14.6. Composite States
14.7. Advanced Pseudostates
14.8. Signals

14.1. Essentials
14.2. States
14.3. Transitions



State machine diagrams are part of the logical model of your system 逻辑视图



14.1. Essentials 组成元素

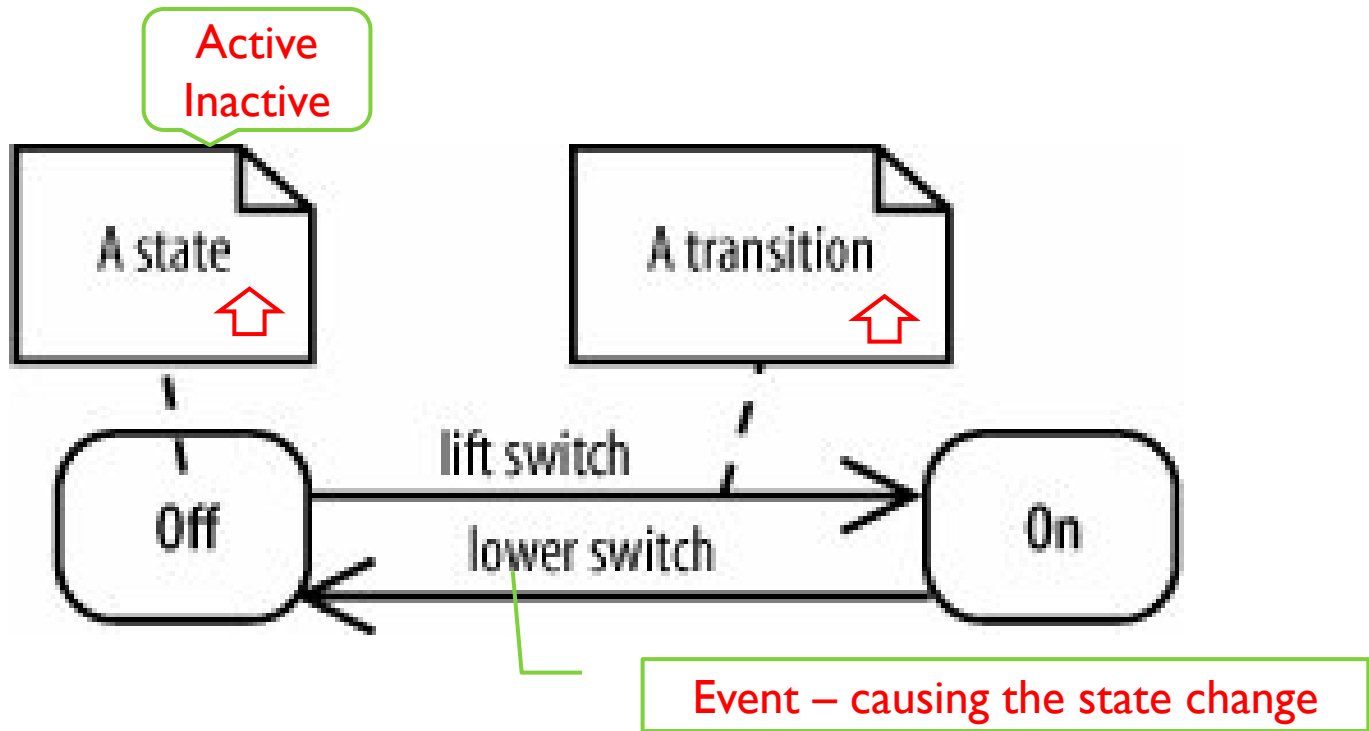
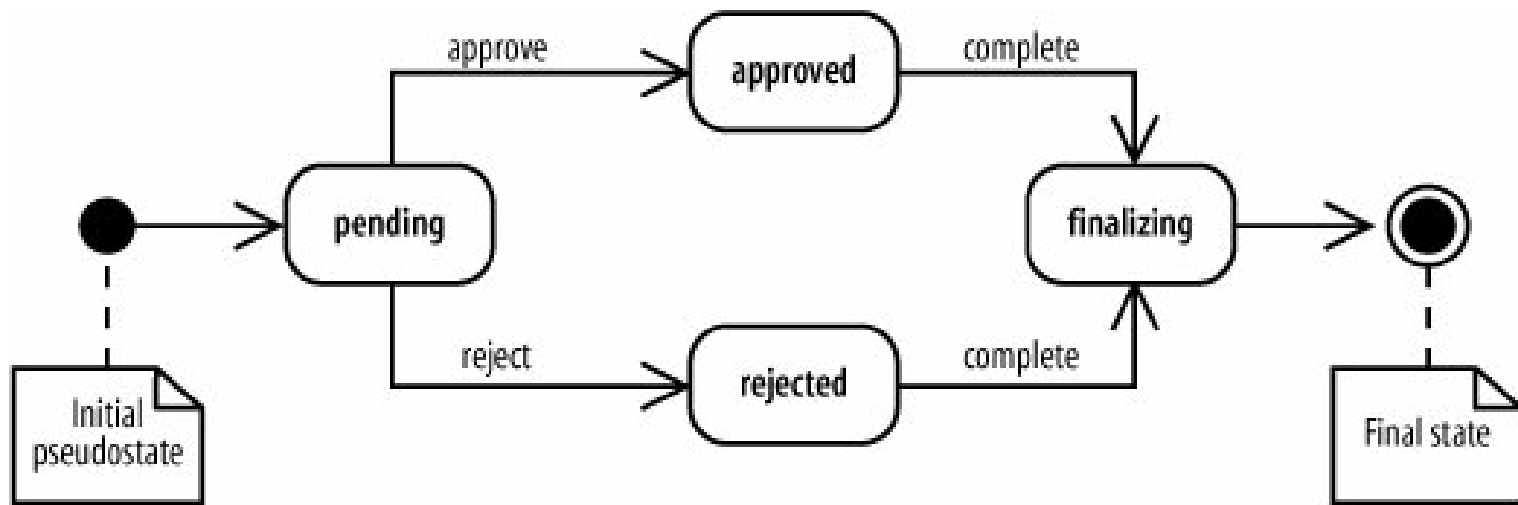


Figure 14-2. The fundamental elements of a state diagram: states and transitions between states 状态+转变

Figure 14-3. Initial pseudostate and final states in an AccountApplication state diagram 初始伪状态, 结束状态



Pseudostates are **special markers** that direct the flow of traffic in a state diagram.



14.2. States 状态

- ▶ A state is a **condition** of being at a certain time.
- ▶ A state can be a **passive quality**, 被动性质
 - ▶ such as On and Off for the light object. 电灯的开关
- ▶ A state can also be an **active quality**, or something that an object is **doing**. 主动性质
 - ▶ For example, a coffeemaker has the state Brewing during which it is brewing coffee. 咖啡机 - 煮咖啡
- ▶ A state is drawn as a rounded rectangle[‘rek,tæŋɡl] with the name of the state in the center, as shown in [Figure 14-4](#). 圆角矩形





Figure 14-4. A rectangle with rounded corners and the name in the center is the most common way to draw a state **状态名**

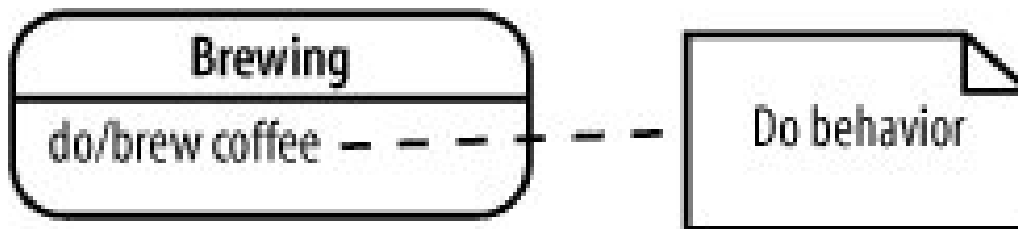


Figure 14-5. Showing the behavior details of a “doing” state **正在做什么**



14.3. Transitions 转变

- ▶ A transition, shown with an arrow, represents a change of states from a **source state** to a **target state**.
- ▶ A **transition description**, written along the arrow, describes the circumstances causing the state change to occur. 转变的描述

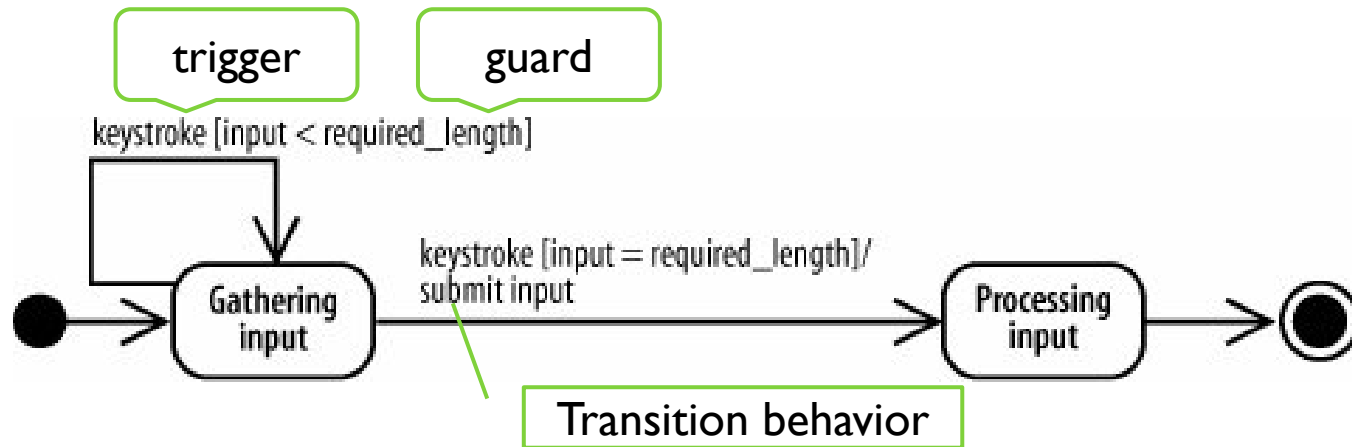


Figure 14-6. This input processing state diagram models features a trigger, guard, and transition behavior along one of its transitions

Figure 14-7. CD player state diagram, featuring a variety of transition descriptions

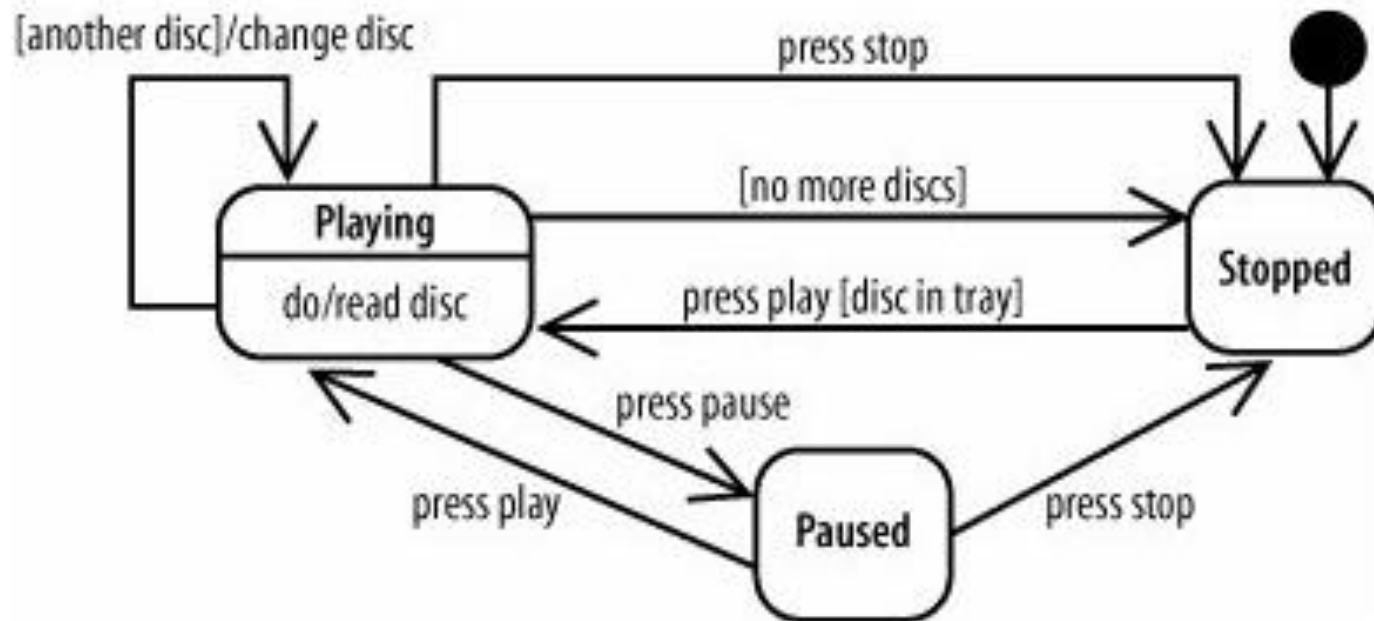


Figure 14-8. The most common type of transition features only a trigger

常见的转变就是一个触发器

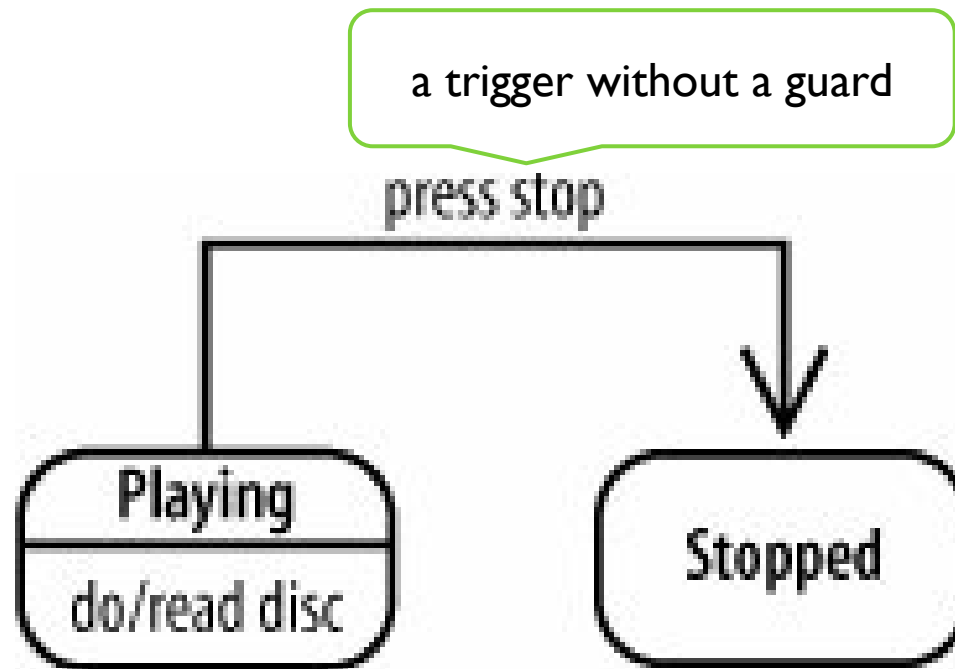


Figure 14-9. A guard will **block** a transition if it evaluates to **false**

守护条件为 false，则阻塞转变

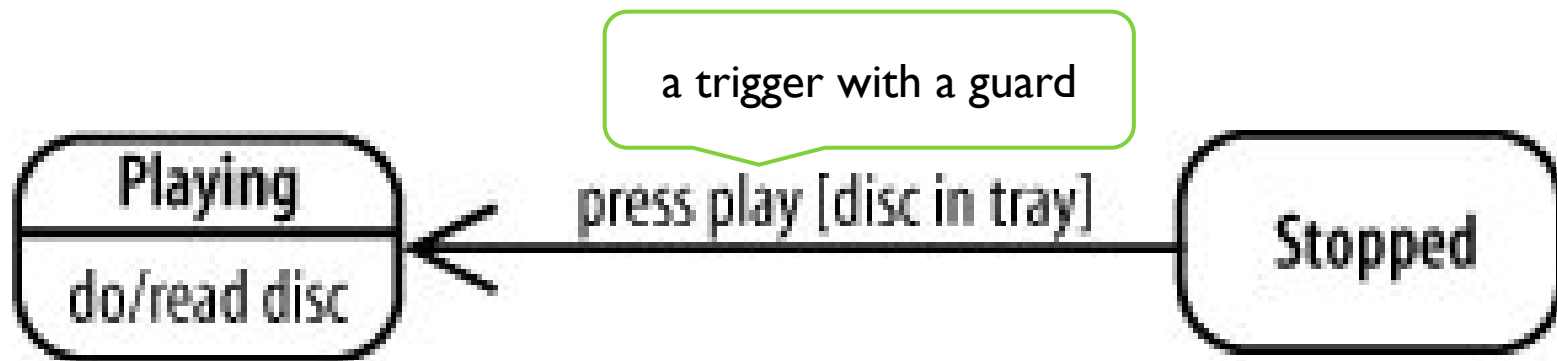


Figure 14-10. In this example, a transition is caused by the completion of **internal behavior**

内部行为触发转变

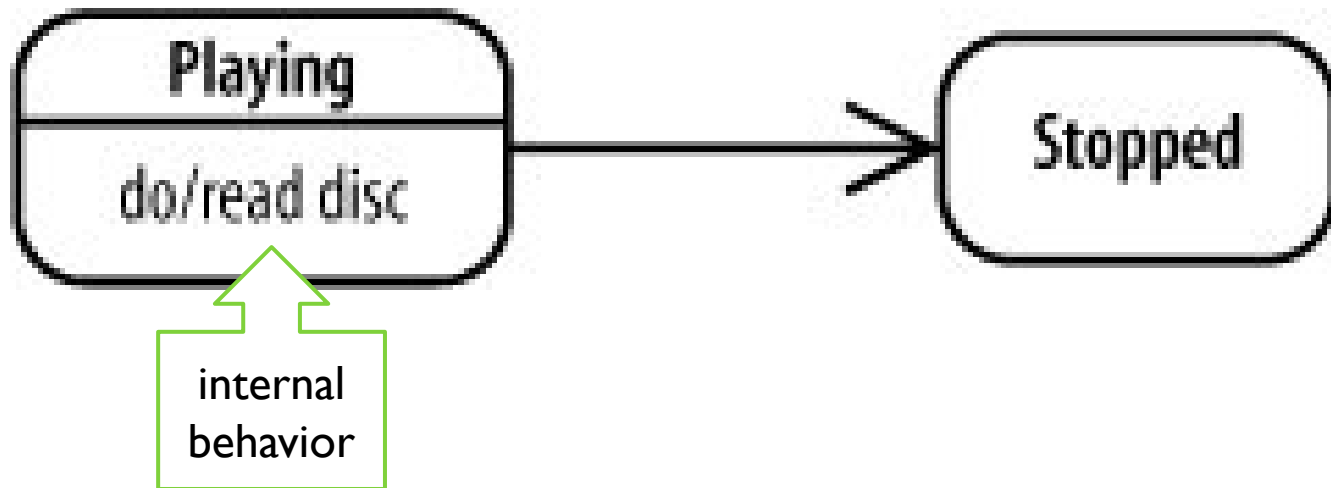
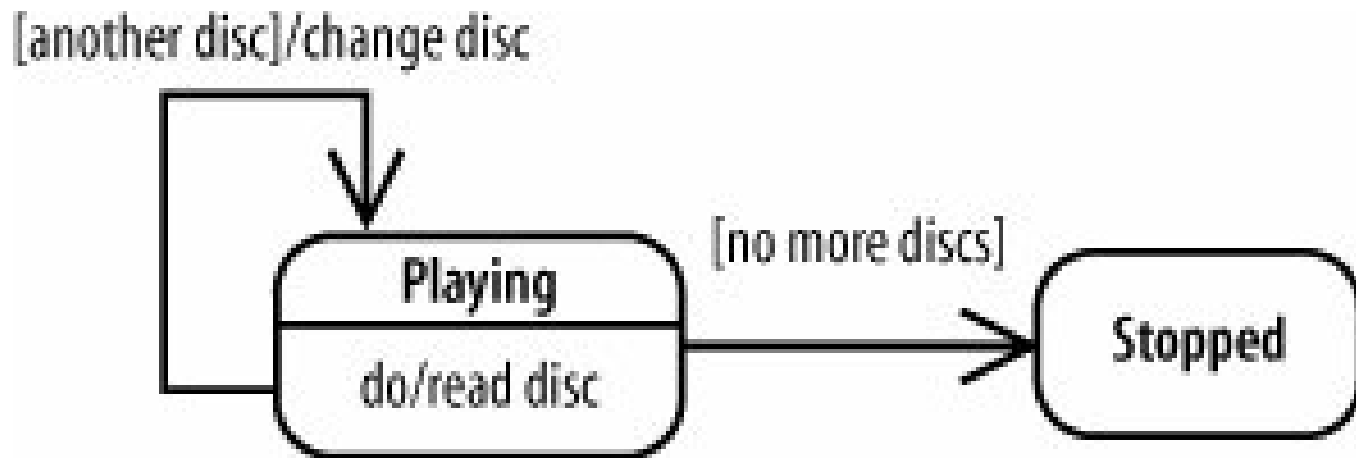


Figure 14-11. Using guards to model a choice between paths

使用守护条件描述路径选择



14.4. States in Software 软件的状态

- ▶ In software, state diagrams model an **object's life cycle**, or the states it goes through during its lifespan.
状态图描述对象生命周期

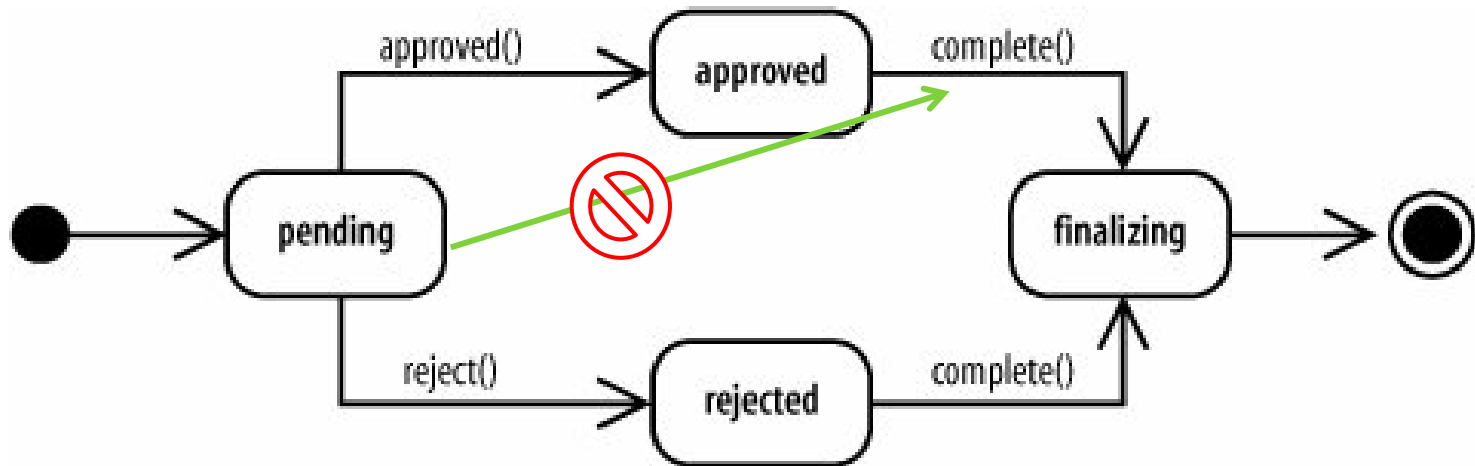
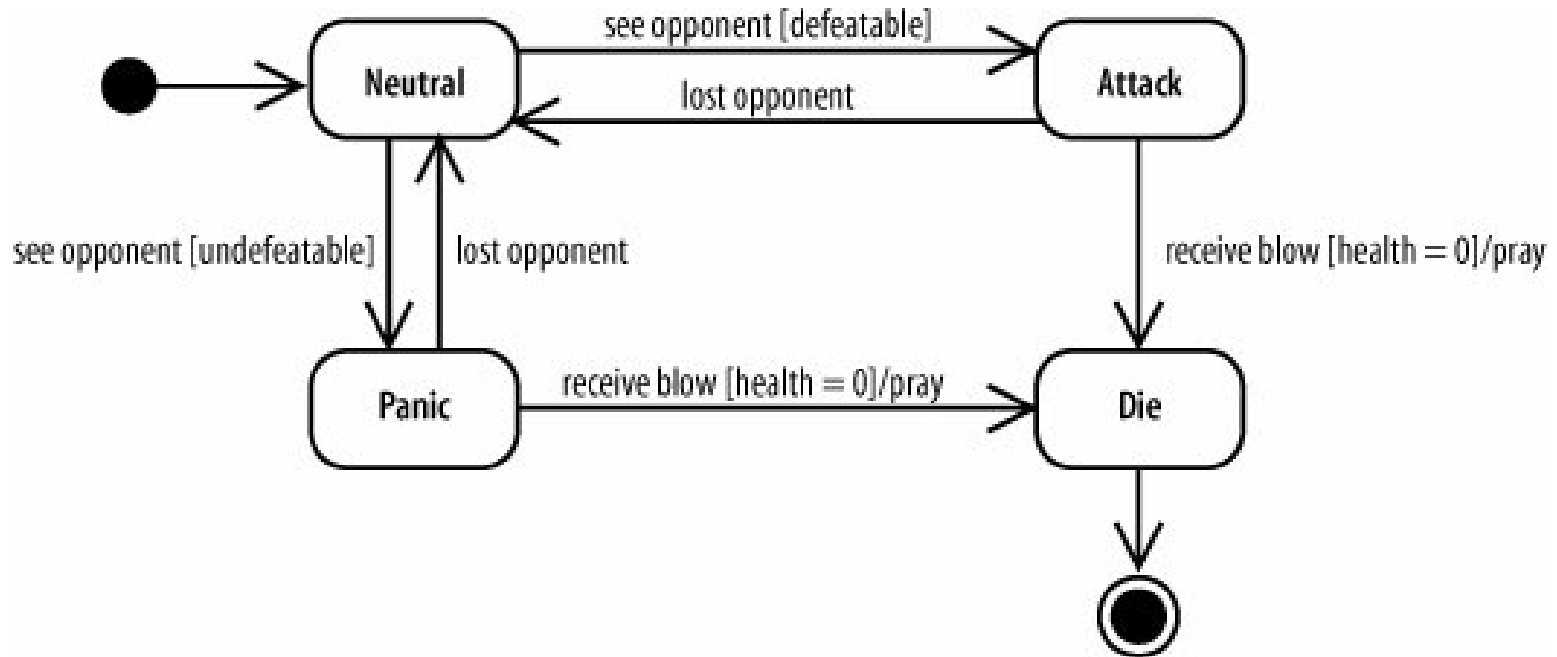


Figure 14-12. The life cycle of an AccountApplication object

Figure 14-13. State diagram modeling a troll in a FPS game; the troll's behavior is determined by his state

第一人称游戏的角色



14.5. Advanced State Behavior 高级状态行为

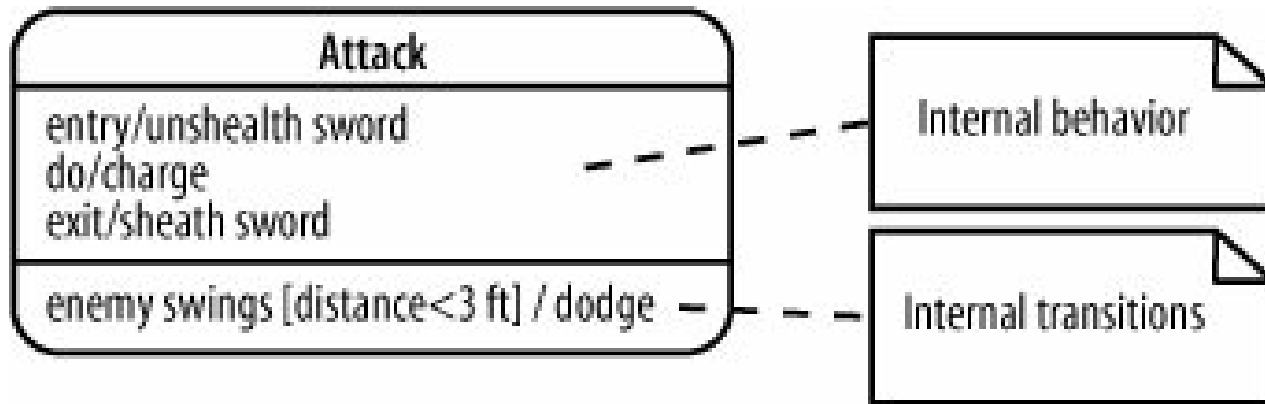


Figure 14-14. Internal behavior and transitions of the Attack state



14.5.1. Internal Behavior 内部行为

- ▶ Internal behavior is any behavior that happens while the object is in a state. 对象在某个状态下的行为
- ▶ Internal behavior is written as **label / behavior**.
 - ▶ do, entry, exit
- ▶ Unlike **do** behavior, **entry** and **exit** behaviors can't be interrupted. 进入与退出行为不可中断

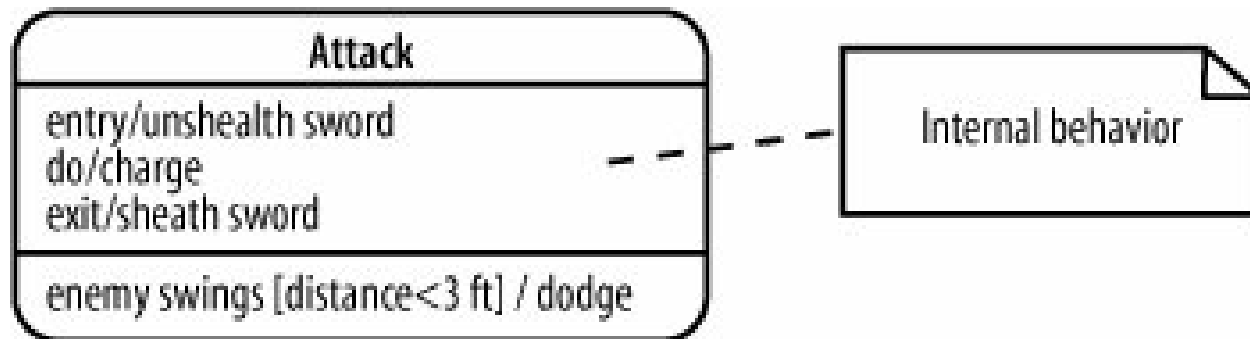
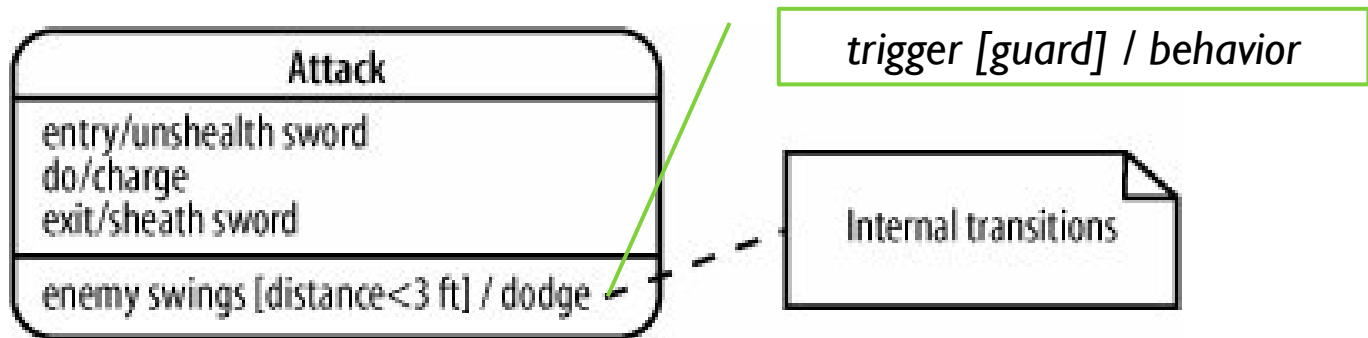


Figure 14-15. The middle compartment shows internal behavior



14.5.2. Internal Transitions 内部转变

- ▶ An internal transition is a transition that causes a **reaction** within a state, but doesn't cause the object to change states. 内部转变是某个状态下拉响应
- ▶ An internal transition is **different** from a self transition because self transitions cause entry and exit behavior to occur whereas internal transitions don't. 与自转变不同：会触发进入和退出行为



▶ Figure 14-16. The bottom compartment shows internal transitions

14.5.2. Internal Transitions (Cont.)

- ▶ Use internal transitions to **model reactions** to events that **don't** cause state changes.

描述不导致状态变化的事件响应

- ▶ For example, you could use internal transitions to show that a pause-and-serve coffee-maker suspends dispensing the coffee when you remove the coffee pot but doesn't leave the Brewing state,

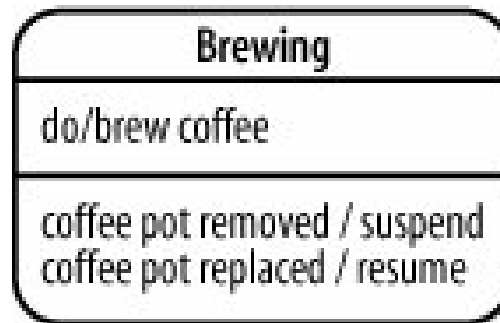


Figure 14-17. An internal transition models a reaction while staying in the same state



14.6. Composite States 组合状态

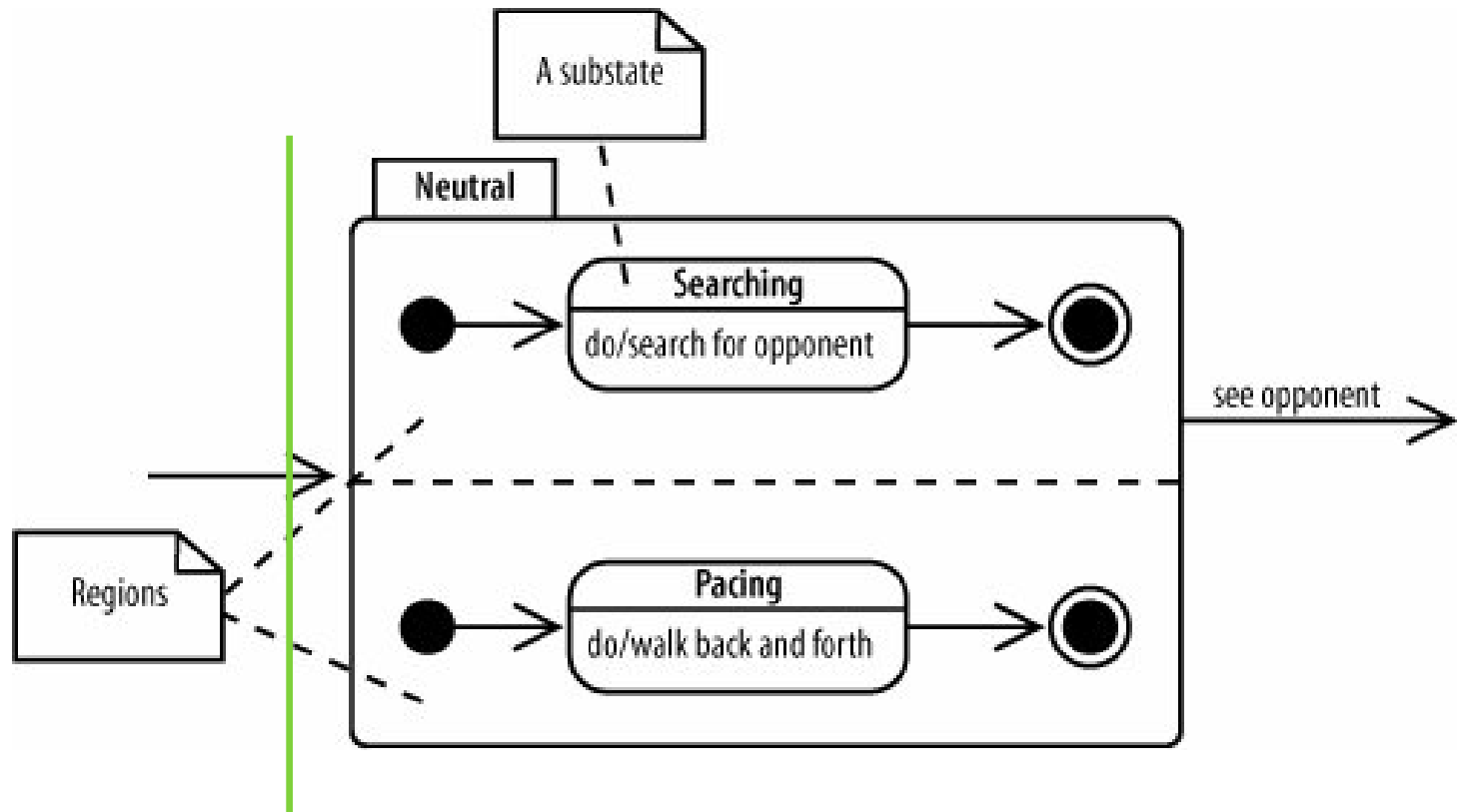


Figure 14-18. Composite states contain **one or more** state diagrams; if they contain more than one state diagram, then the state diagrams **execute in parallel**



14.7. Advanced Pseudostates 高级伪状态

- ▶ A **choice** pseudostate is used to emphasize that a Boolean condition determines which transition is followed. 决策伪状态
- ▶ A choice has **guards** on each of its outgoing transitions, and the transition that is followed depends on the guard.

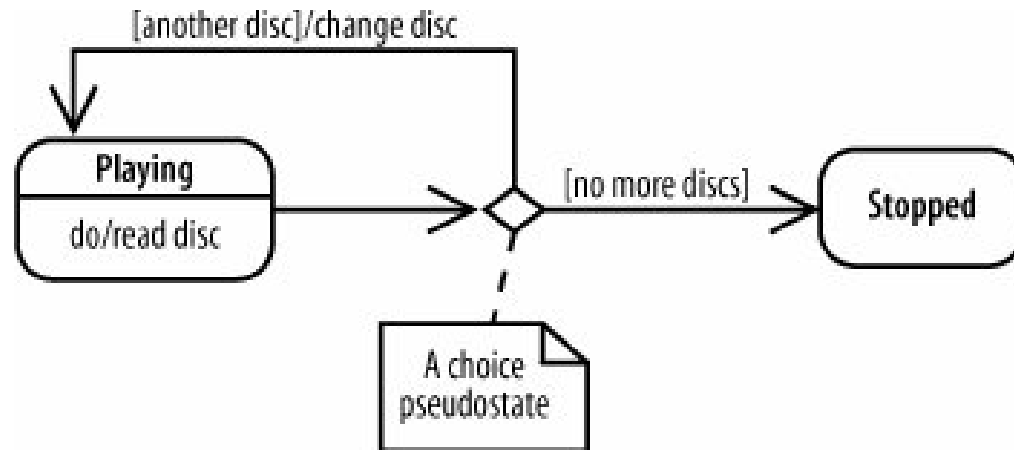


Figure 14-19. The path followed after a choice depends on the guard



14.7. Advanced Pseudostates

- ▶ **Fork** and **join** pseudostates show branching into concurrent states and then rejoining. 分叉与联合

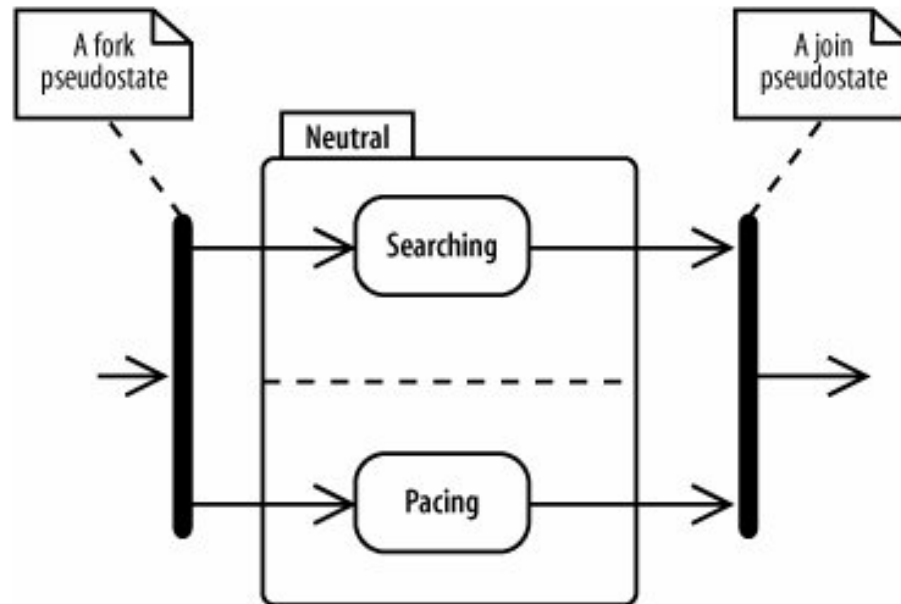


Figure 14-20. Forks and joins show concurrent states

14.8. Signals 信号

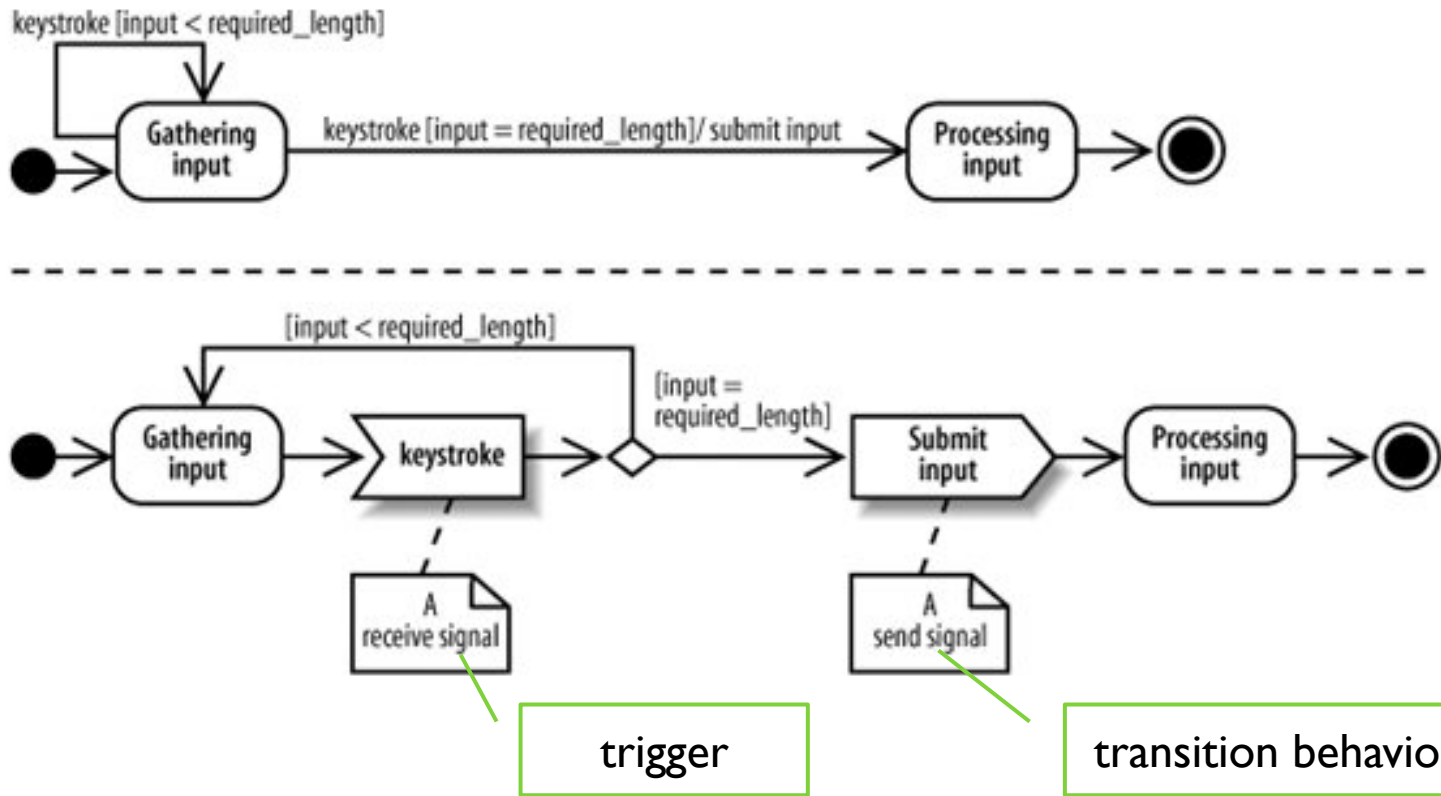


Figure 14-21. The bottom diagram draws transitions and transition behavior as receive and send signals

14.9. Protocol State Machines 协议状态机

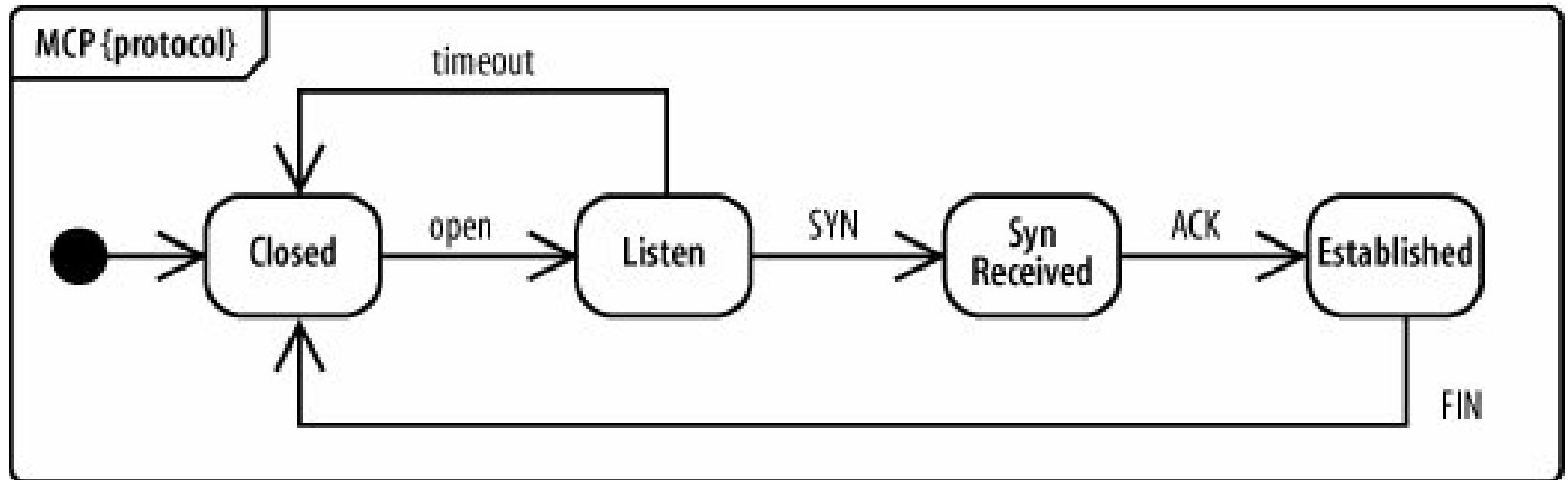


Figure 14-22. Protocol state machine modeling the receiver side of a simplified communication protocol called My Communication Protocol (MCP)



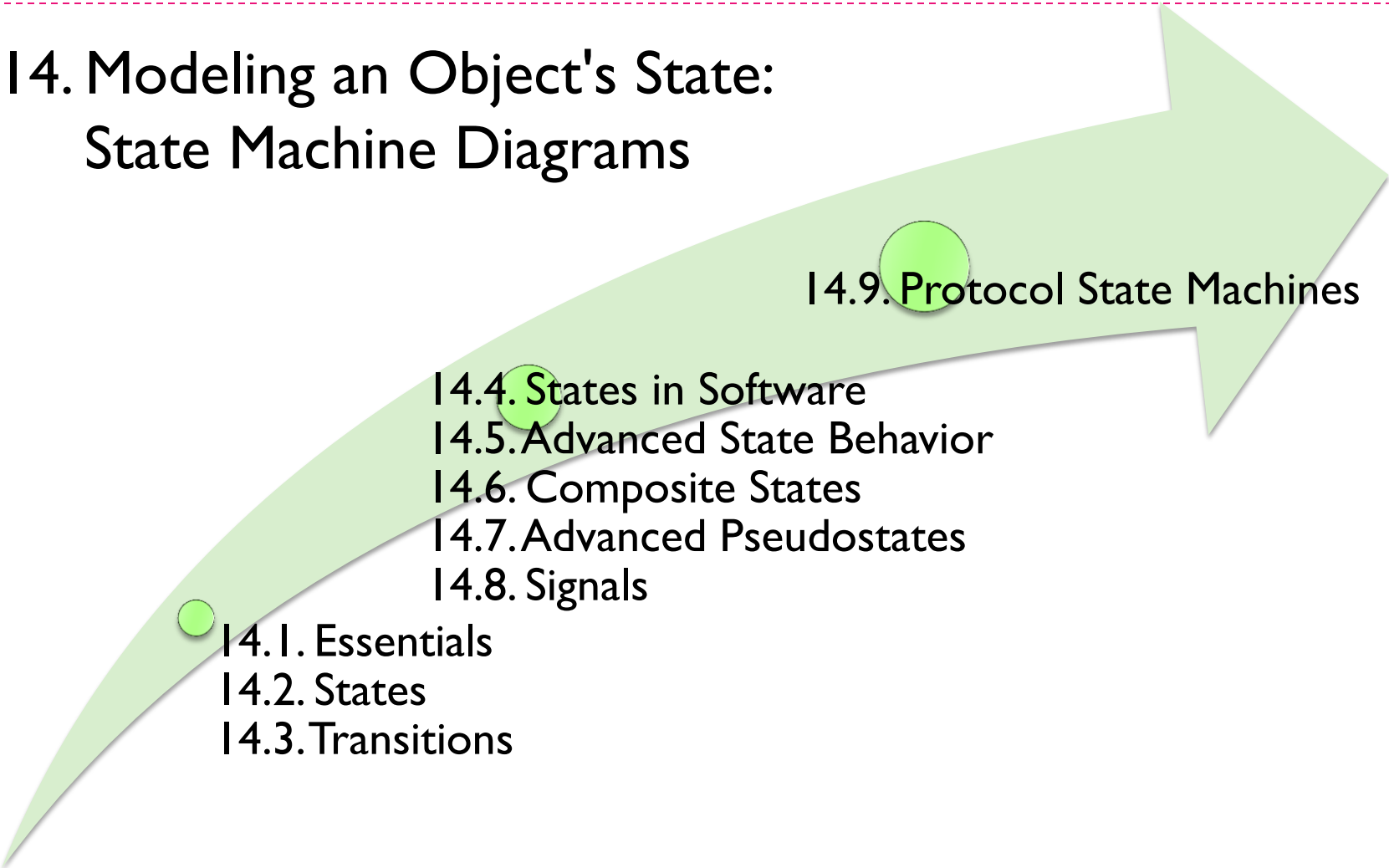
14.9. Protocol State Machines

- ▶ Protocol state machines are a **special kind of** state machine focusing on how a protocol, such as a communication protocol (e.g., TCP), works.
- ▶ The main **difference** between protocol state machines and behavioral state machines is that protocol state machines don't show behavior along transitions or inside states. Instead, they focus on showing a legal sequence of events and resulting states. 与行为状态机的区别
- ▶ Protocol state machines are drawn in a **tabbed** rectangle with the name of the state machine in the tab followed by **{protocol}**.



Summary

14. Modeling an Object's State: State Machine Diagrams



14.9. Protocol State Machines

14.4. States in Software
14.5. Advanced State Behavior
14.6. Composite States
14.7. Advanced Pseudostates
14.8. Signals

14.1. Essentials
14.2. States
14.3. Transitions



See you ...

