

# Timing Diagram

Lecture # 35,36  
30 Nov, 2 Dec

Rubab Jaffar  
[rubab.jaffar@nu.edu.pk](mailto:rubab.jaffar@nu.edu.pk)

## Software Design and Analysis CS-324



# Today's Outline

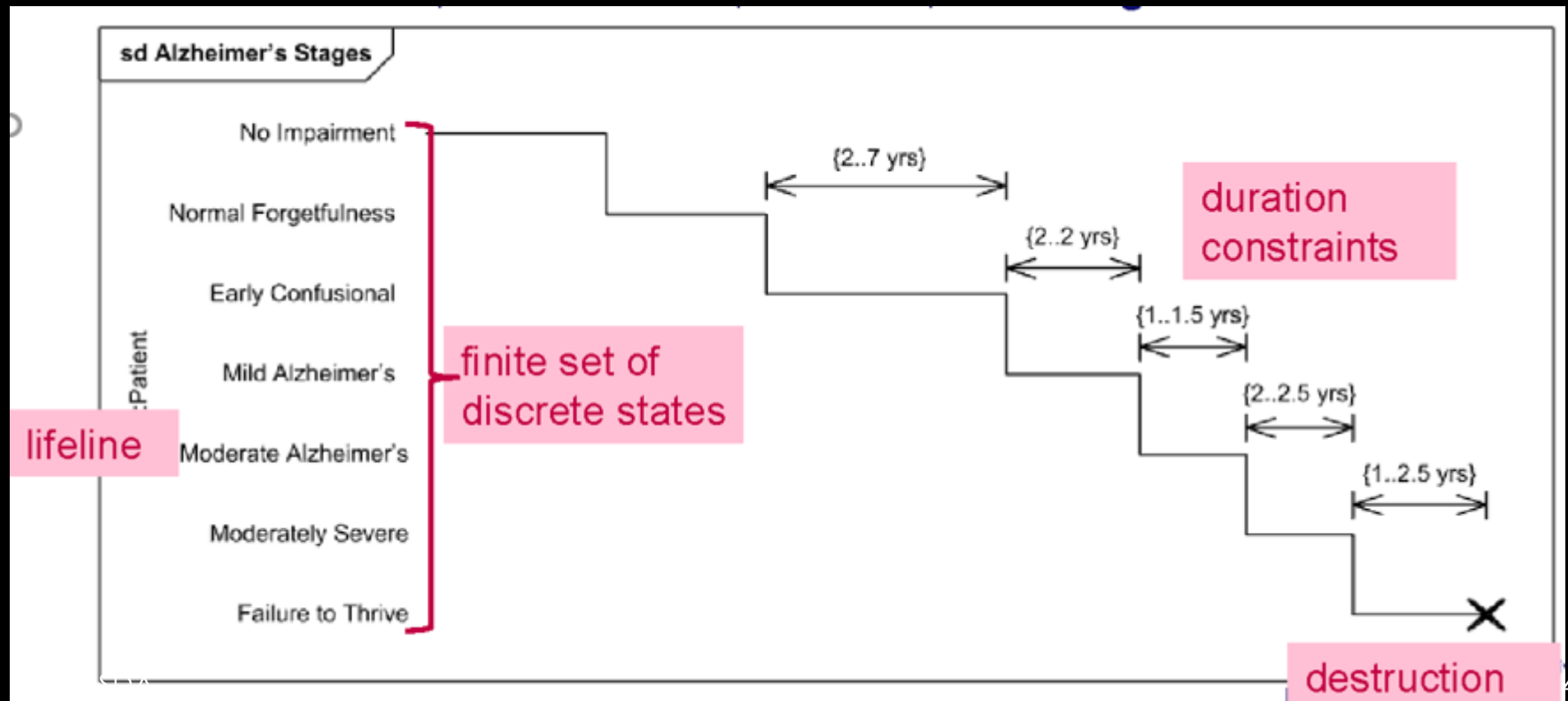
- Interaction Diagram
  - Timing Diagram
- Elements of Timing Diagram
- Timing Diagram Notations
- Examples

# Timing Diagrams

- A timing diagram allows you to show the interaction of objects and changes in state for those objects along a time axis.
- A timing diagram provides a convenient way to show active objects and their state changes during their interactions with other active objects and system resources.
- The X-axis of the timing diagram has the time units, while the Y-axis shows the objects and their states.
- Timing diagrams describe behavior of both individual **classifiers** & **interactions of classifiers**, focusing attention on time of events causing changes in the modeled conditions of the lifelines.

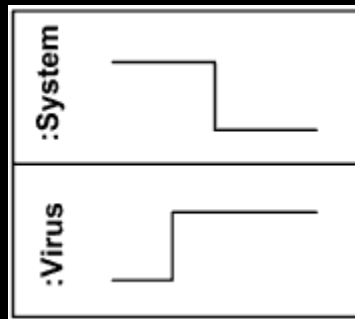
# Timing Diagram Elements

- Timing Diagrams are Interaction diagram for reasoning about time.
- **Basic elements:** lifelines, states, duration/time constraints, destruction, events, messages



# Lifeline

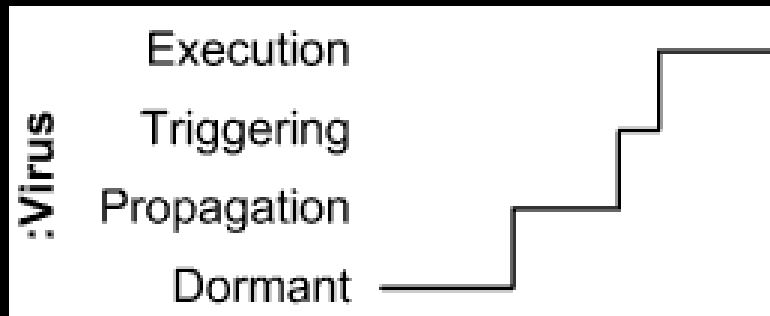
- **Lifeline** is a **named element** which represents an **individual participant** in the interaction. Lifelines represent **only one** interacting entity.
- Lifeline on the timing diagrams is represented by the **name** of classifier or the instance it represents. It could be placed inside diagram frame or a "swimlane".



*representing instances of  
System and Virus*

# State or Condition Timeline

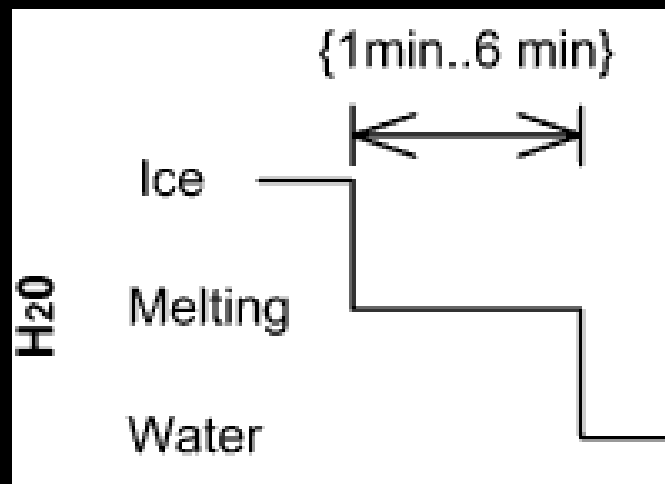
- Timing diagram could show **states** of the participating **classifier** or attribute, or some testable **conditions**, such as a discrete or enumerable value of an attribute.



- Timeline shows Virus changing its state between Dormant, Propagation, Triggering and Execution state*

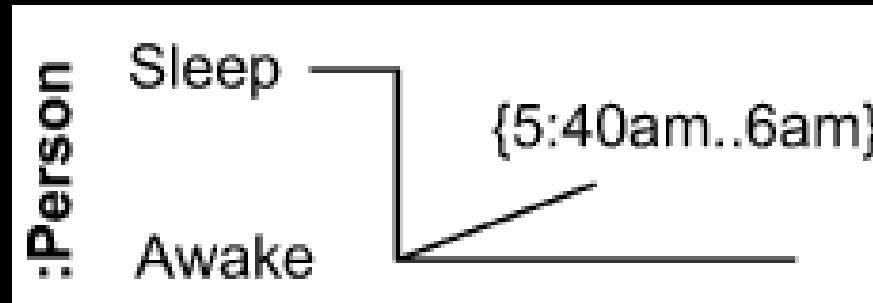
# Duration Constraint

- **Duration constraint** is an **interval constraint** that refers to a **duration interval**. The duration interval is duration used to determine whether the constraint is satisfied.
- E.g., *Ice should melt into water in 1 to 6 minutes*



# Time Constraint

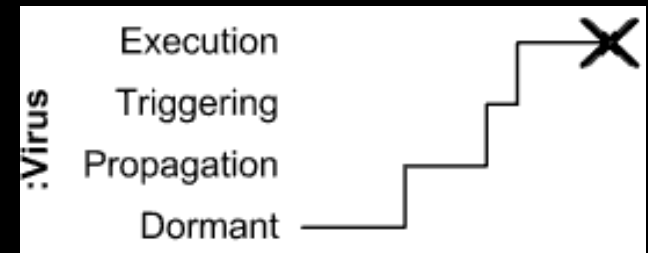
- **Time constraint** is an **interval constraint** that refers to a **time interval**. The time interval is time expression used to determine whether the constraint is satisfied.
- Typically this graphical association is a small line, e.g., between an occurrence specification and a time interval.
- E.g., *Person should wake up between 5:40 am and 6 am*



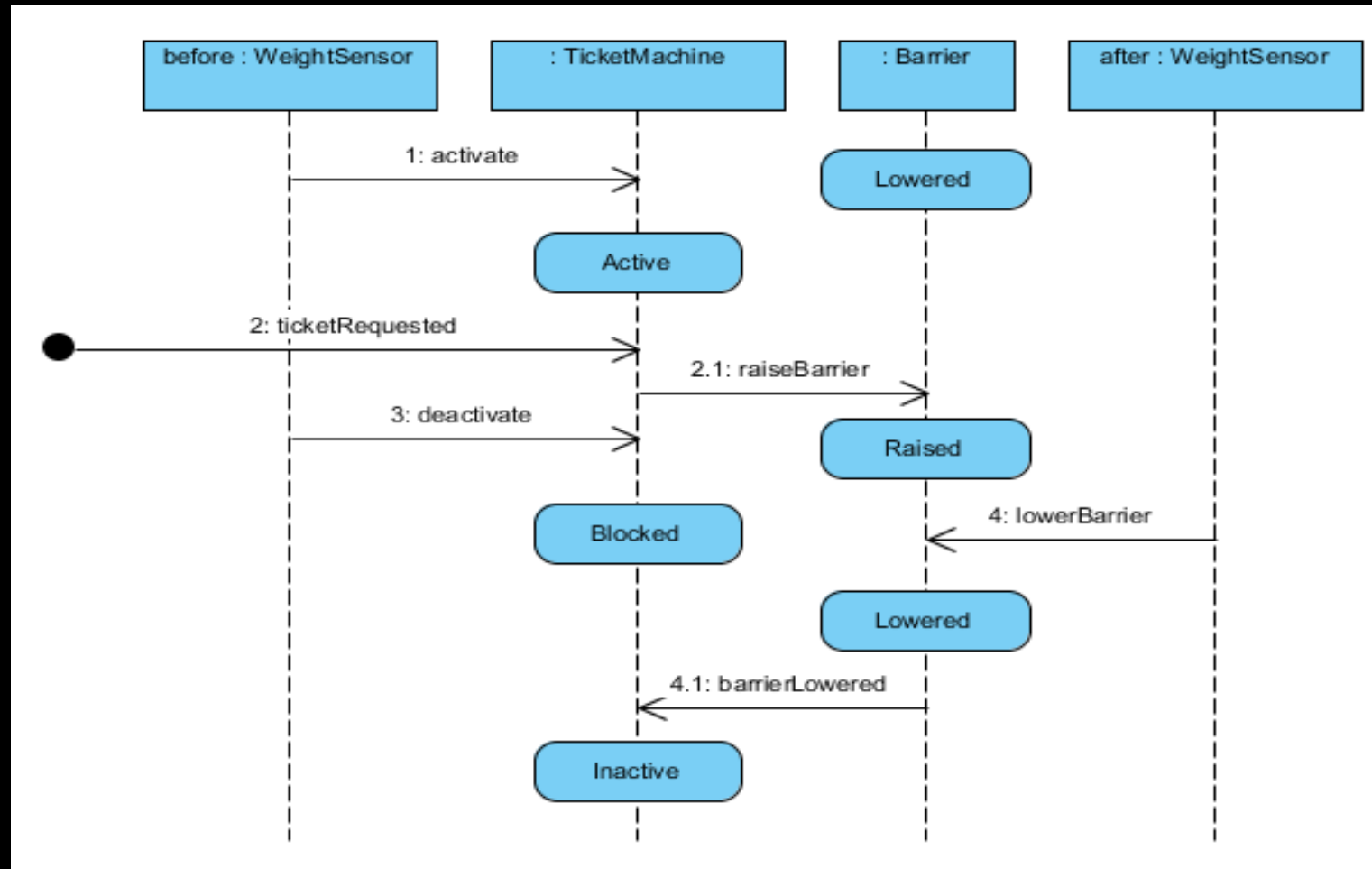


# Destruction Occurrence

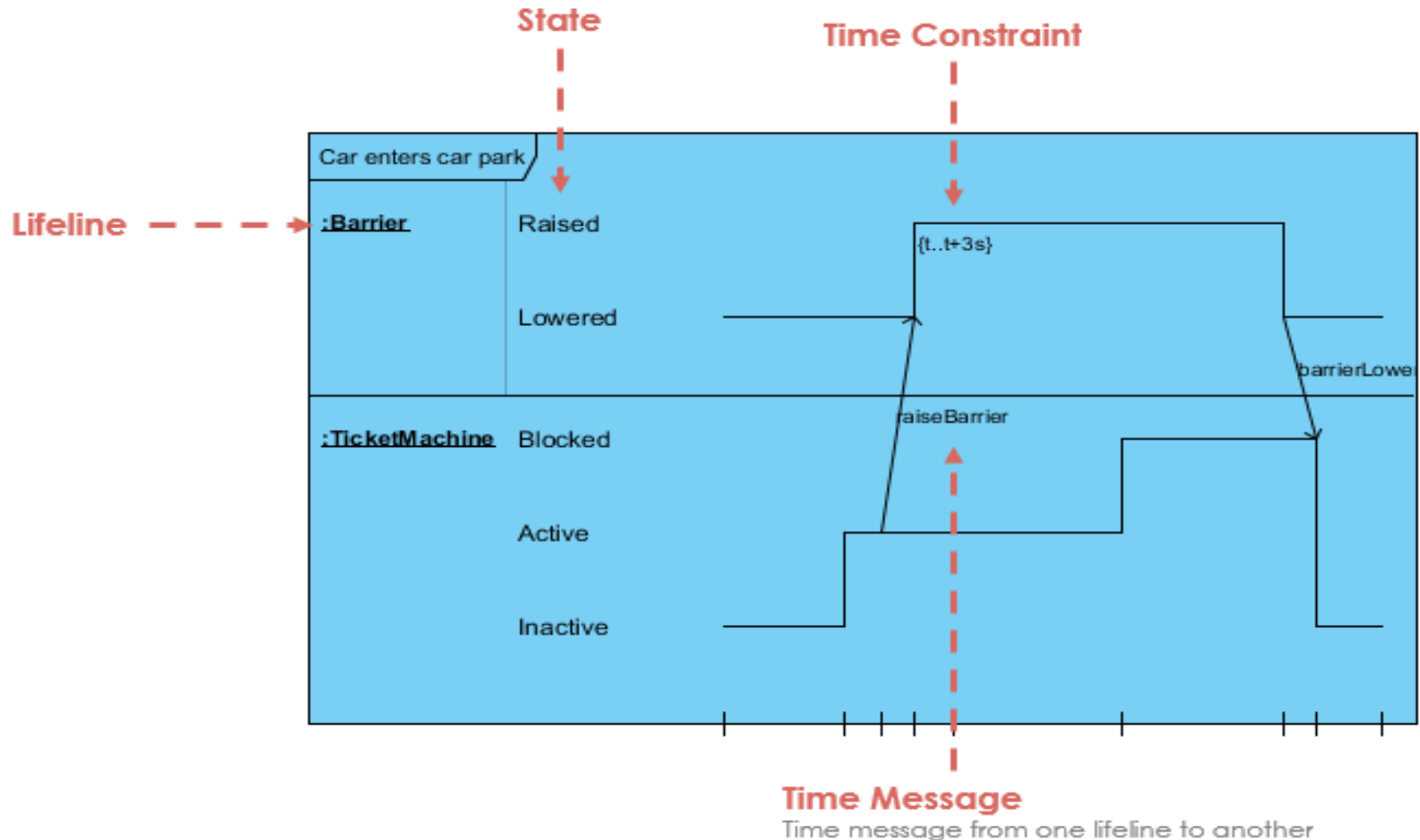
- **Destruction occurrence** is a **message occurrence** which represents the destruction of the instance described by the **lifeline**.
- It may result in the subsequent destruction of other objects that this object owns by **composition**.
- No other occurrence may appear after the destruction event on a given lifeline.
- **Notation**
- The destruction event is depicted by a cross in the form of an **X** at the end of a timeline.



# Carpark Example

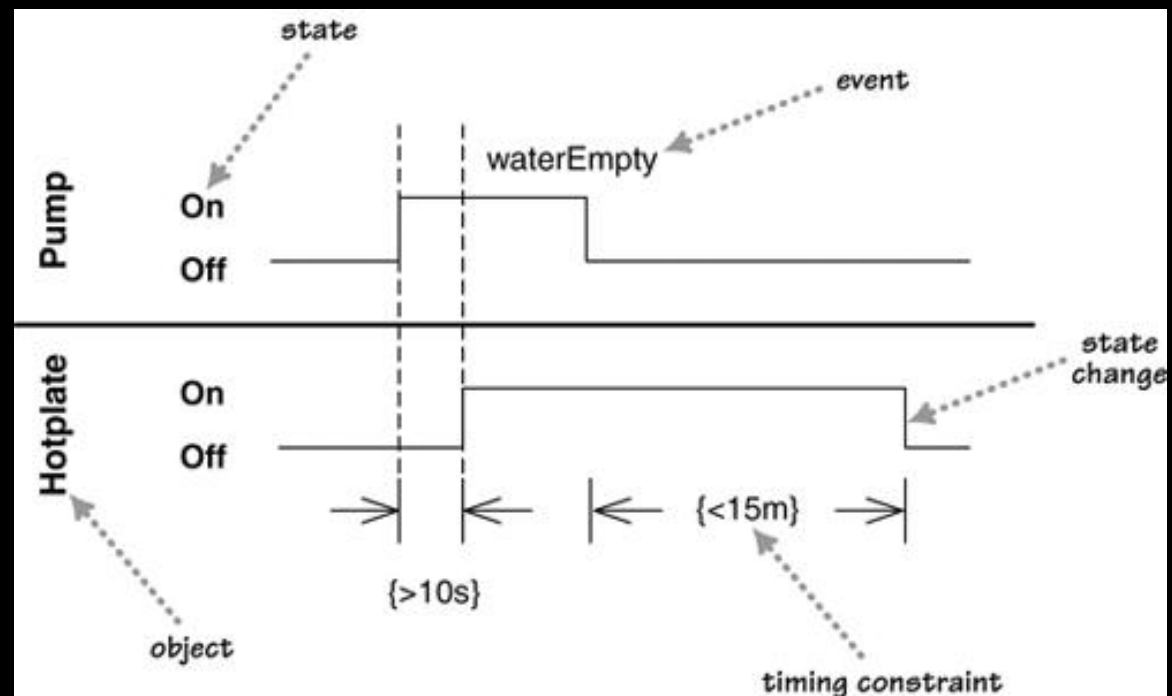


# Carpark Example



# Ex 1: Coffee Pot

- Let's take a simple scenario based on the pump and hotplate for a coffee pot. Let's imagine a rule that says that at least 10 seconds must pass between the pump coming on and the hotplate coming on. When the water reservoir becomes empty, the pump switches off, and the hotplate cannot stay on for more than 15 minutes more.



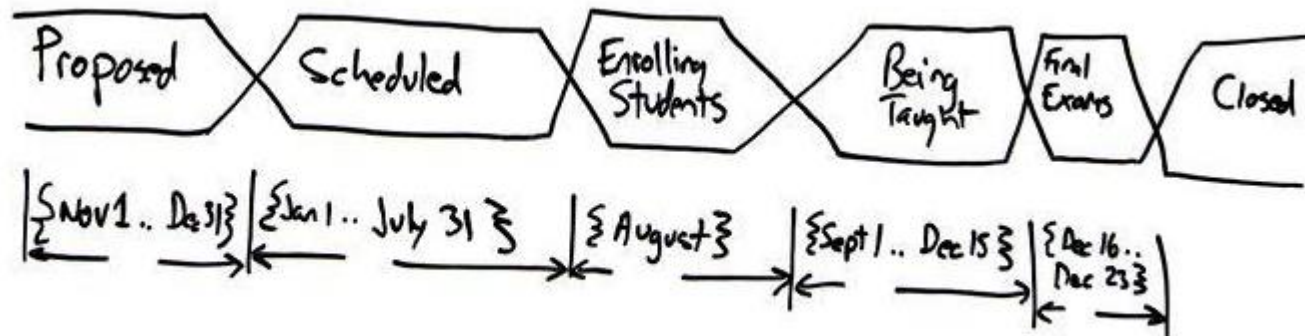
# Timing Diagram Notations

- There are two basic flavors of timing diagram,
  - Concise notation
  - Robust notation
- Showing states as lines – They are known as **State Lifeline diagram** or robust notations
- Showing states as areas - They are known as **Value Lifeline diagram** or concise notations

# Timing diagram-concise notation

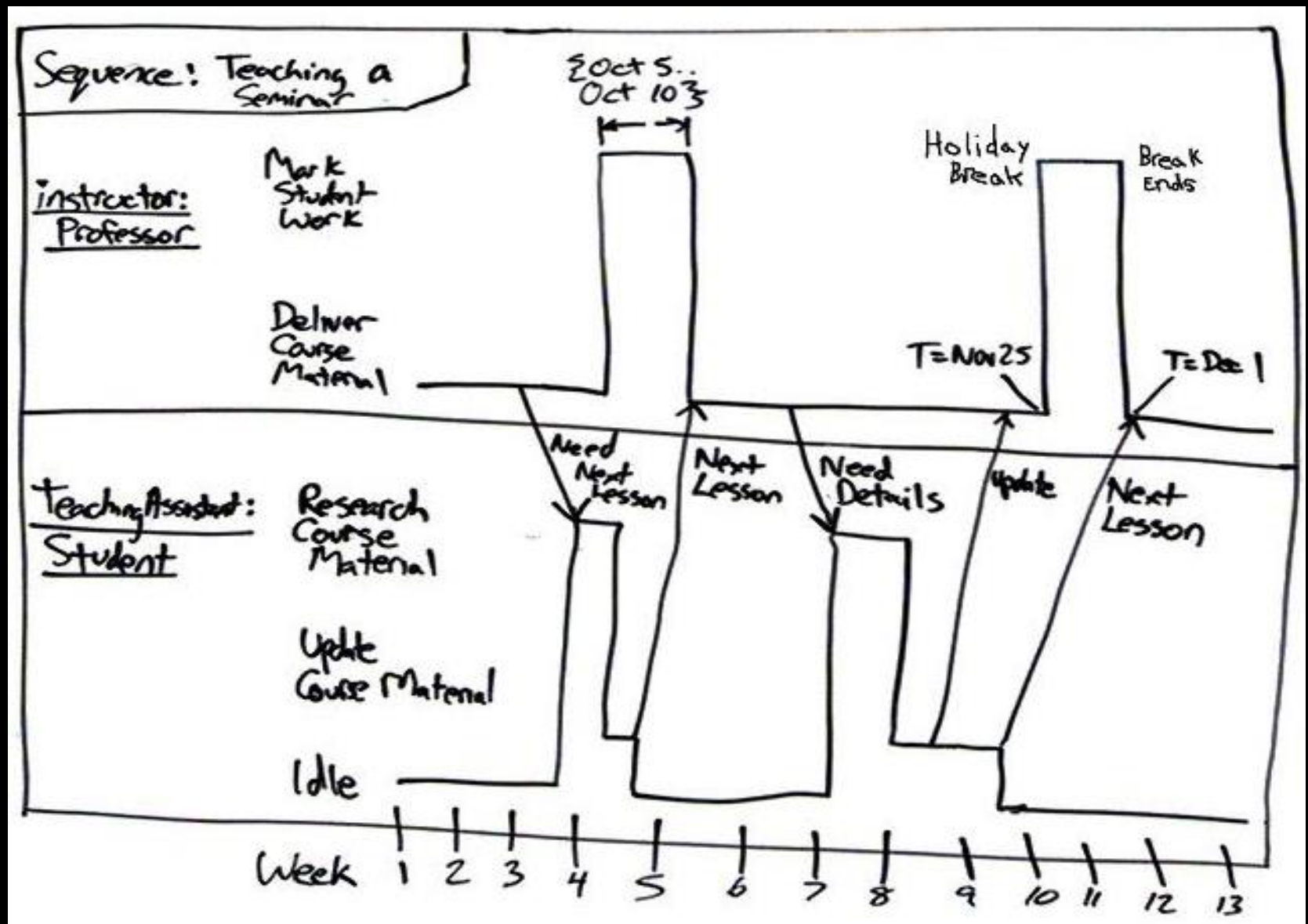
- Life Cycle of Seminar
- The critical states that the seminar exhibits
  - Proposed,
  - Scheduled,
  - Enrolling Students,
  - Being Taught,
  - Final Exams,
  - Closed

: Seminar



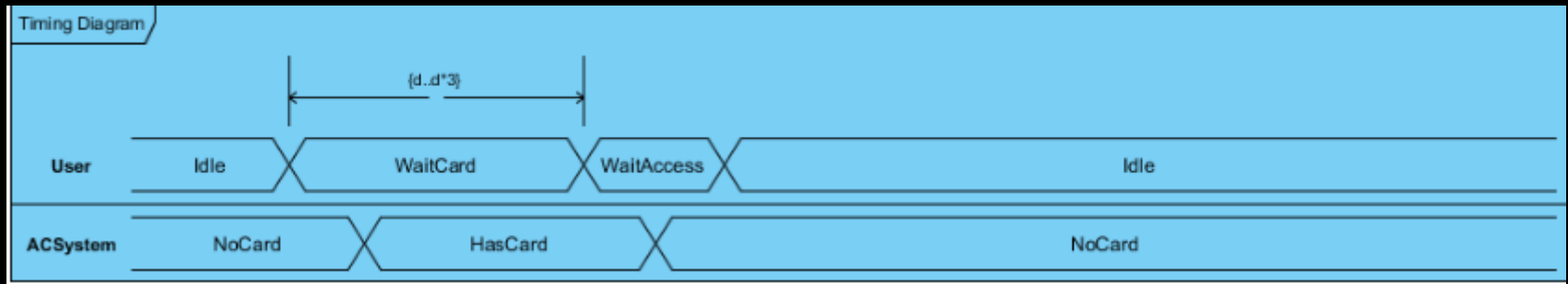
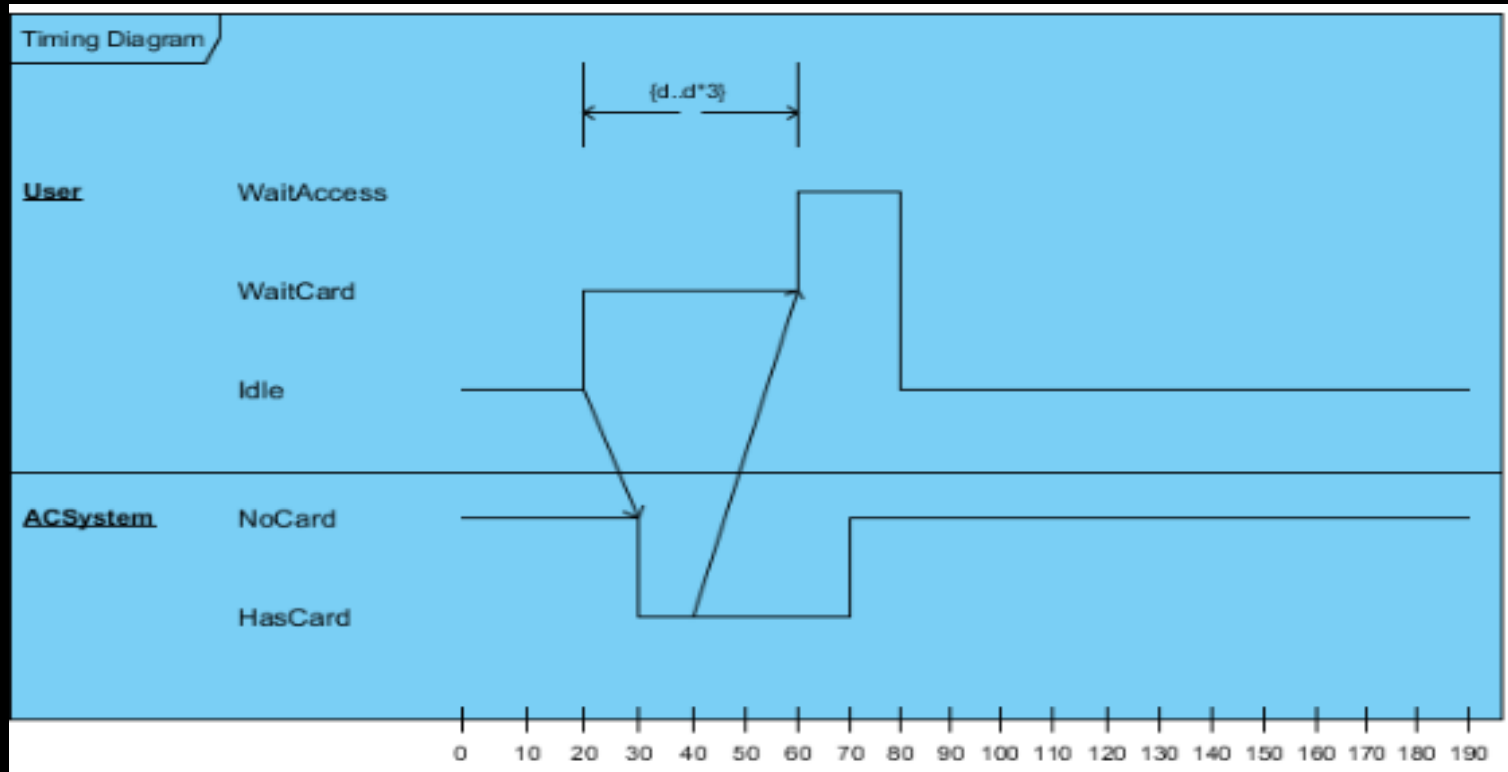
# Timing diagram-robust notation

- **Happenings while a seminar is being taught.**
- The professor delivers the seminar and marks student work
- The teaching assistant develops the course material by doing the research on course material and then update the material just in time for it to be taught.
- Total duration of seminar is 13 weeks. Events/stimuli, such as *Holiday Break* and *Break Ends* indicate the reason for the change.
- The arrows between timelines are messages between the objects.





# Another Example

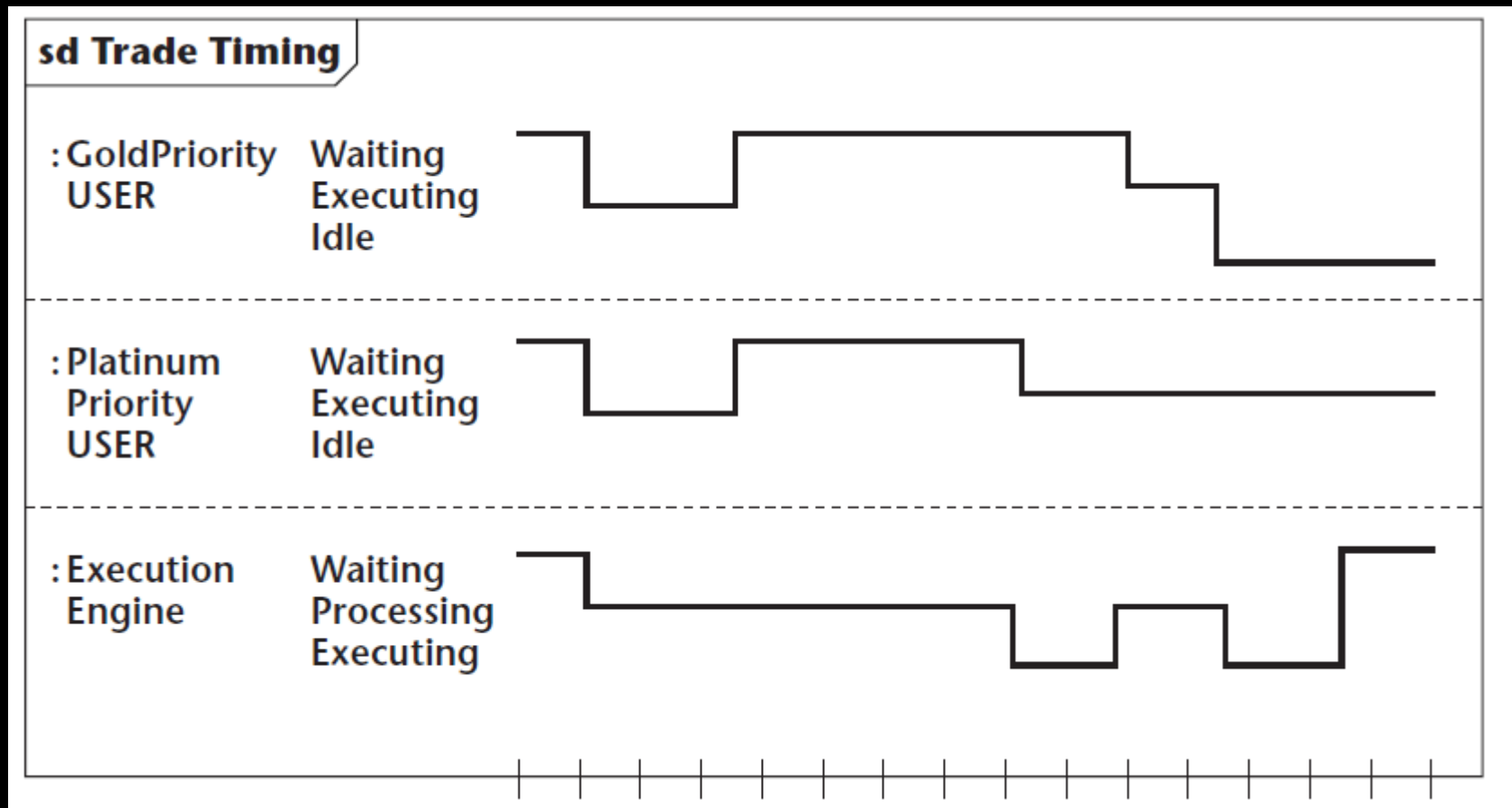


# Example 1

- Figure 6.19 shows a case where two active objects share a common resource. In this case, both objects show a trade that requires the execution engine for some time. One object is for a “platinum” user who is guaranteed a trade within 10 time units, and the other is a “gold” user who has no performance guarantee. The user objects have a waiting and executing state along with an idle state. The execution engine has a processing state that takes five time units for each trade and an executing state that takes two time units.

# Example 1

- Different priority objects shown on a timing diagram.

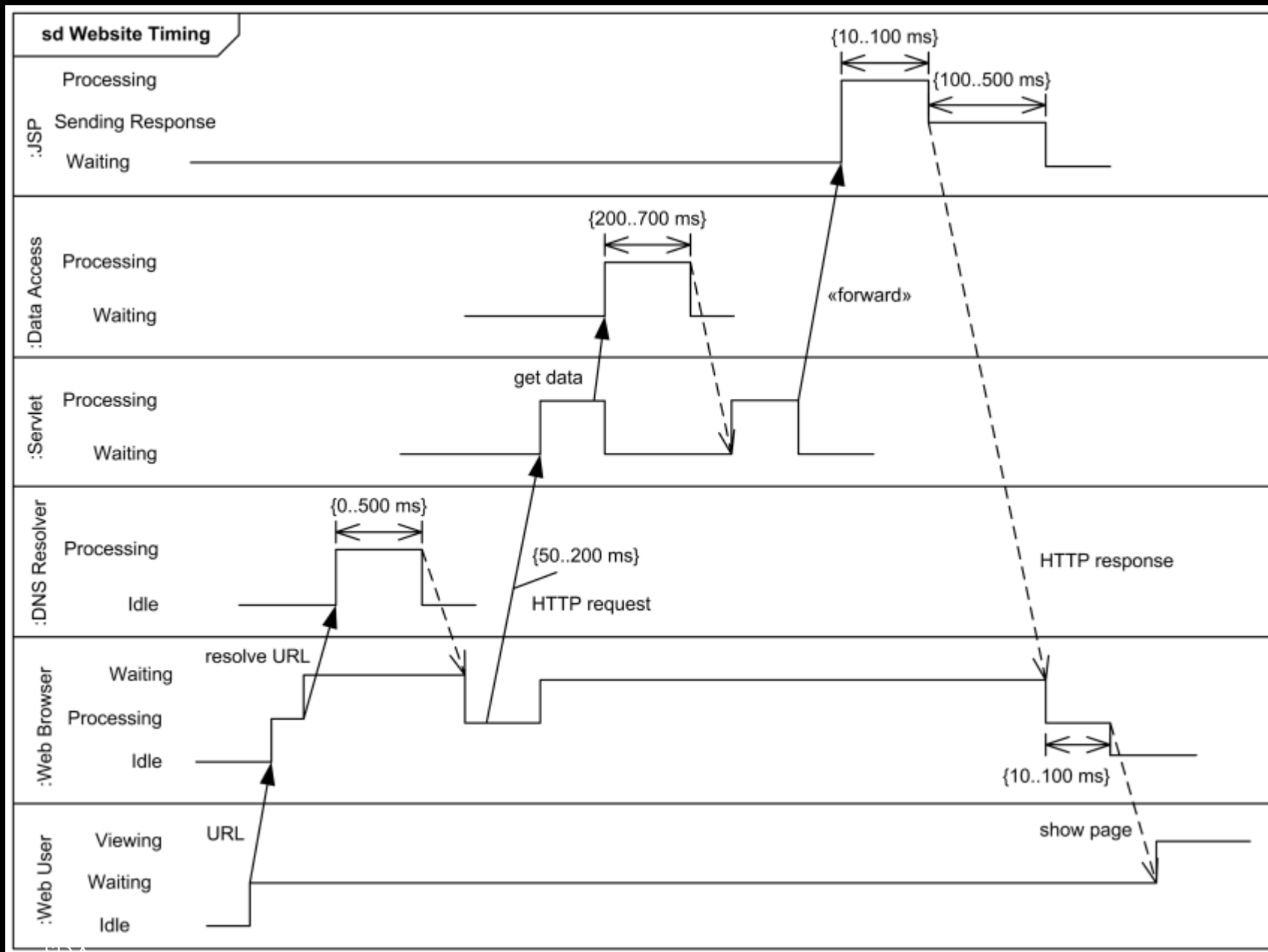


# Ex 2: Website Latency

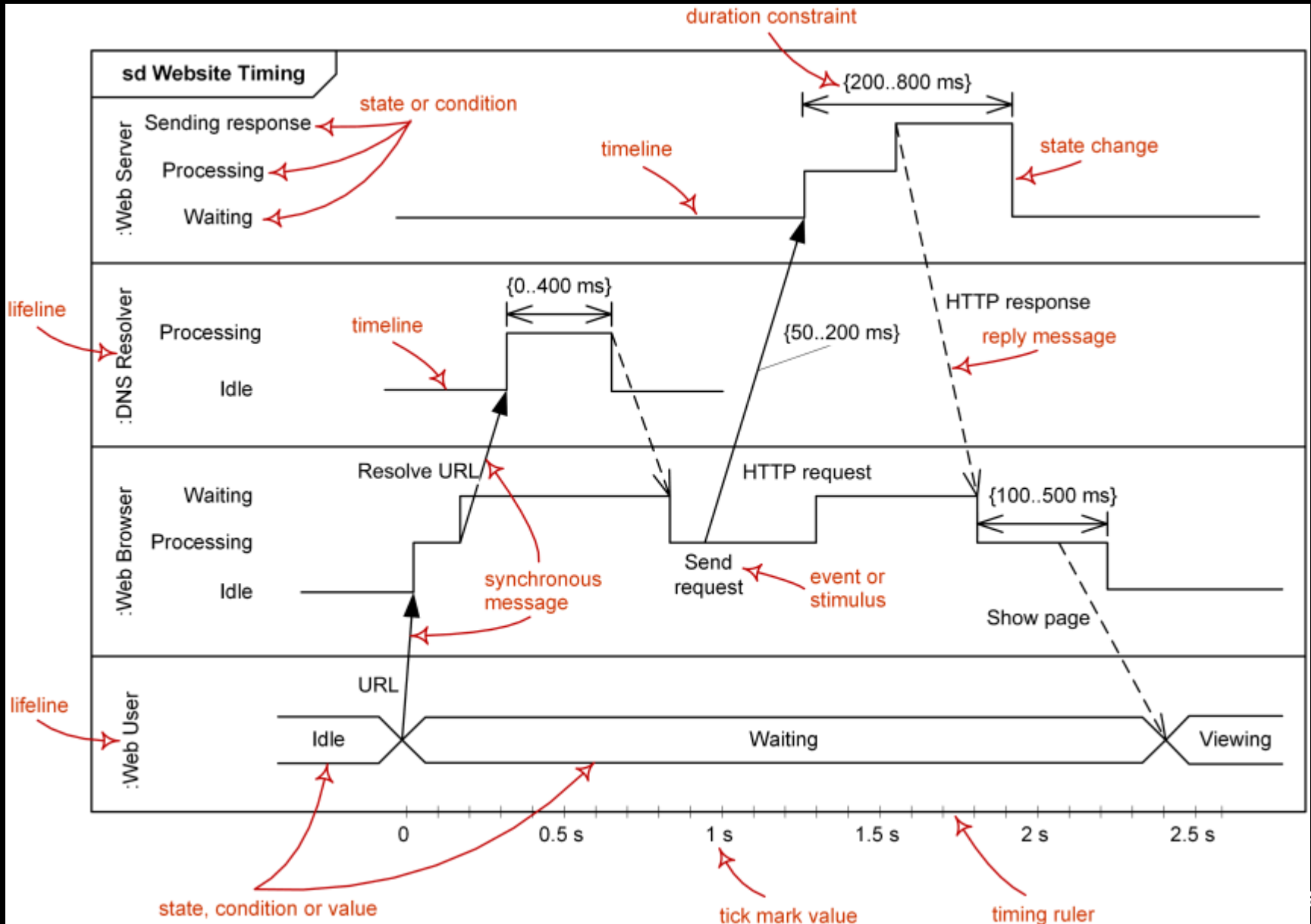
- An example of **timing diagram** which shows some **duration constraints** for a fabricated website to evaluate how long web user should wait to see something rendered on his/her display.
- After web user enters web page URL, the URL should be resolved to some IP address. DNS resolution can add some tangible waiting time to the response latency as perceived by user. Latency delays related to DNS resolution could range from 1 ms (local DNS cache) to several seconds.

# Ex 2: Website Latency

- With simple Model–View–Control (MVC) implementation, Java servlet gets control and requests some data from "model". Communication with data sources usually takes some discernible time. After data is received and processed, servlet forwards request processing to JSP ("view"). Buffered HTTP response is sent back to the browser.
- Web browser takes some time to process HTTP response and HTML page to start rendering the page view to the web client. (Note, that after that it could take even more time for the web browser to request other resources like CSS, JavaScript, images, which is not shown on the diagram.)



# Ex2 : Notations

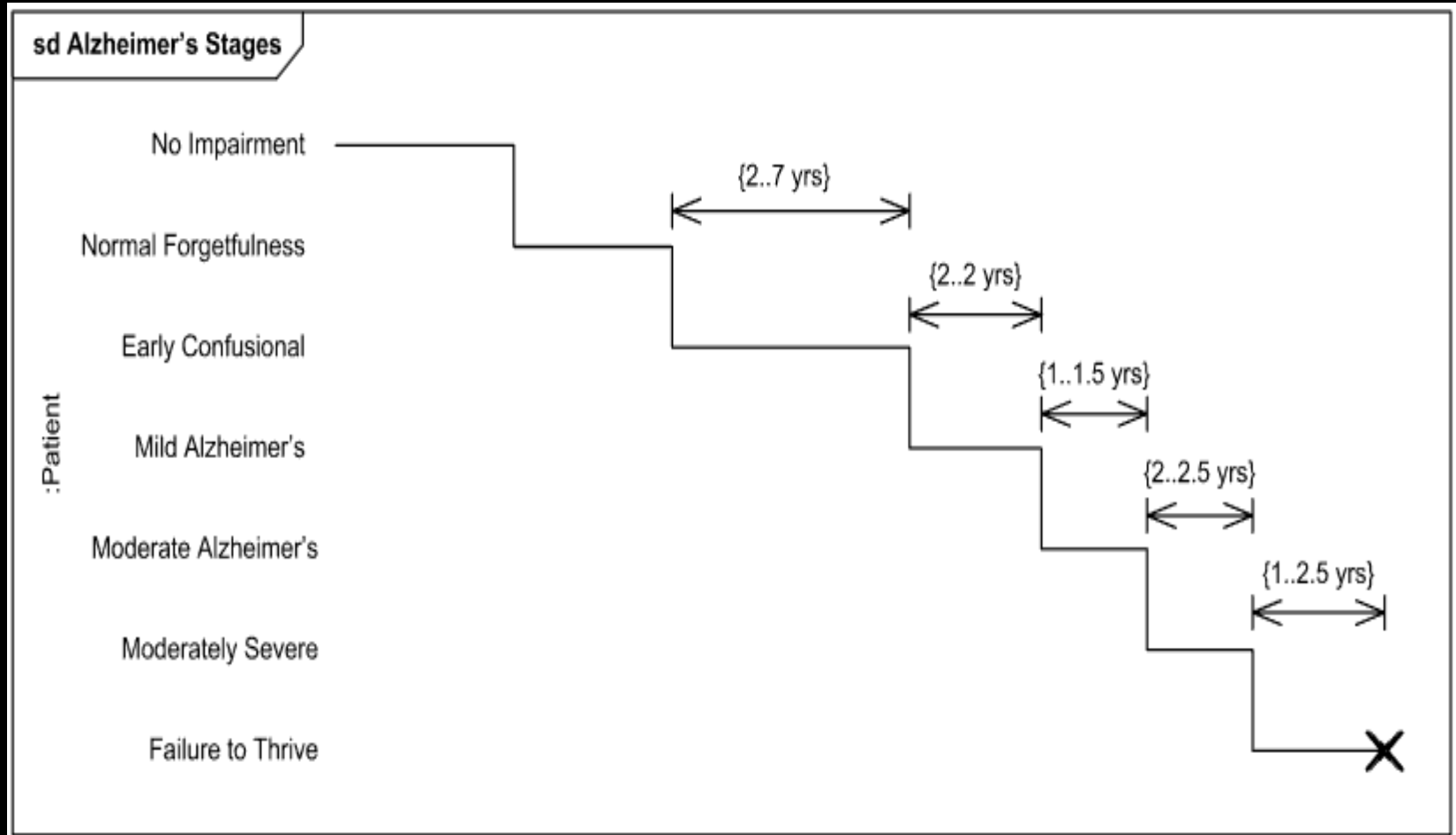


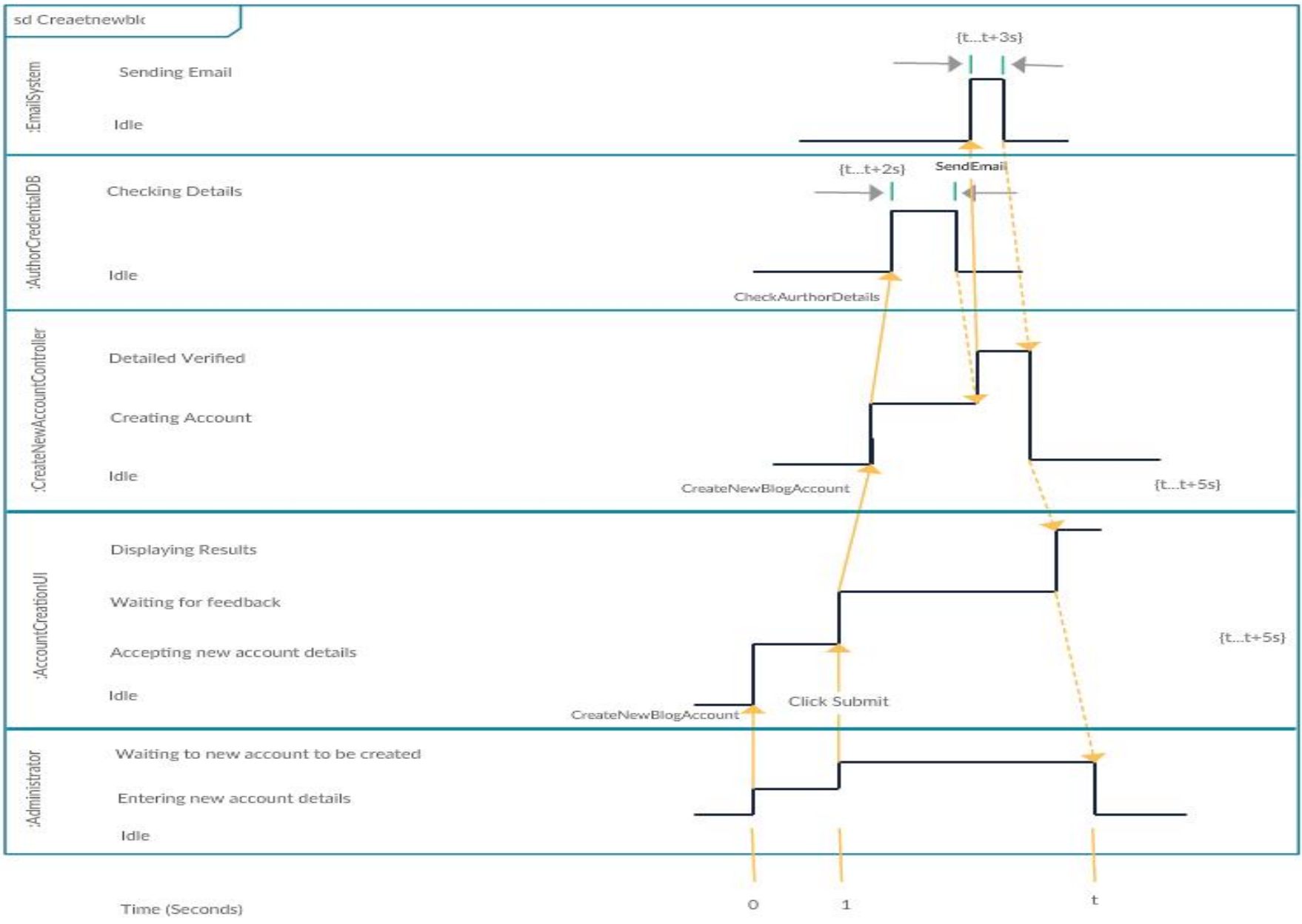
# Ex 3: Stages of Alzheimer's Disease

- **Alzheimer's disease** (AD) is a a progressive, ultimately fatal brain disease that causes loss of memory and intellectual abilities. The cause of the disease is unknown. AD has no cure and is one of the leading causes of death in the United States.
- For Alzheimer's disease doctor may use a diagnostic framework with three to seven levels (stages). Progression through these stages may last from 8 to 10 years, and in some cases up to 20 years from the time neuron changes start.
- Example of timing diagram below shows timing for the seven stage framework.



# Ex 3: Stages of Alzheimer's Disease

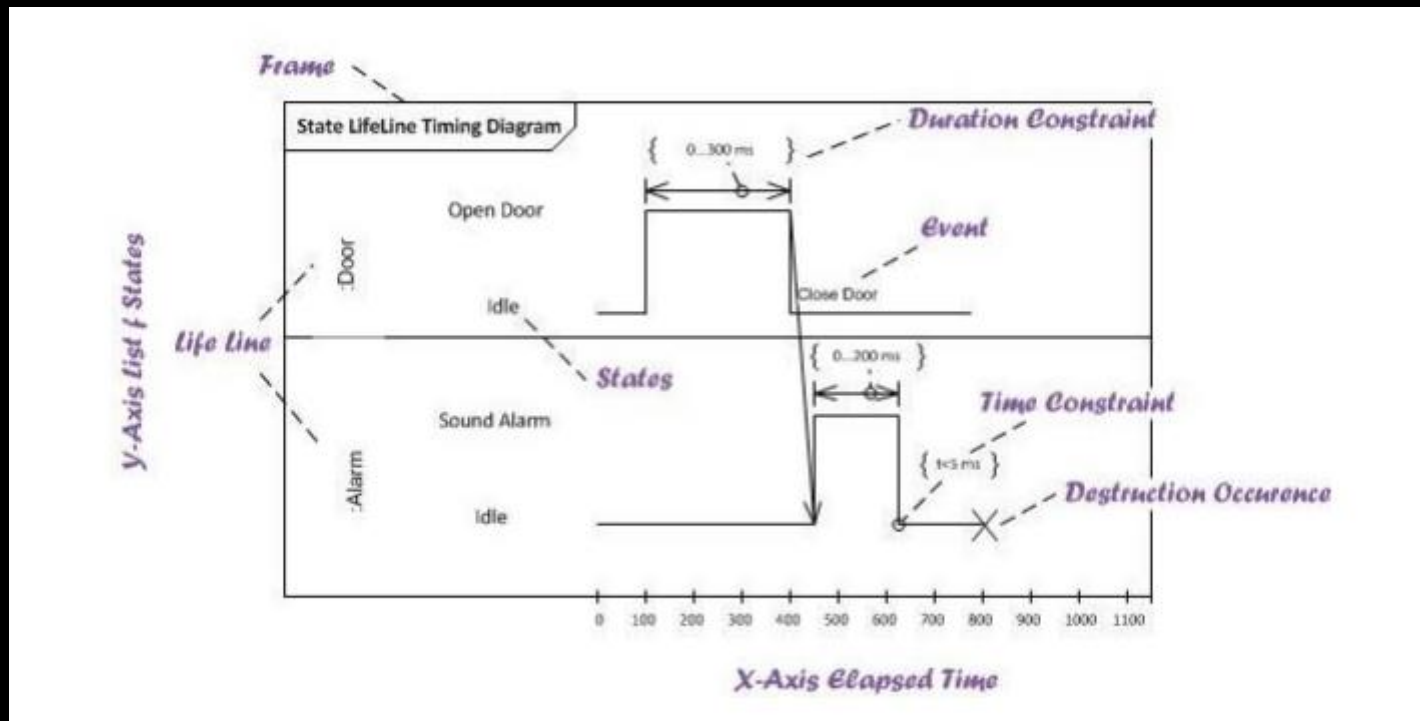




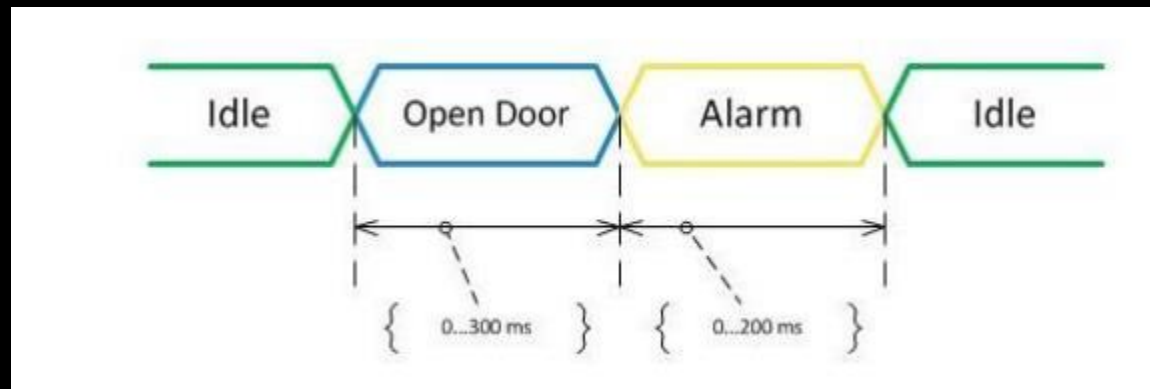
# Example

- Let us understand timing diagram with an example. I have an automatic door, it stays open for 300 ms, after that it is programmed to close itself. While closing it will raise an alarm for 200ms indicating others that the door is closing.

# State Life line



# Value lifeline





That is all