Probabilistic And Stochastic UML Statecharts

Speaker: Abdallah Salim

Tutor: Dr. David N. Jansen

Chair for Software Modeling and Verification
RWTH Aachen, Germany
SS 2007

Overview

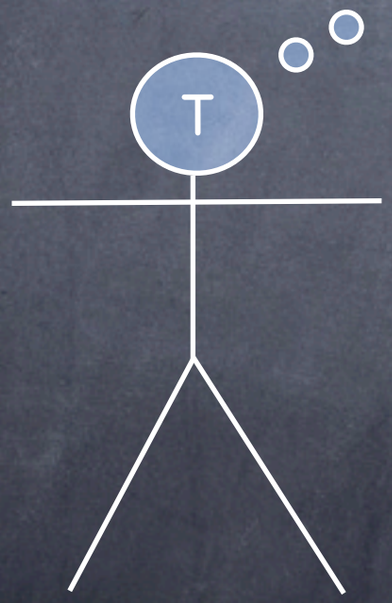
- An introduction to Statecharts
  - StoCharts
    - Syntax
    - Semantics
  - Case Study: A Beverages dispenser
  - Conclusion

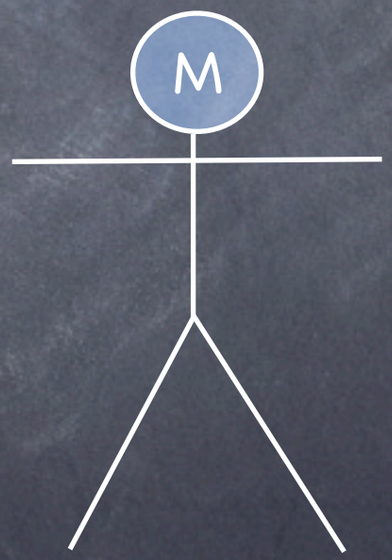
The UML perspective - A scenario

Name: Mr Troy aka Mr T
Age: 29
Position: Software Engineer

Name: Mr McNamara aka Mr M
Age: 27
Position: Visual designer

T seems to have an idea !



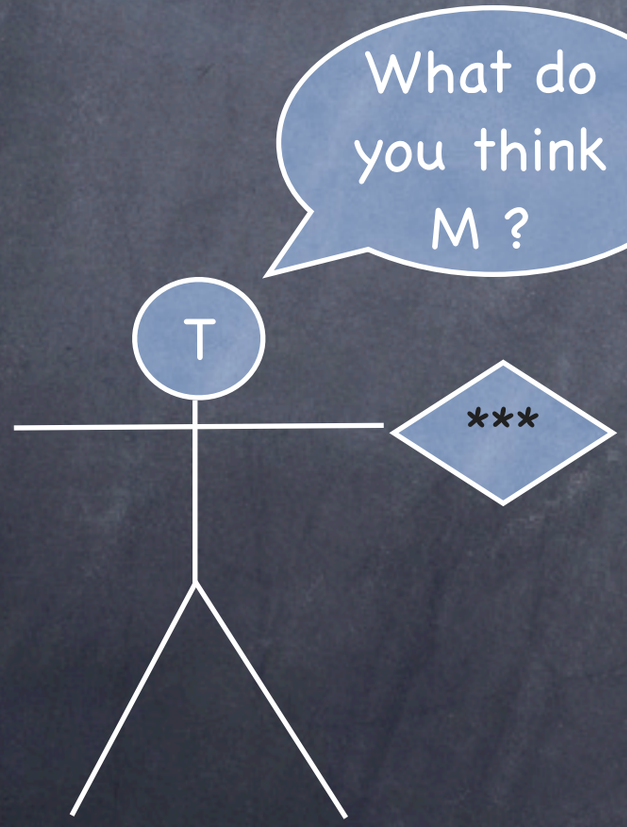


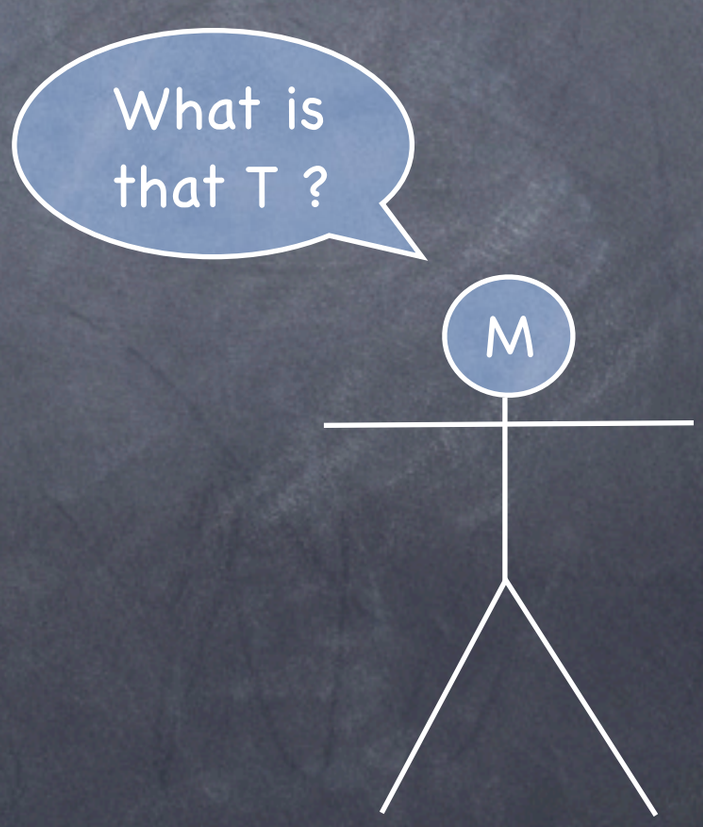
The UML perspective - Getting the message across

Mr. T has a great design idea

but ...

Mr. M doesn't seem to understand T's design



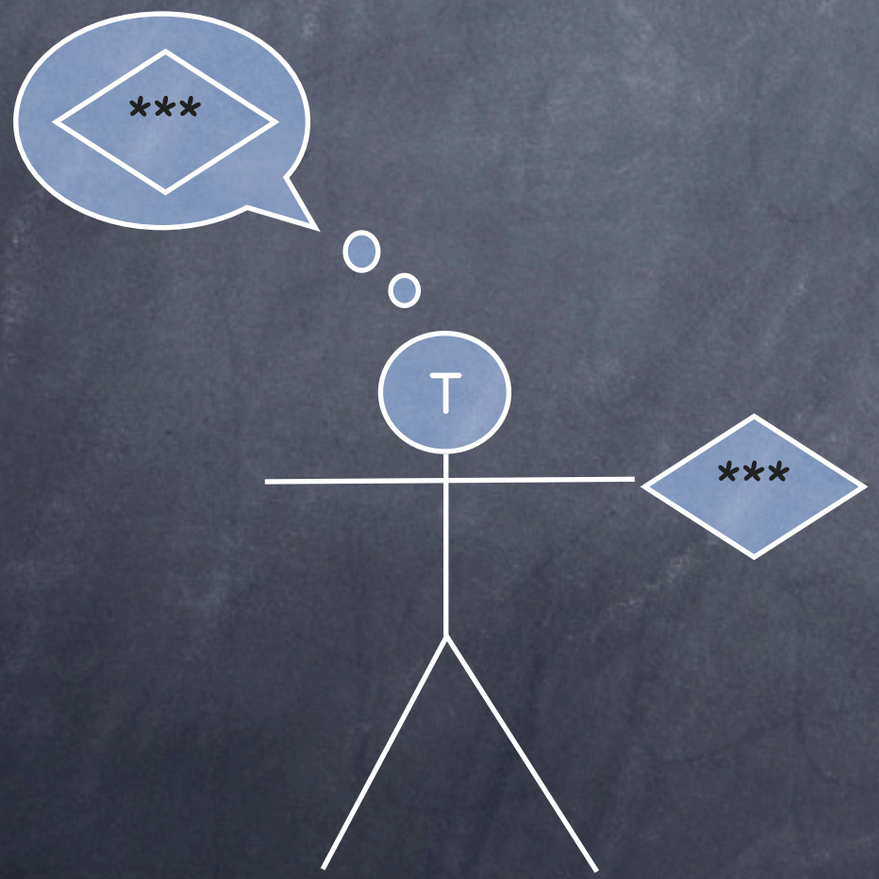


The UML perspective - The Problem

Mr. T has an idea in his head

but ...

Mr. M forms a completely different one in his head



This is ...

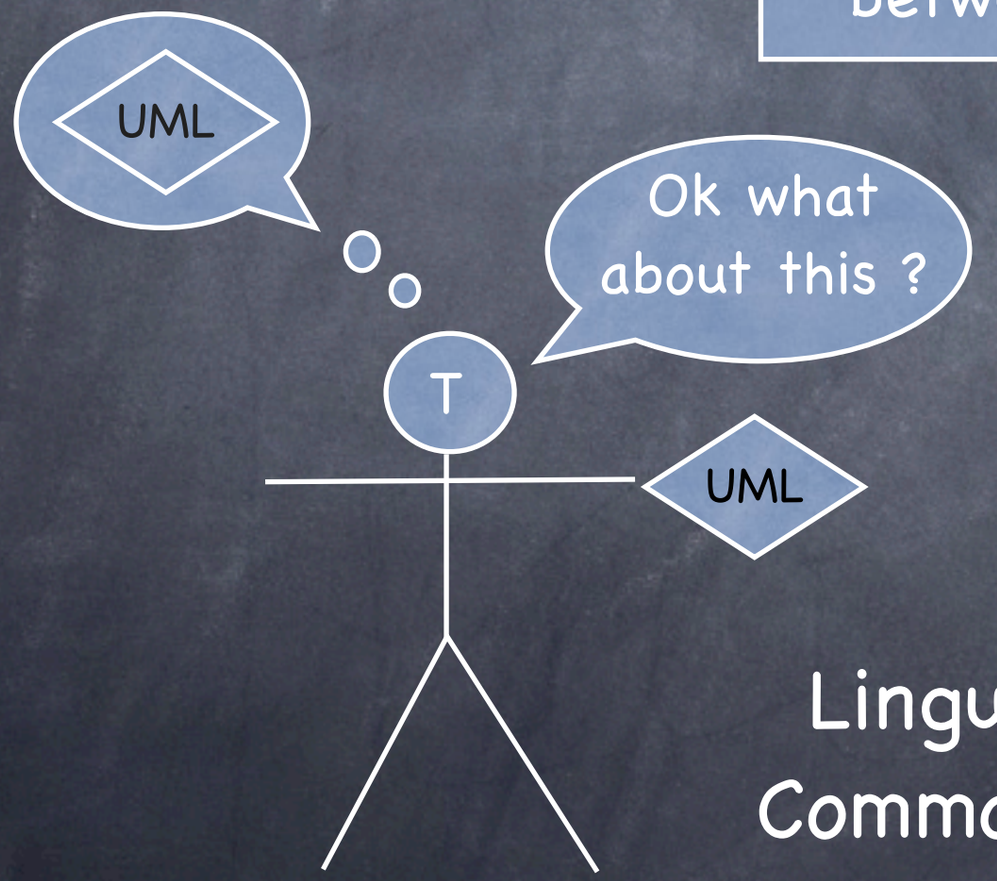
NOT good

The UML perspective - Enter UML

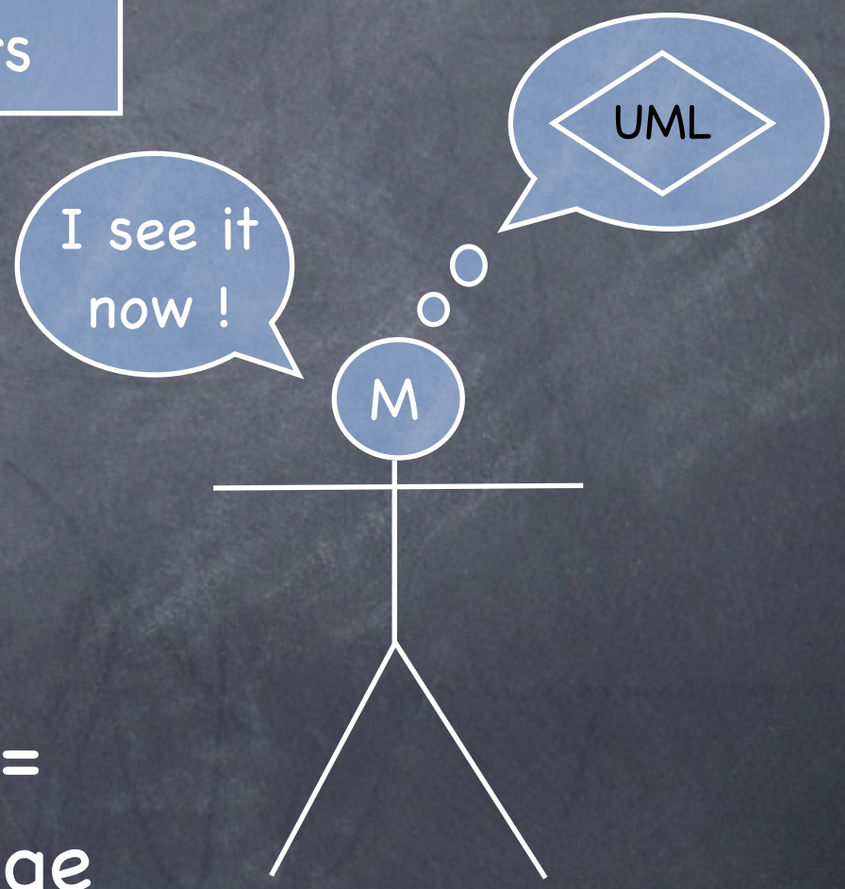
Why does Mr M get it ?

UML provides a Lingua franca that promotes communication between designers

Mr. T redesigns with UML....



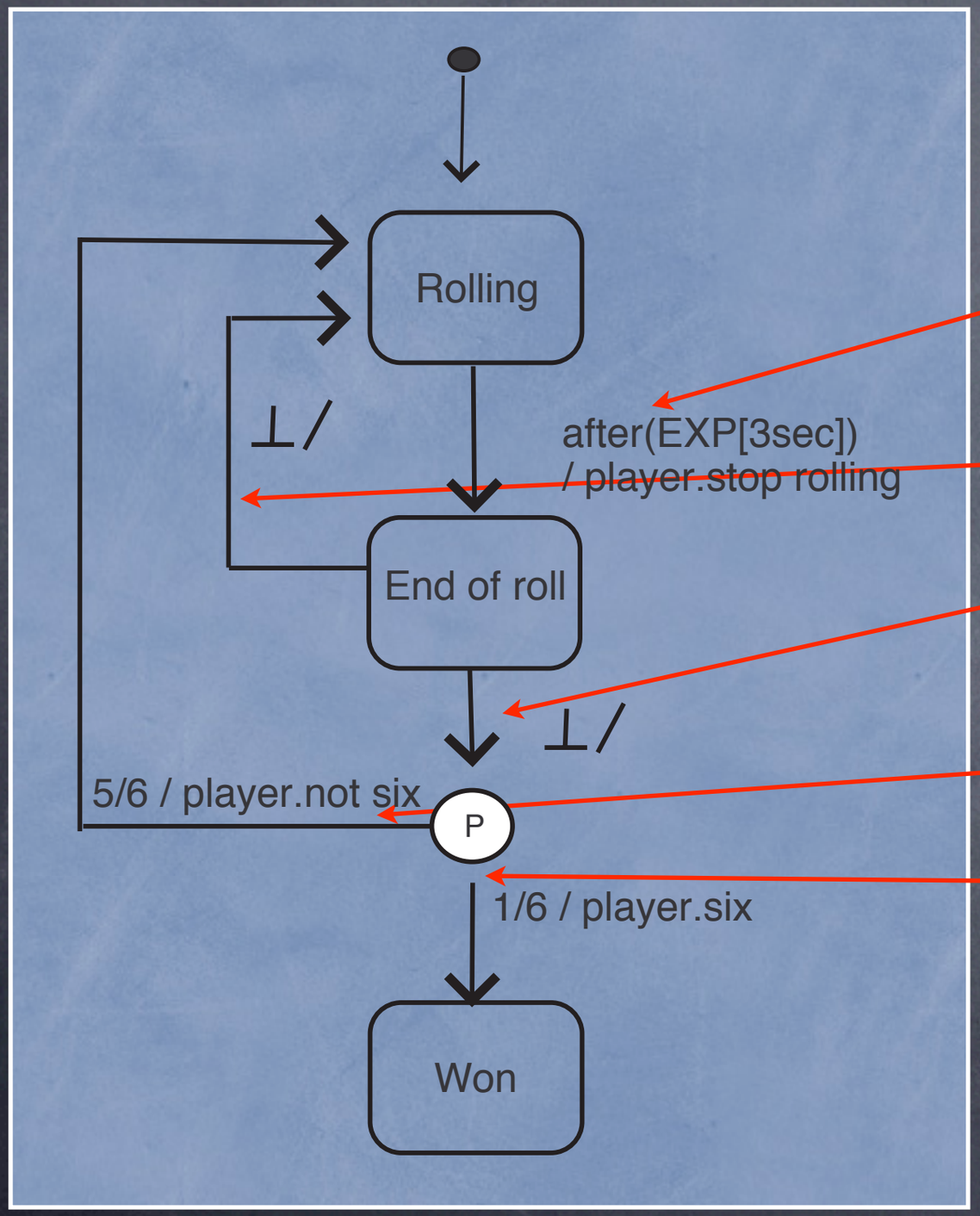
Mr. M gets it !!!



Lingua franca = Common Language

Introducing StoCharts - An Intuition

StoChart Example: 6 wins the game



(1) Dice starts rolling
(takes some time to stop: 3 sec)

(2) Nondeterministic choice
- Dice cannot be read
- Dice can be read

(3) Probabilistic choice
- {1,2,3,4,5} face-up (start rolling again)
- 6 face-up (game over)

⊥ : No need to wait to take edge

Some Distinctions - Probabilistic choice

(Example: Throwing a fair coin)

Probabilistic choice:

It is not clear which possibility out of a discrete set a system may take.

So what does "not clear" mean ?

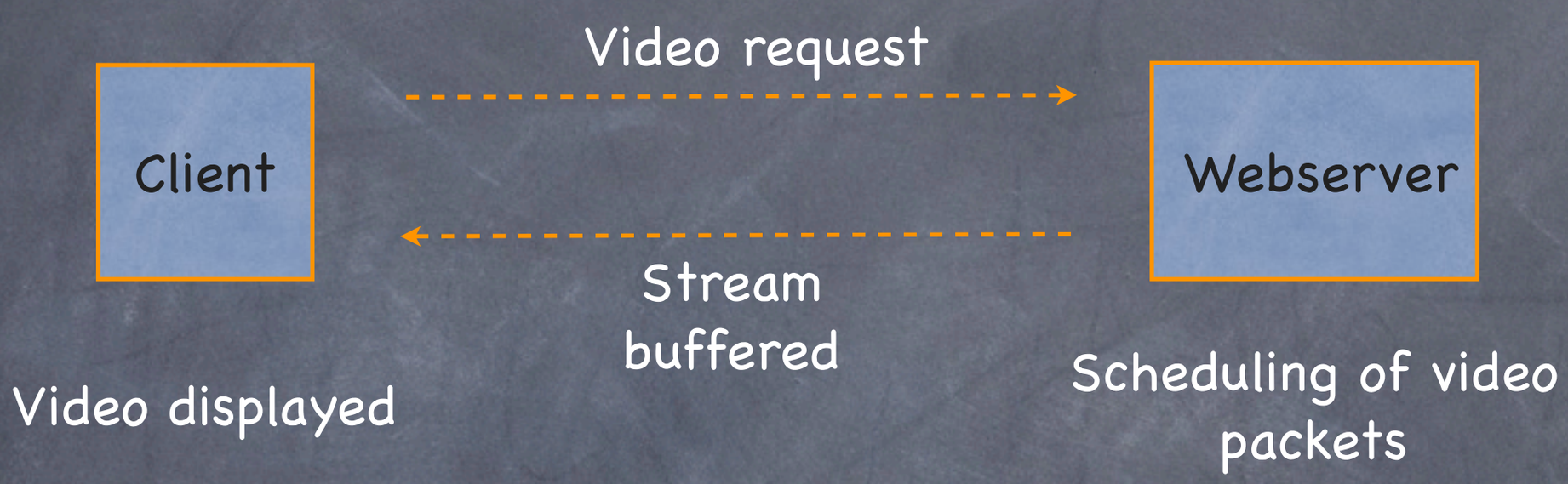
=>

- Prediction: 50% of the time heads/tails will occur.

- BUT impossible to predict the actual outcome for each toss i.e., whether it's a head or a tail.

Some Distinctions - Stochastic timing

(Example: Video Streaming)



Stochastic timing ?

=>

- Some time needed to schedule video packets (server).
- Some time for video to be displayed (client).

Some Distinctions - Probabilistic vs (non) deterministic Phenomena

Probabilistic (Examples: Weather predictions - Probability of rain, snow etc; Rolling of dice in a board game)

Deterministic (Example: Forced moves in chess)

Nondeterministic (Examples: Free moves in chess and Tic Tac Toe)

Some Distinctions - Non/Not determinism

Nondeterministic <> Not determinstic

Not deterministic Phenomena

Only infers that a phenomenon is either probabilistic or nondeterministic (and nothing more !!)

An unfortunate terminology but one we have to live with :-(

Some Distinctions - System vs Environment randomness

Environment randomness

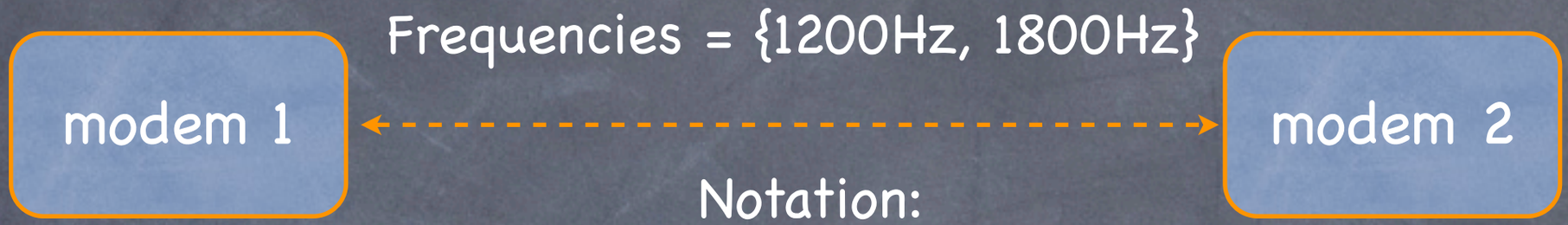
Environment randomness (Example: Reactive system - Bank ATM, Dispenser, etc)

Intuition: System exposed to external random stimuli and reacts.

System reaction is NOT random

Some Distinctions - System randomness

Requirement : Modems must transmit on different frequencies



Notation:

- Hj : "Heads" for modem j (1 <= j <=2)
[Semantics: Modem j transmits at 1200 Hz]

- Tj : "Tails" for modem j (1 <= j <=2)
[Semantics: Modem j transmits at 1800 Hz]

Randomised Algorithm (modem 1)

(1) Throw a coin.

(2) (i) Heads => Transmit at 1200 Hz.
(ii) Tails => Transmit at 1800 Hz.

(3) No response received (modem 2 using similar frequency) after some time => goto 1
else end. (modems using different frequencies)

Overview

- An introduction to Statecharts
- StoCharts
  - Syntax
  - Semantics
- Case Study: A Beverages dispenser
- Conclusion

UML Extensions - P-Statecharts & StoCharts

We can extend UML Statecharts: 2 fold Extension

Extension 1: Probabilistic choice
(Example: Heads/Tails choice on a coin toss)

Extension 2: Stochastic timing
(Example: Timing in a Drink dispenser)

Statecharts + Extension 1
= Probabilistic Statecharts (aka P-Statecharts)

Statecharts + Extension 1 + Extension 2
= Stochastic Statecharts (aka StoCharts)

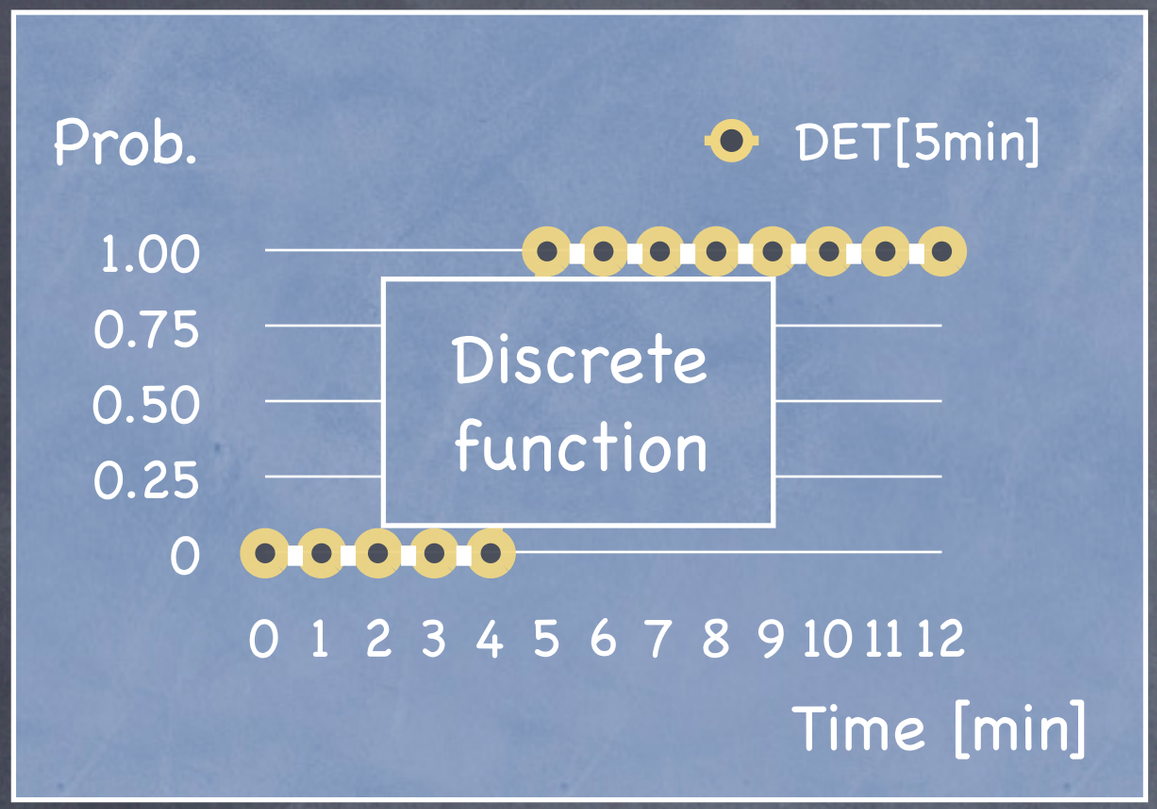
UML Extensions - Random probability distributions

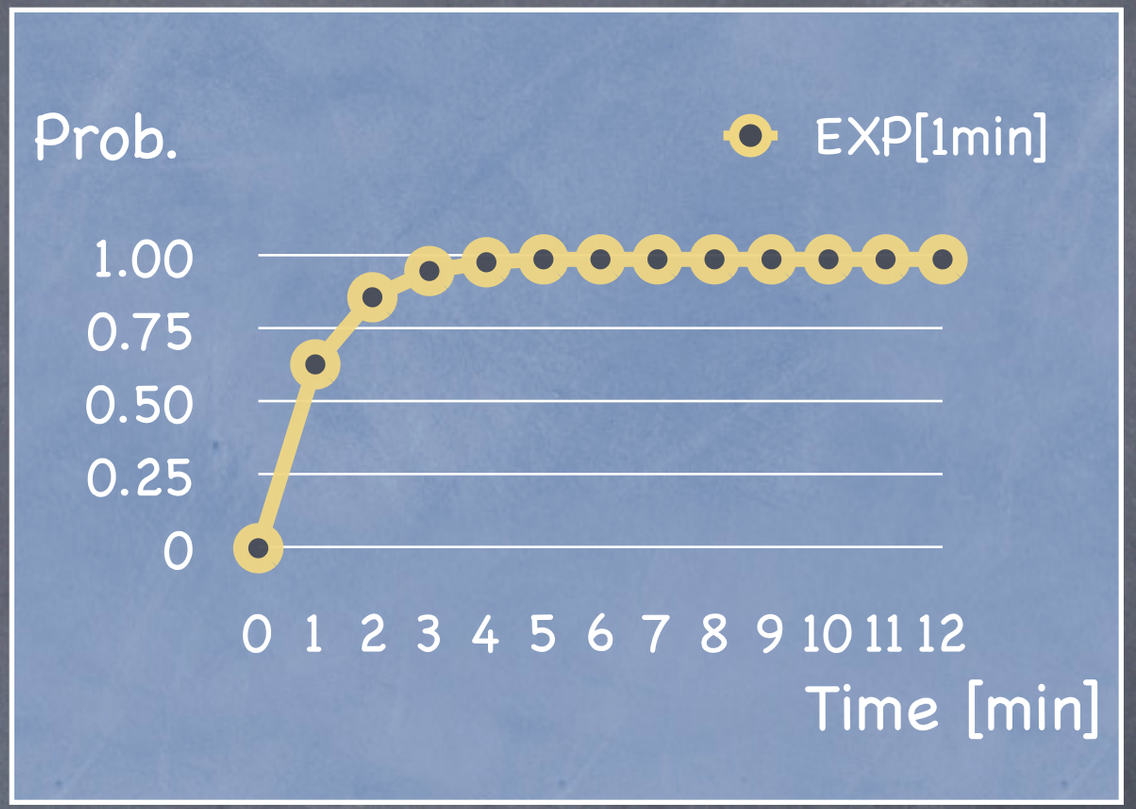
DET [x min]
A deterministic distribution with a duration of x minutes

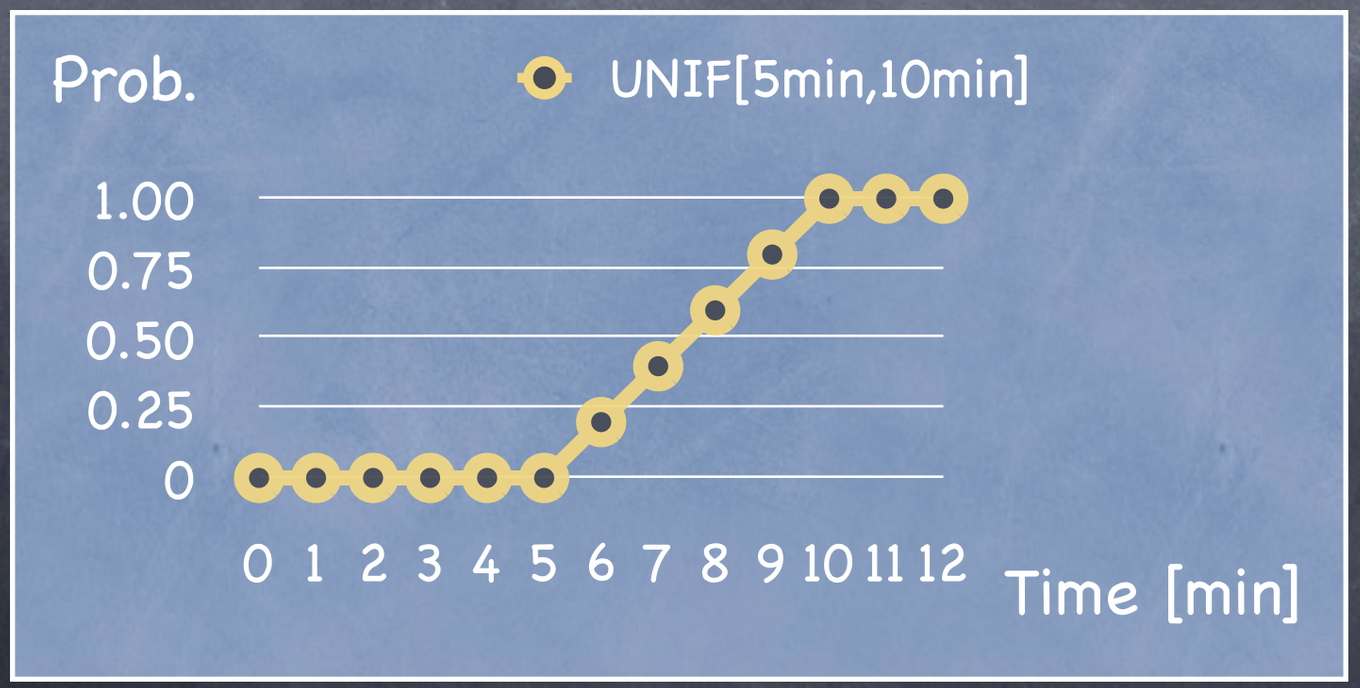
EXP [x min]
A negative exponential distribution with a mean of x minutes

UNIF [x min, y min]
A uniform distribution in the interval from x to y minutes

UML Extensions - Random probability





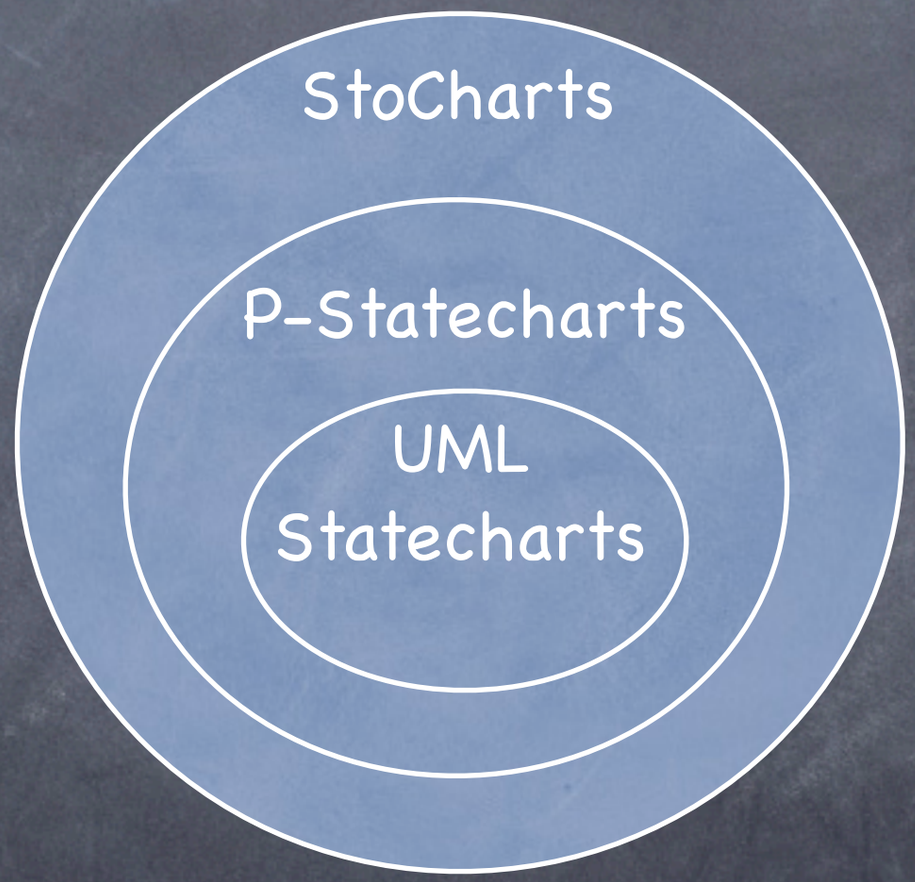


UML Extensions - Statechart relations

In a nutshell ...

UML Statecharts are trivial P-Statecharts

UML Statecharts are trivial StoCharts



UML Statecharts + Probabilistic choice => P-Statecharts

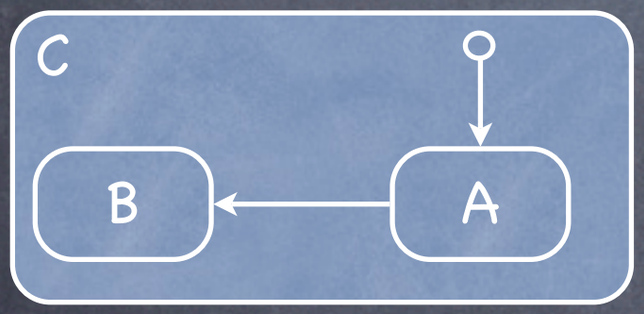
P-Statecharts + Stochastic timing => StoCharts

Overview

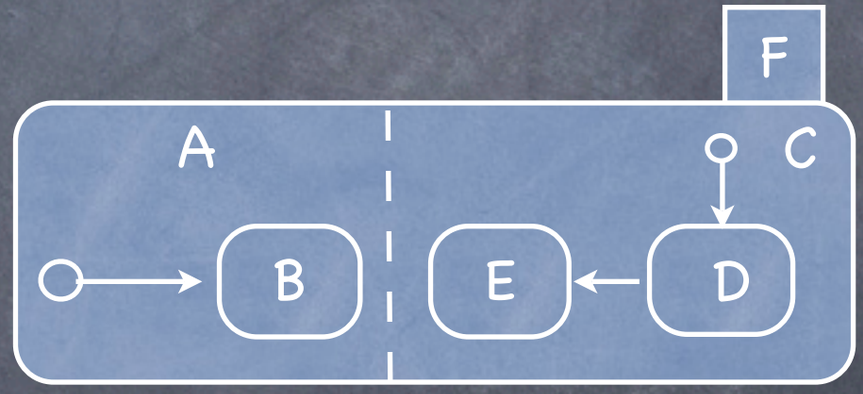
- An introduction to Statecharts
- StoCharts
  - Syntax
    - Semantics
- Case Study: A Beverages dispenser
- Conclusion

A syntactical perspective - StoChart drawing

Nodes



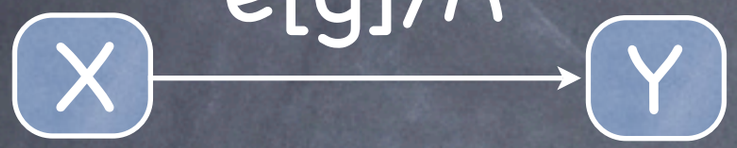
Parent node C encloses its children A, B
(Default node: A)



F (AND) node partitioned by (OR) children A and C
(Default nodes: B,D)

A syntactical perspective - StoChart drawing

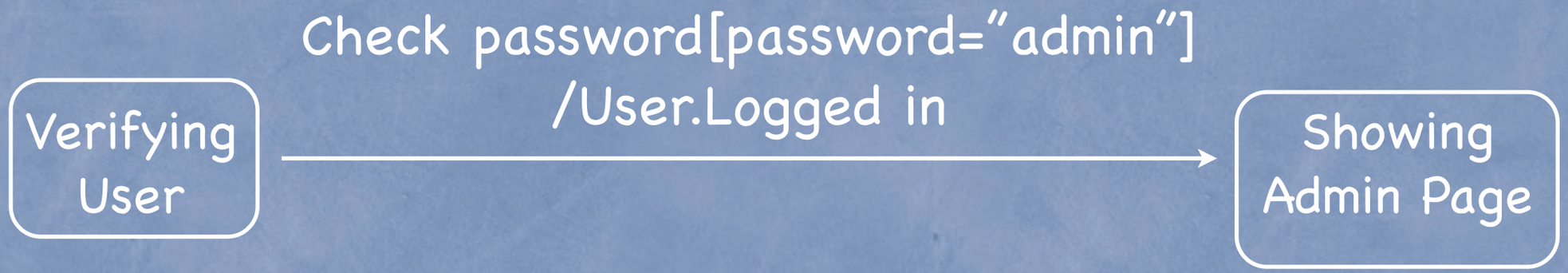
Trivial P-edges



X : Set of source nodes
Y : set of target nodes
e : event
g : guard
A : action set

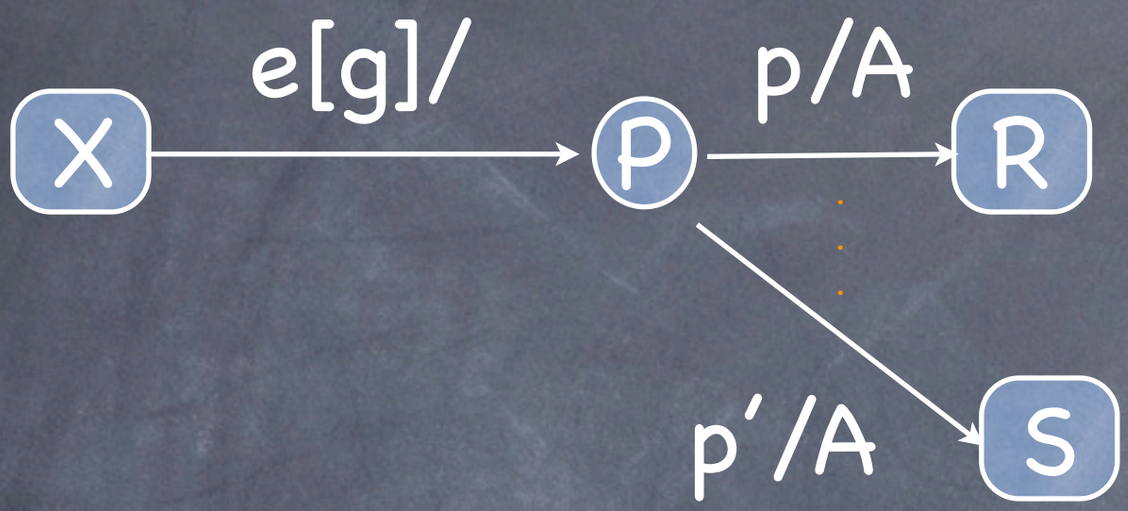
Here: Actions "A" have the form (send) j.e (send event e to the external component identified by j)

Example (Router Login)



A syntactical perspective - StoChart drawing

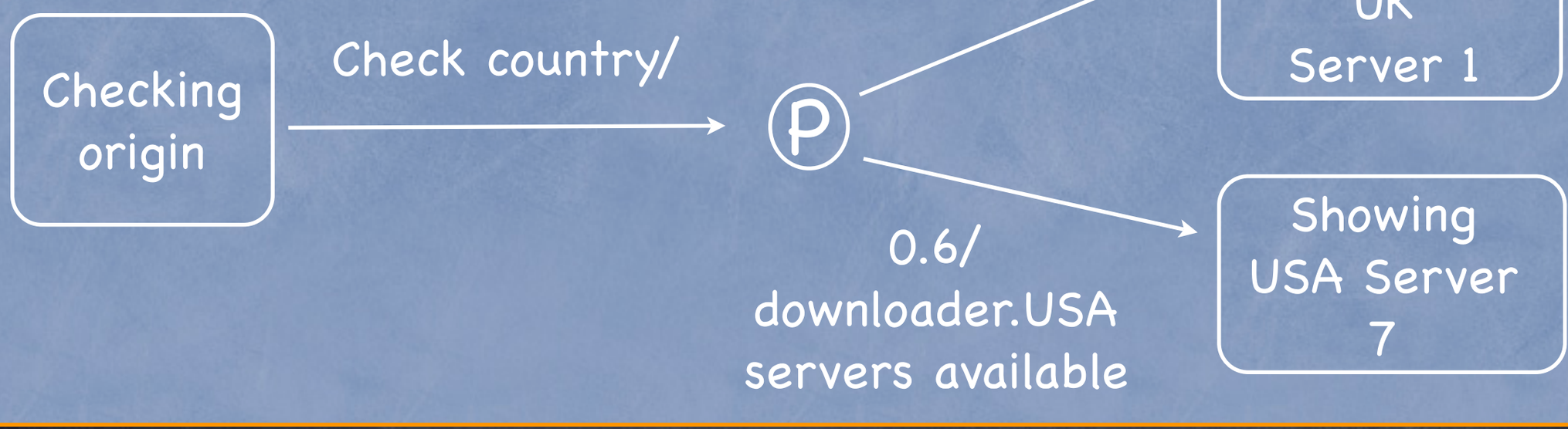
Non trivial P-edges



X : Set of source nodes
Y : set of target nodes
e : event
g : guard
A : action set
p,p': Probabilities

p=p' is also possible

Example (File downloading)



A syntactical perspective - System Perspective

- Intuition: A system is a finite collection of intercommunicating StoCharts

- Formally: Denotable by the set seen below.

$$\{SC_i \mid 1 \leq i \leq n, n \in N\}$$

A syntactical perspective - StoChart

StoChart 4-Tuple

$$SC_i = (Nodes_i, Events_i, Vars_i, PEdges_i)$$

$Nodes_i$ A finite set of nodes structured in a tree

$Events_i$ A finite set of events

$Vars_i$ A finite set of local variables

$PEdges_i$ A finite set of P-edges

A syntactical perspective - StoChart Events

$$SC_i = (Nodes_i, Events_i, Vars_i, PEdges_i)$$

- A finite set of events (Signals that may trigger a state transition)
  - pseudo event after(F) - stochastic random delays
  - $\perp \notin Events_i$ No event is required to trigger a P-edge
  - $PSEvents_i$ Set of Pseudo events

A syntactical perspective - StoChart Variables

$$SC_i = (Nodes_i, Events_i, Vars_i, PEdges_i)$$

- Intuition: A finite set of local variables with an initial value that assigns initial values to variables

- Formally: Denotable by the function below

$$V_0 : Vars_i \rightarrow \mathbb{Z}$$

A syntactical perspective - StoChart P-edges

P-edge 4-Tuple (X, e, g, P)

- A finite set of P-edges
  - $X \subseteq Nodes_i$ A non-empty set of source state nodes
  - $e \in Events_i \cup PSEvents_i \cup \{\perp\}$ Triggering event that occurs
  - $g \in Guards_i$ A guard existing on the edge
  - P denotes the possible actions and target state nodes (formal description needed)

A syntactical perspective - StoChart P-edges

A formal description of P

$$(X, e, g, P)$$

P can be formally described by a probability measure in a discrete probability space

$$(Pow(Actions_i) \times (Pow(Nodes_i) - \{\emptyset\}), P)$$

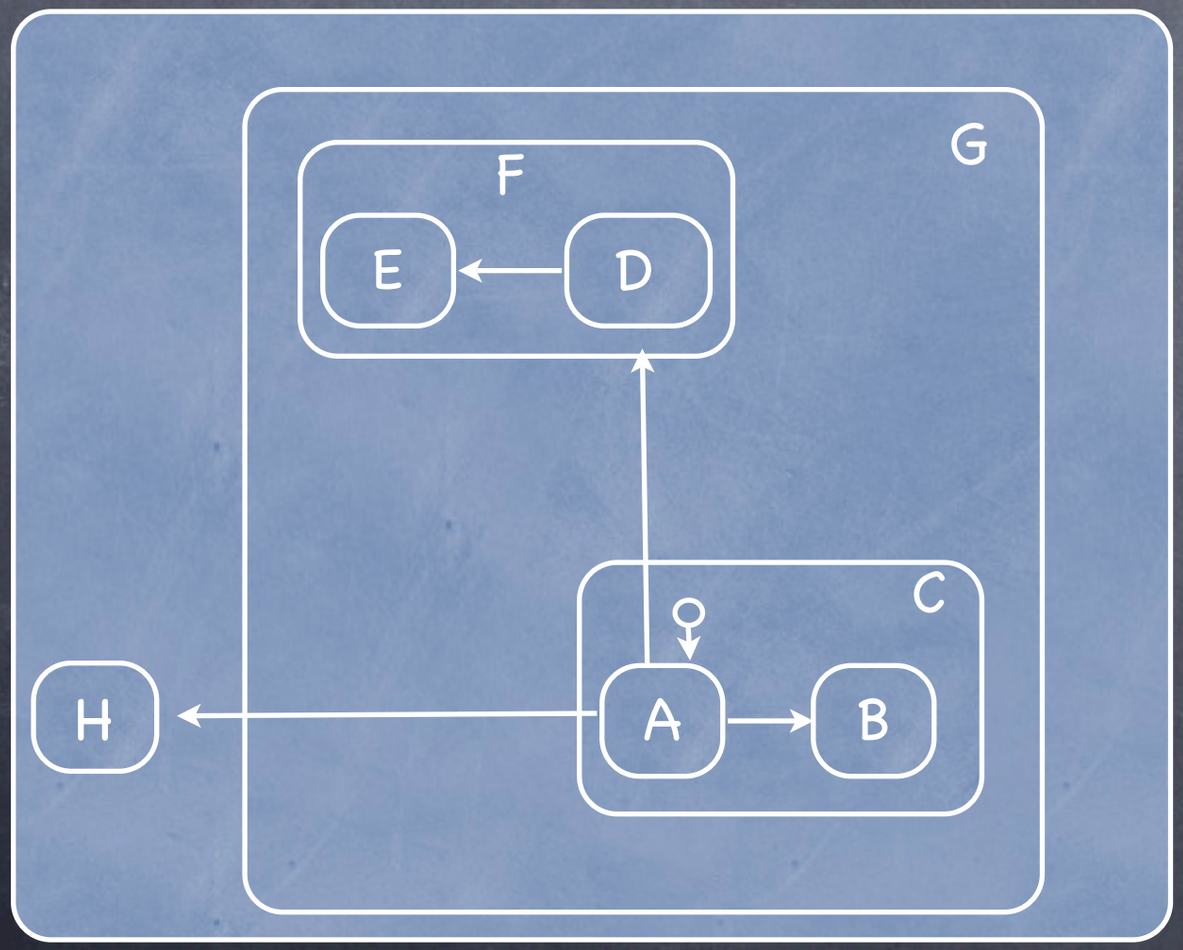
Ω

Denoting the set of possible outcomes

Pow: Denotes a Power set

A syntactical perspective - StoChart scopes

Intuition: The scope of a P-edge is the smallest (in the parent-child hierarchy) OR-node containing **both** the source nodes and the target nodes



Scope(A->H) = root
Scope(A->D) = G
Scope(A->B) = C

Overview

- An introduction to Statecharts
- StoCharts
  - Syntax
  - Semantics
- Case Study: A Beverages dispenser
- Conclusion

A semantical perspective - Automaton approach

IOSA 7-Tuple $(L, l_0, T, A, I, O, \Delta)$

IOSA = Stochastic I/O-Automaton

L — A set L of locations with an initial location $l_0 \in L$

T — A set of timers (each possessing a cdf)

A — A set of actions partitioned into input-actions (from external environment) and output-actions (system intrinsic)

I — An input transition relation

O — A probabilistic output transition relation

Δ — A delay transition relation

A semantical perspective - Automaton approach

IOSA 7-Tuple $(L, l_0, T, A, I, O, \Delta)$

I

An input transition relation

$$I : L \times A^{in} \rightarrow L$$

A semantical perspective - Automaton approach

IOSA 7-Tuple $(L, l_0, T, A, I, O, \Delta)$

O

A probabilistic output transition relation

$$O \subseteq L \times P(A^{out} \times Pow(T) \times L)$$

A semantical perspective - Automaton approach

IOSA 7-Tuple $(L, l_0, T, A, I, O, \Delta)$

Δ

A delay transition relation

$$\Delta \subseteq L \times T \times L$$

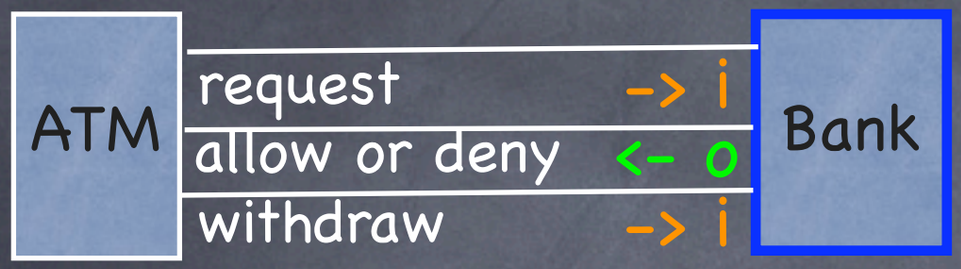
A semantical perspective - Enabled Transitions

IOSA 7-Tuple $(L, l_0, T, A, I, O, \Delta)$

o: output transition
i: input transition
d: delay transition

Transition	Enabled
o	Always
i	Only if Input action present (Input enabledness)
d	Only if timer expires (reaches zero)

A semantical perspective - IOSA example



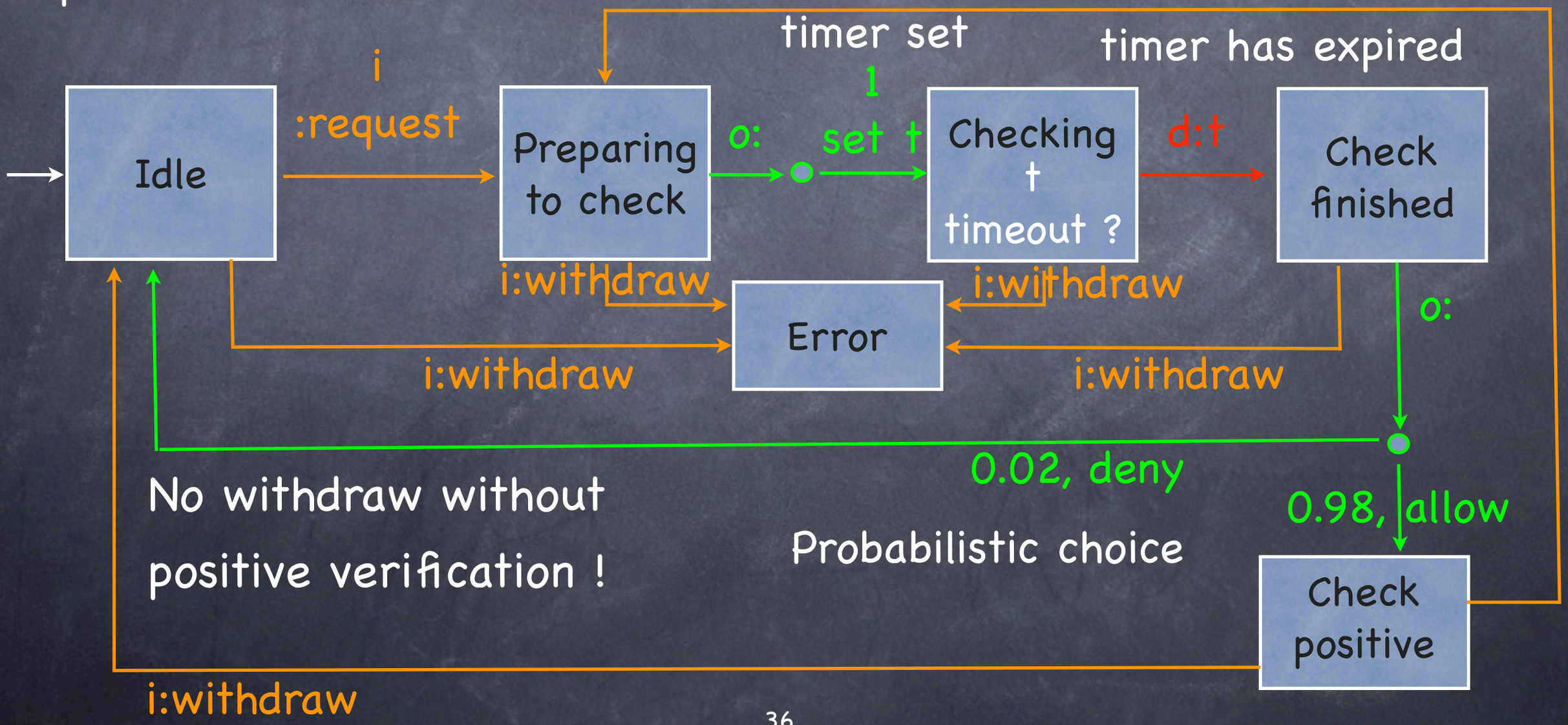
o: output transition, i: input transition
d: delay transition

Bank behavior being modeled !

input transition enabled

output transitions always enabled !

i:request



No withdraw without positive verification !

Probabilistic choice

Overview

- An introduction to Statecharts
- StoCharts
  - QoS Extensions
    - Syntax
    - Semantics
  - Case Study: A Beverages dispenser
- Conclusion

A beverage dispenser - Intuition

How does it work ?

- Drinks offered ?
  - "Espresso" coffee and "oolong" tea.
- Payment ?
  - Money chips - 1 Chip per drink.

A beverage dispenser - Modeling perspective

What are we modeling ?

The behavior of the system (system randomness), as seen by the customer.

What we are not modeling

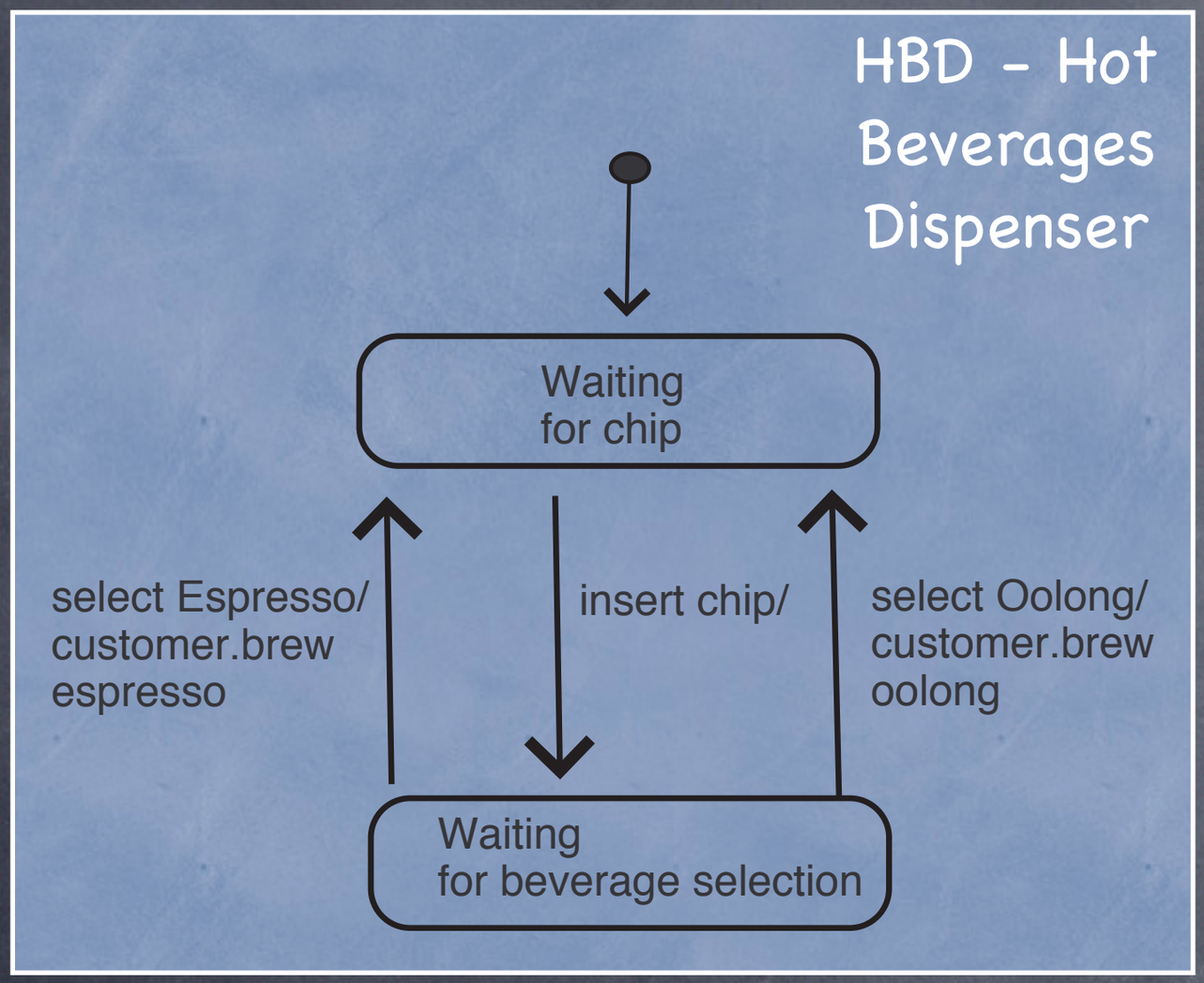
The expected/desired behavior of the customer (environmental randomness) is **NOT** being considered here.

A naive UML Statechart - Events

- Events are sent from the customer to the HBD.
  - insert chip.
  - select espresso.
  - select oolong.

HBD = Hot Beverages Dispenser

A naive UML Statechart - Closer look

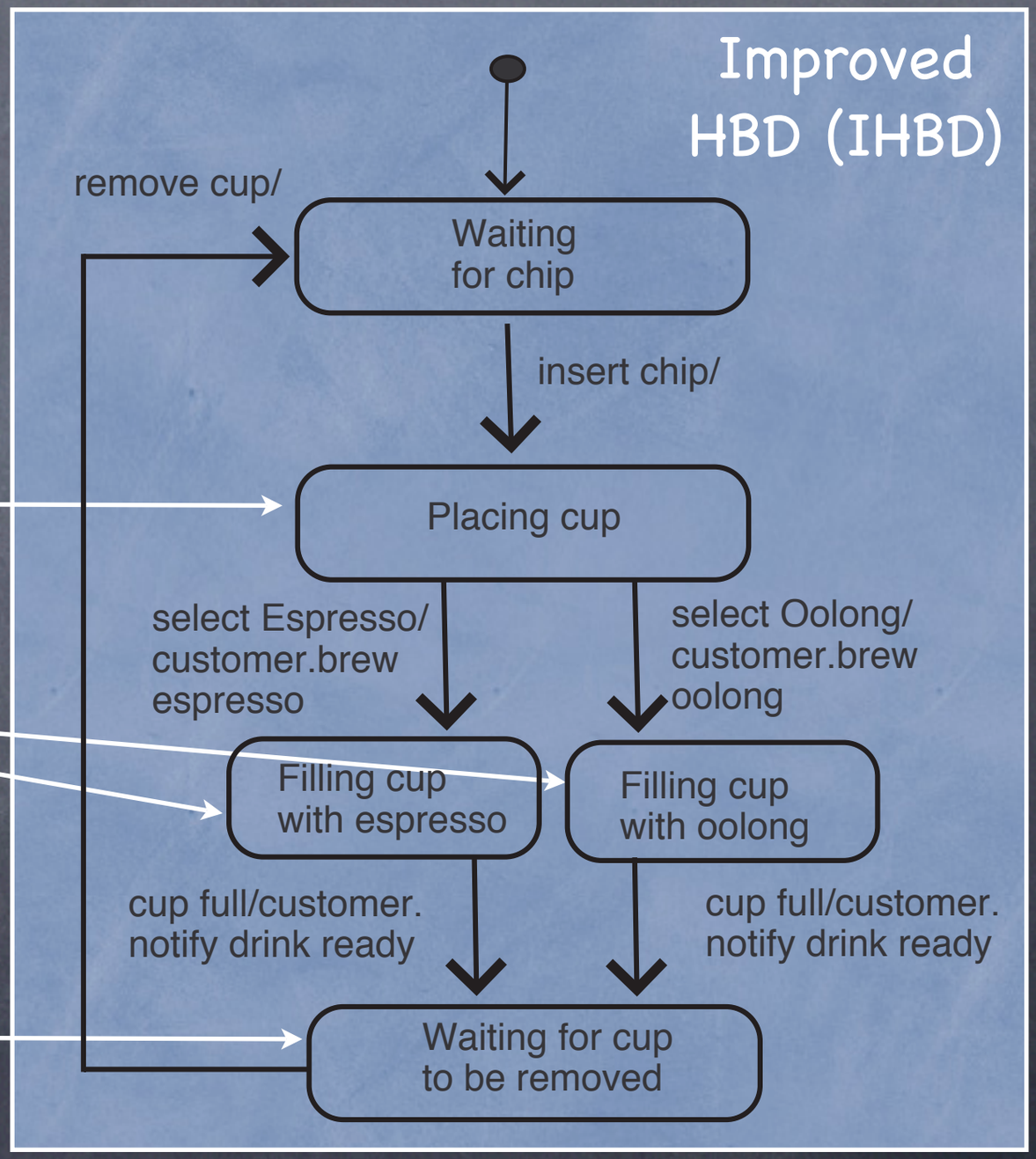


- insert chip/:
  Intuition: In the event of a chip being inserted, the system evolves to its next state
- select oolong/customer.brew oolong:
  Intuition: In the event that oolong is selected, send the event "brew oolong" to the customer
- Analog for espresso

An improved UML Statechart - Closer look

What's wrong with our naive design ?

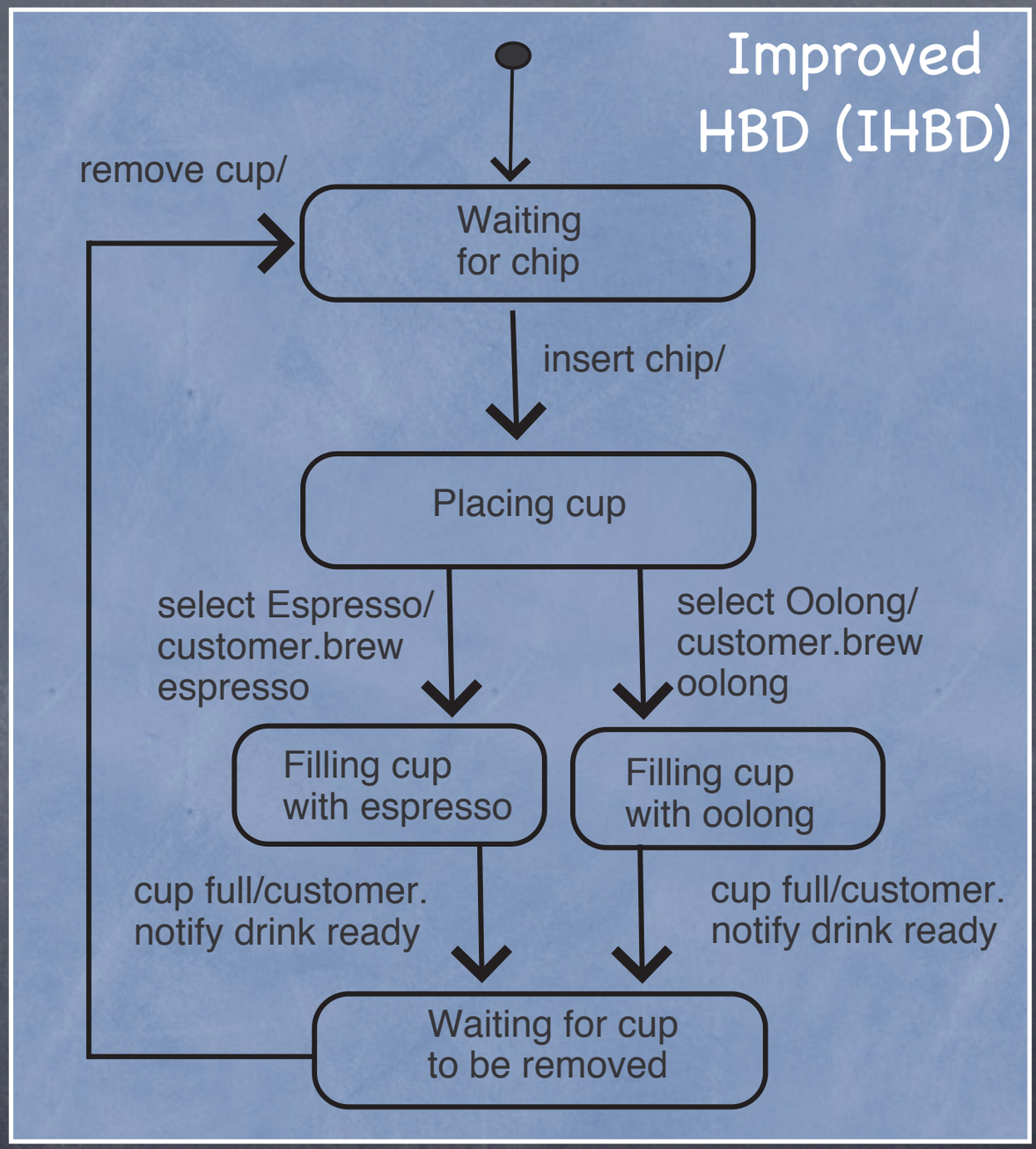
- Incomplete - Drinking what we paid for ?
- Completing the model
  - "Placing" of cups in cup holder
  - "Filling" of the cups
  - "waiting" for someone (not a system) to remove the cup



Extending UML Statecharts - UML Limitations

Limitations of the IHBD

- Modeling system requirements ?
  - (1) No cups => coin returned to customer
  - (2) Return a chip exactly after 10 seconds ("timeout")
  - (3) Brewing takes about 5 seconds ("timeout")



Realisation: Impossible to model our requirements without extensions !!!

Extending UML Statecharts - Modeling requirements

Requirements to be modeled

- (1) No cups => coin returned to customer
- (2) Return a chip exactly after 10 seconds ("timeout")
- (3) Brewing takes about 5 seconds ("timeout")

IHBD extended (probabilistic choice & stochastic timing)

1 → "Probabilistic choice" Enough cups vs not enough cups

2 → "Stochastic timing" after(DET [10s]) / customer.return chip

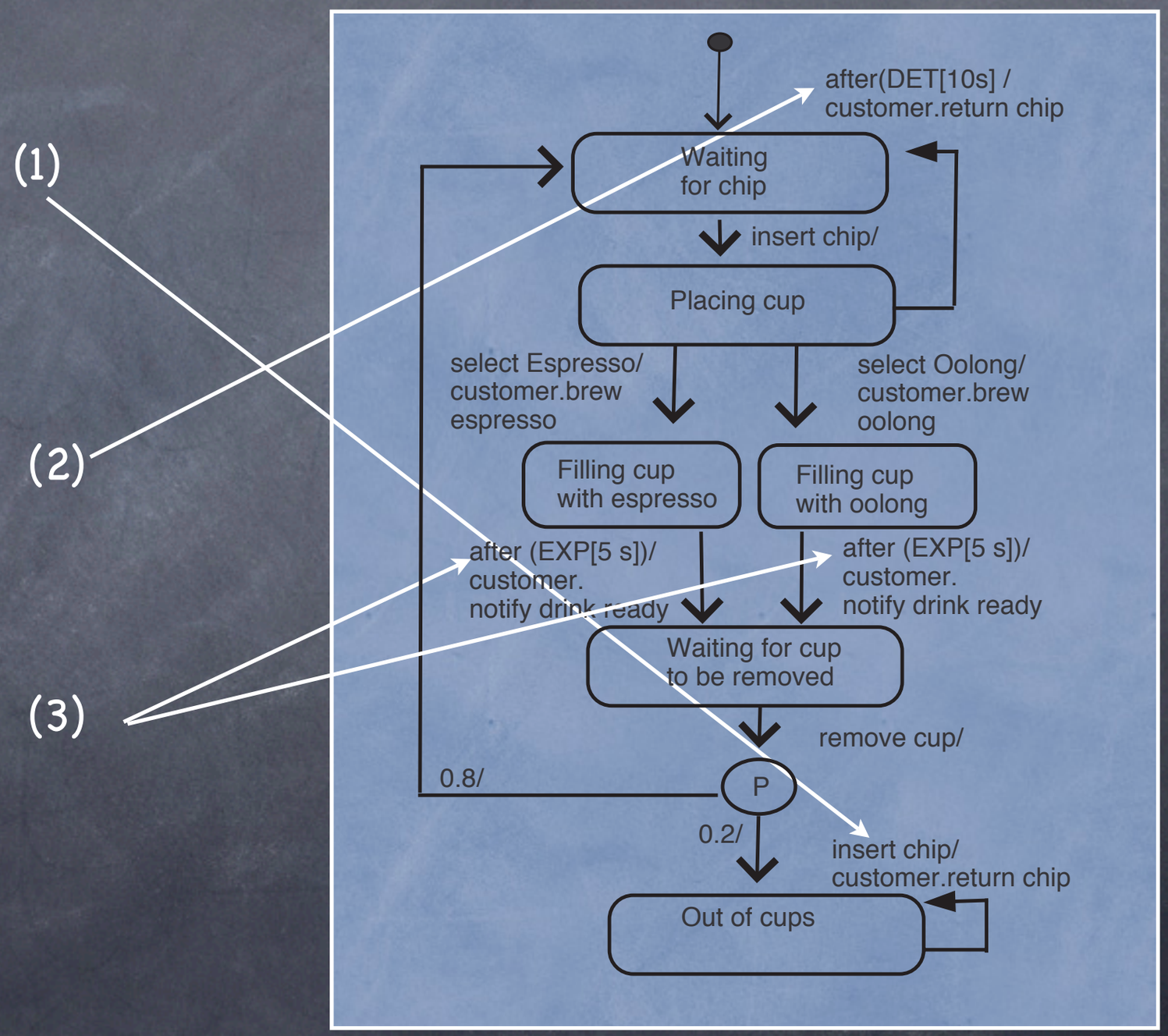
3 → "Stochastic timing" after(EXP [5s]) / customer.notify drink ready

Extending UML Statecharts - Birth of a StoChart

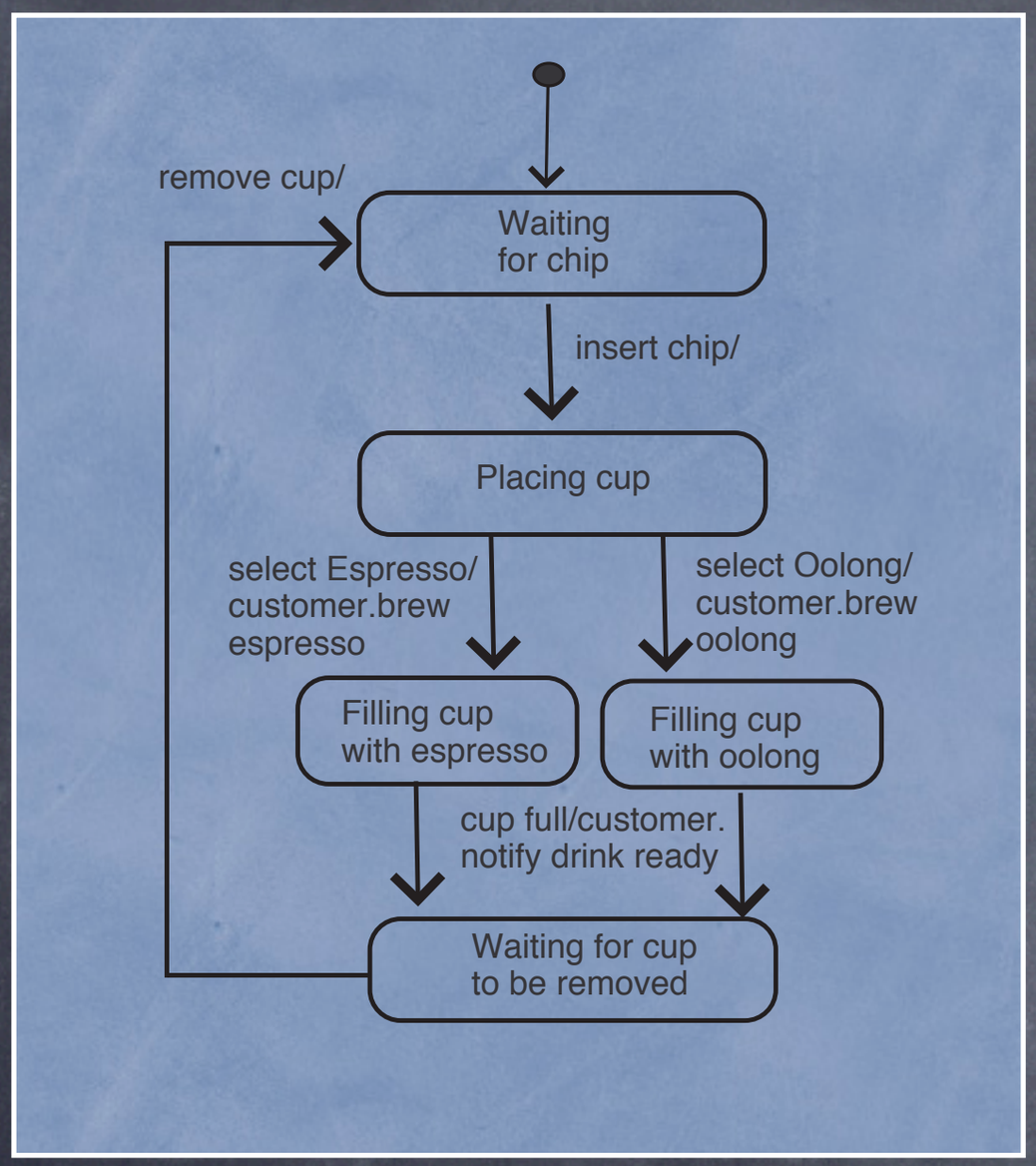
Requirements to be modeled

- (1) No cups => coin returned to customer
- (2) Return a chip exactly after 10 seconds ("timeout")
- (3) Brewing takes about 5 seconds ("timeout")

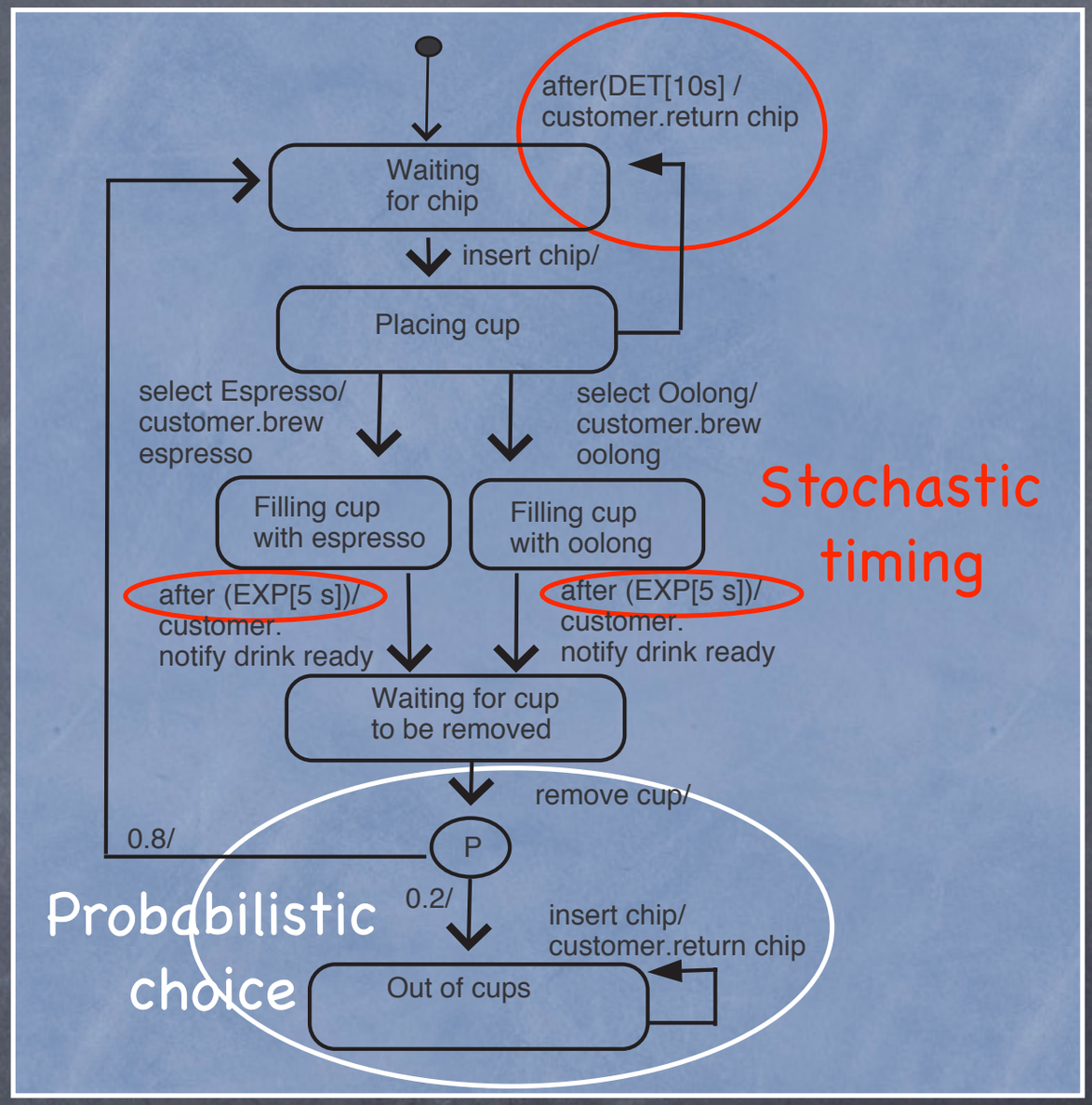
Stochastic IHBD (S-IHBD)



Extending UML Statecharts - Statecharts compared



IHBD - without extensions.
Modeling requirements involving Stochastic behavior not possible !



S-IHBD (IHBD with extensions).
Modeling requirements involving Stochastic behavior now becomes possible !!

Overview

- An introduction to Statecharts
- StoCharts
  - QoS Extensions
  - Syntax
  - Semantics
- Case Study: A Beverages dispenser
- Conclusion

What have we learnt ?

- Awareness of real UML limitations and the need to extend them.

- Extending UML Statecharts with :

    - Probabilistic choice (P-Statecharts).

    - Stochastic timing (StoCharts).

- Informal and formal syntax of StoCharts (with examples).

- Informal semantics : Stochastic Input/Output Automata (IOSA).

- Application of UML extensions: Case study of beverage dispenser.

Some useful reading

- Martin Fowler. UML Distilled: A Brief Guide to the standard Object Modeling Language, Third edition. Addison-Wesley Professional, 3rd edition, 2003.
- David N. Jansen. Extensions of Statecharts with Probability, Time, and Stochastic Timing. PhD thesis, Universiteit Twente, Bern, 2003.
- Scott Kim. Interdisciplinary cooperation. In Brenda Laurel, editor, the Art of Human Computer Interface Design, pages 31-44, Addison Wesley, 1990.

Thank you
for your attention !