

Lecture 6

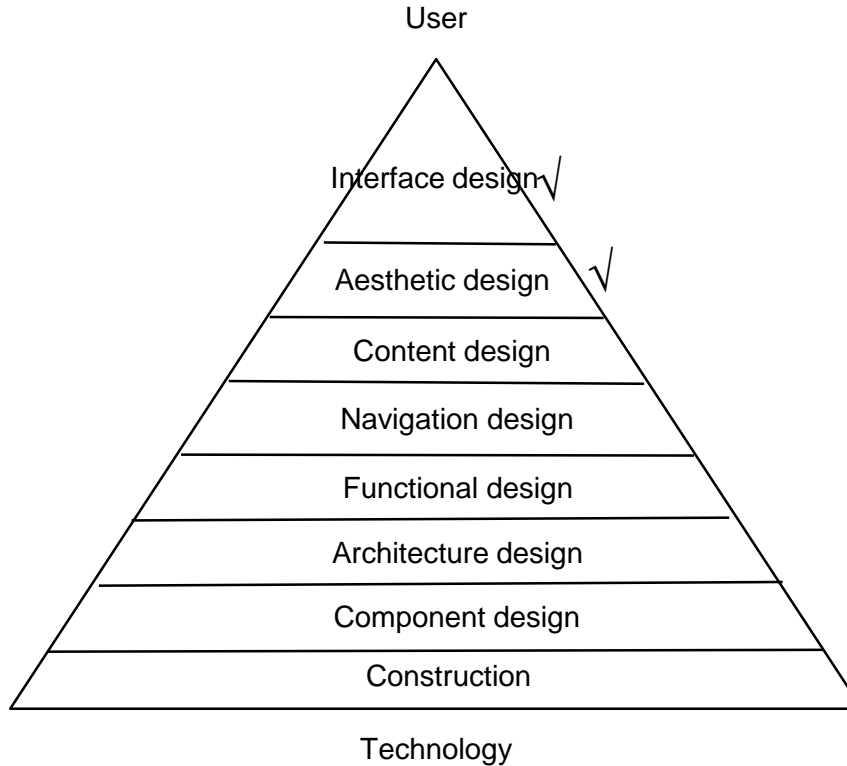
Design Modeling for WebApps

Outcomes of This Session

- ▶ Navigation design using UML profiling
- ▶ Planning in agile- scrum approaches



WebE Design Pyramid

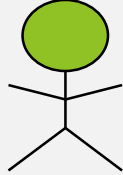
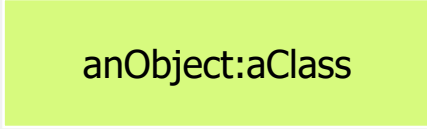






Content classes and sequence diagram

The interaction model indicates how software will respond to external events. To create the model, you should perform the following steps:

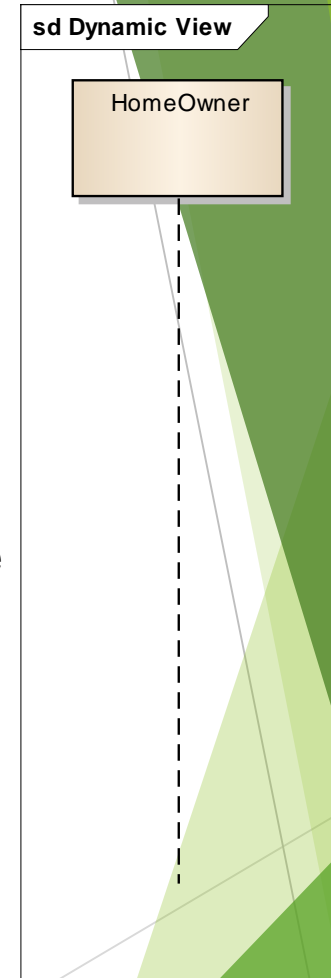
1. Evaluate all use cases to fully understand the sequence of interaction within the system.
2. Identify events that drive the interaction sequence and understand how these events relate to specific objects.
3. Identify classes and its operations
4. Create a sequence diagram for each use case.
5. Review the models to verify accuracy and consistency.

Sequence diagrams syntax

AN ACTOR	
AN OBJECT	
A LIFELINE	
A FOCUS OF CONTROL	
A MESSAGE	
OBJECT DESTRUCTION	

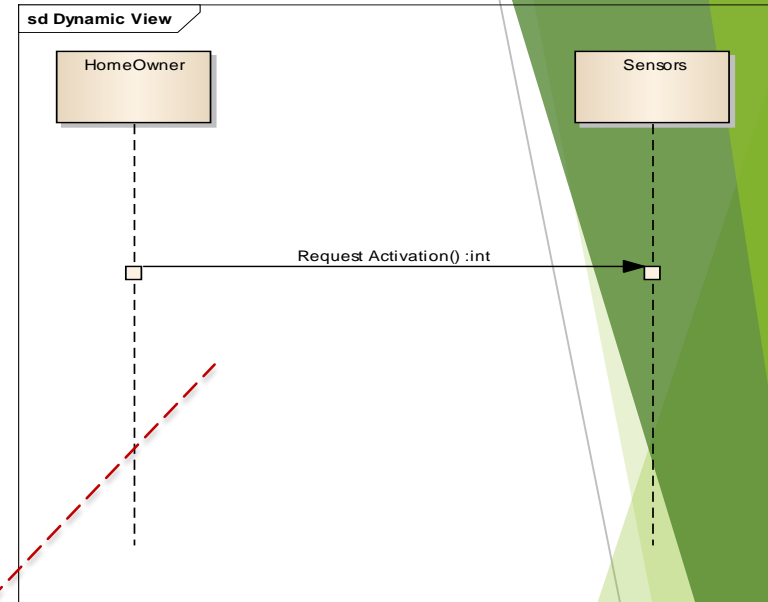
Lifeline

- Lifelines
- When drawing a sequence diagram, lifeline notation elements are placed across the top of the diagram.
- Lifelines represent either roles or object instances that participate in the sequence being modeled.
- Lifelines are drawn as a box with a dashed line descending from the center of the bottom edge (Figure 3).
- The lifeline's name is placed inside the box.



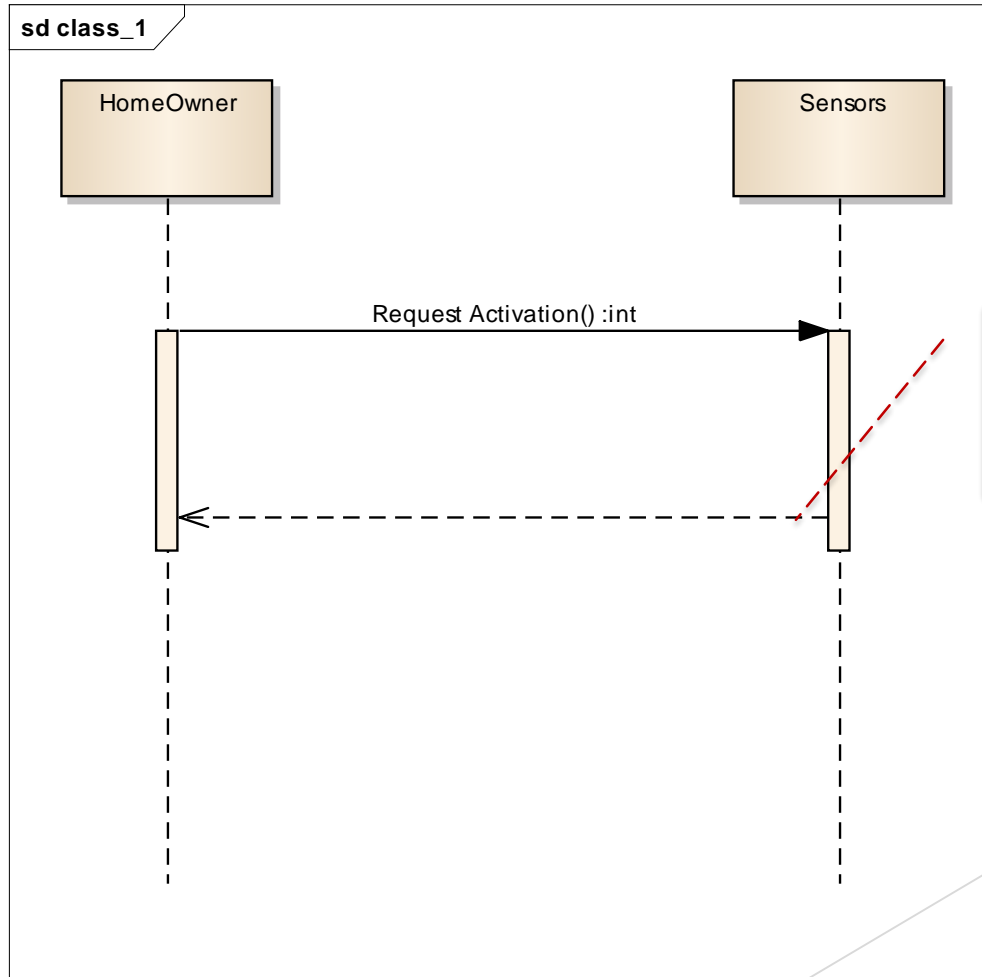
Message

- The message
- A message is sent from sender to receiver, directed line with solid arrow head
- The message represents an operation/method that the receiving object's class implements.
- HomeOwner is calling a method which has been implemented in the class of sensor object



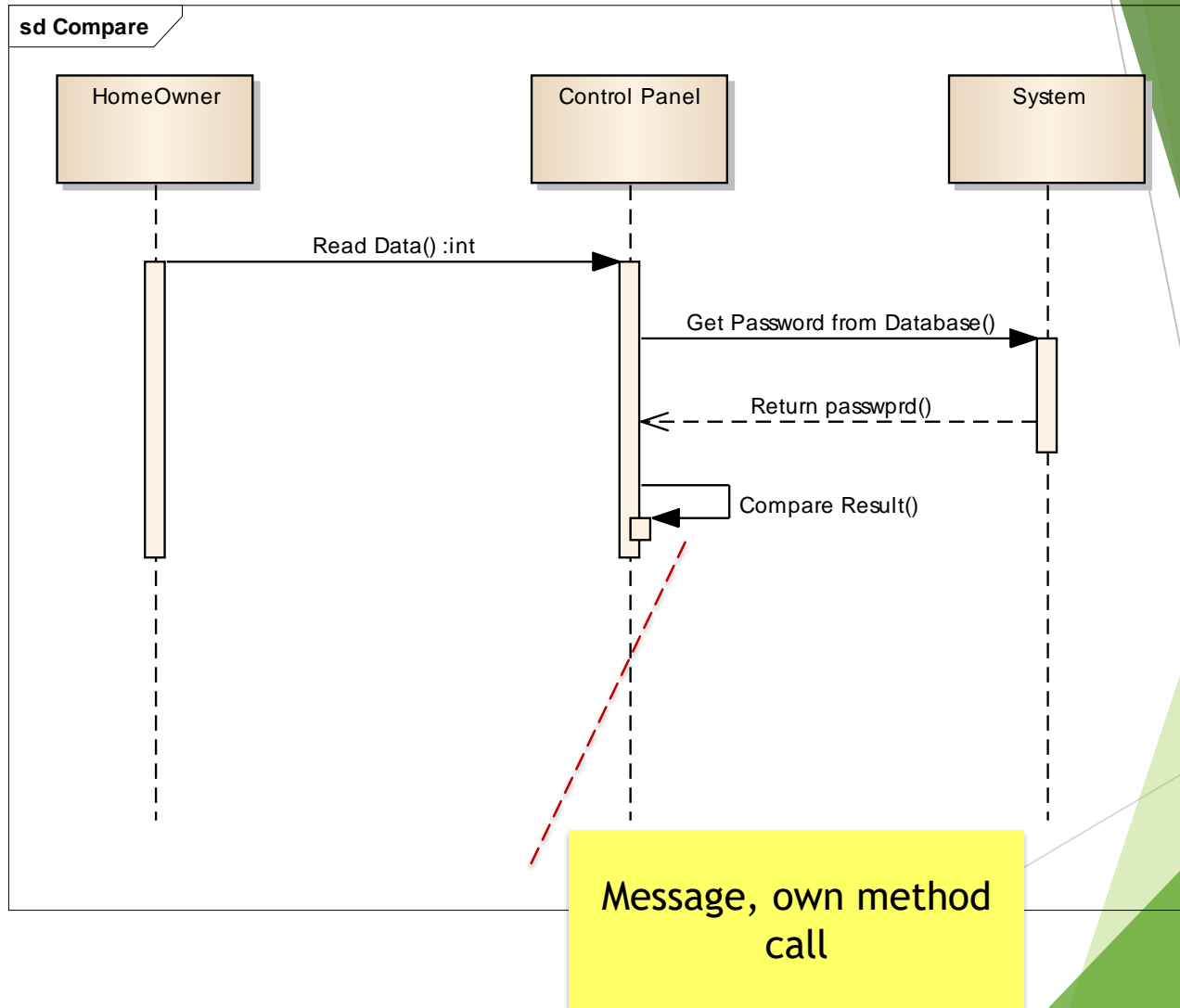
Message

Message Return Value

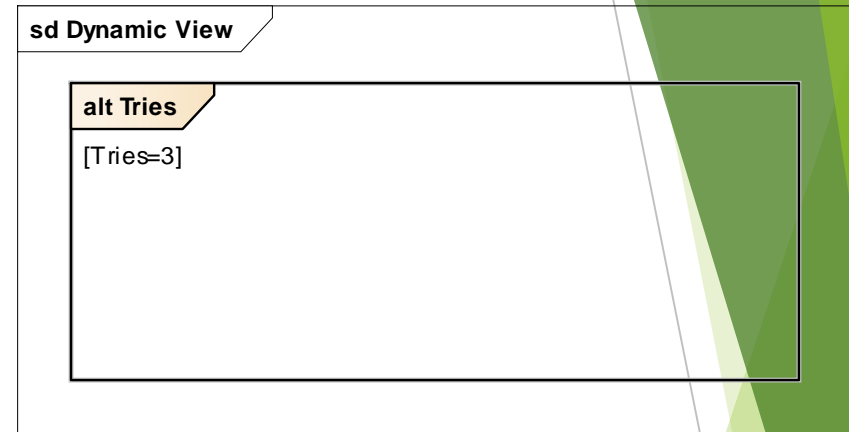
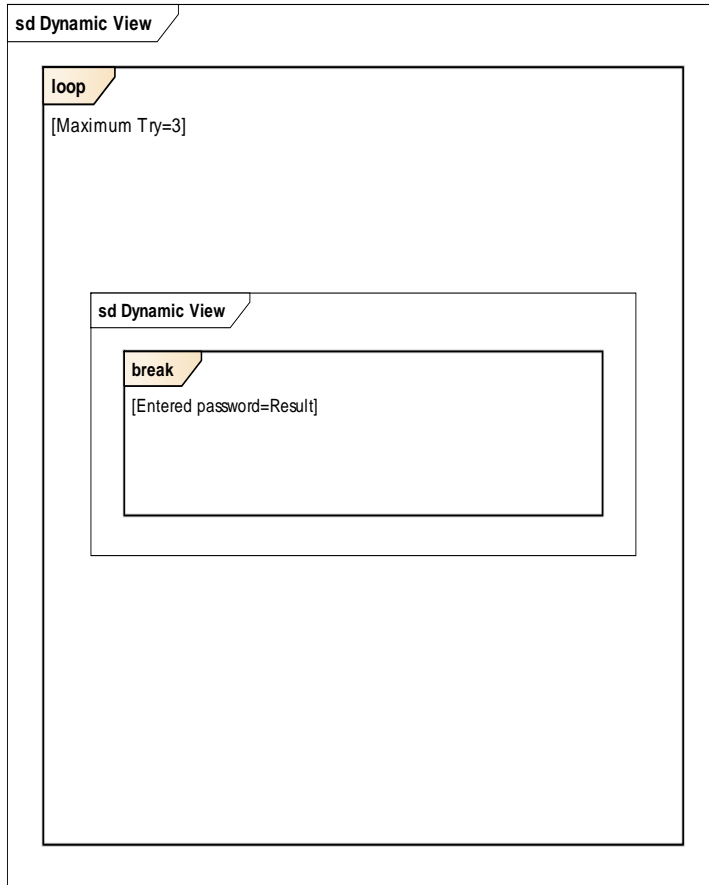


Returning values

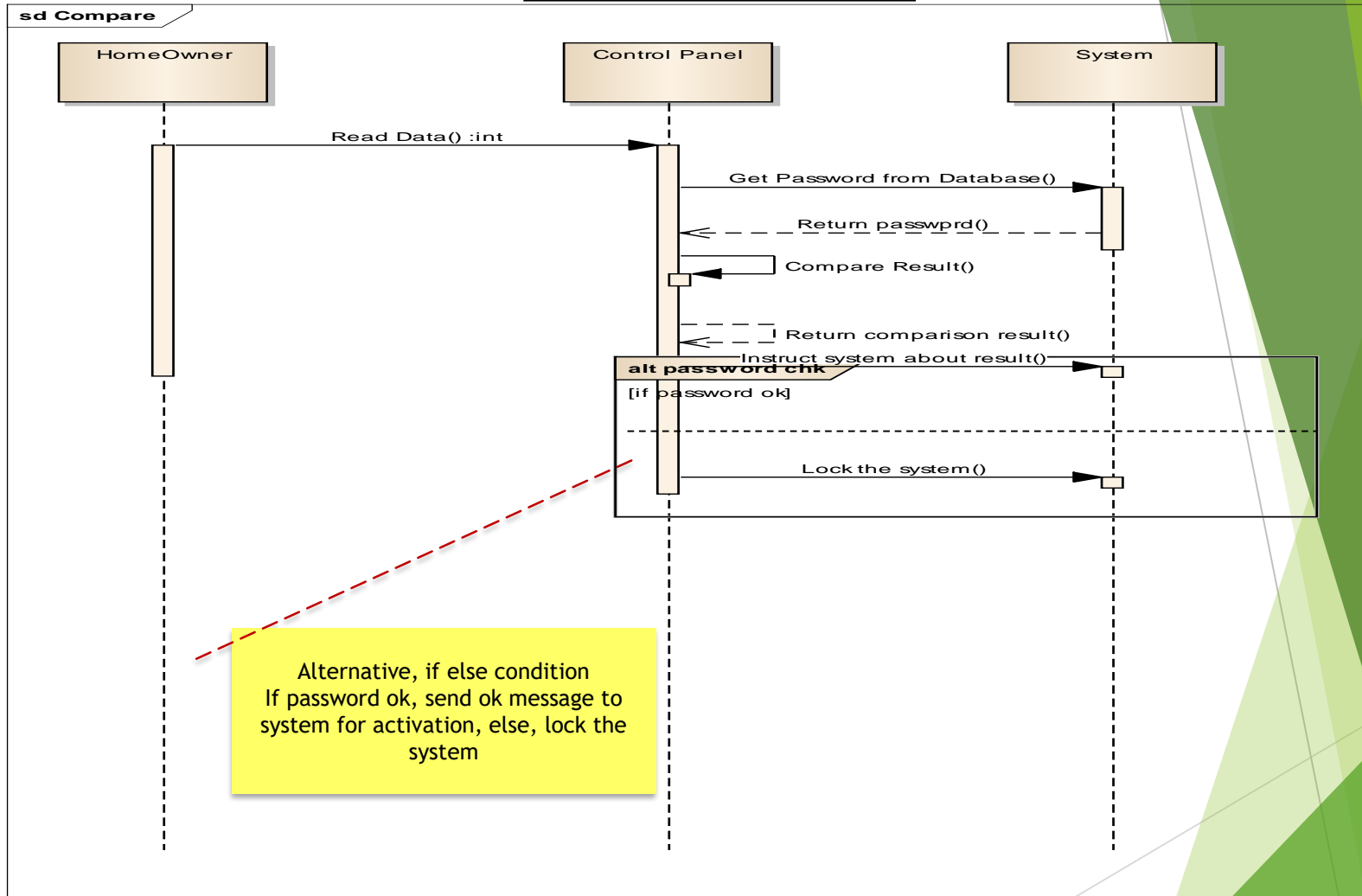
Self Message



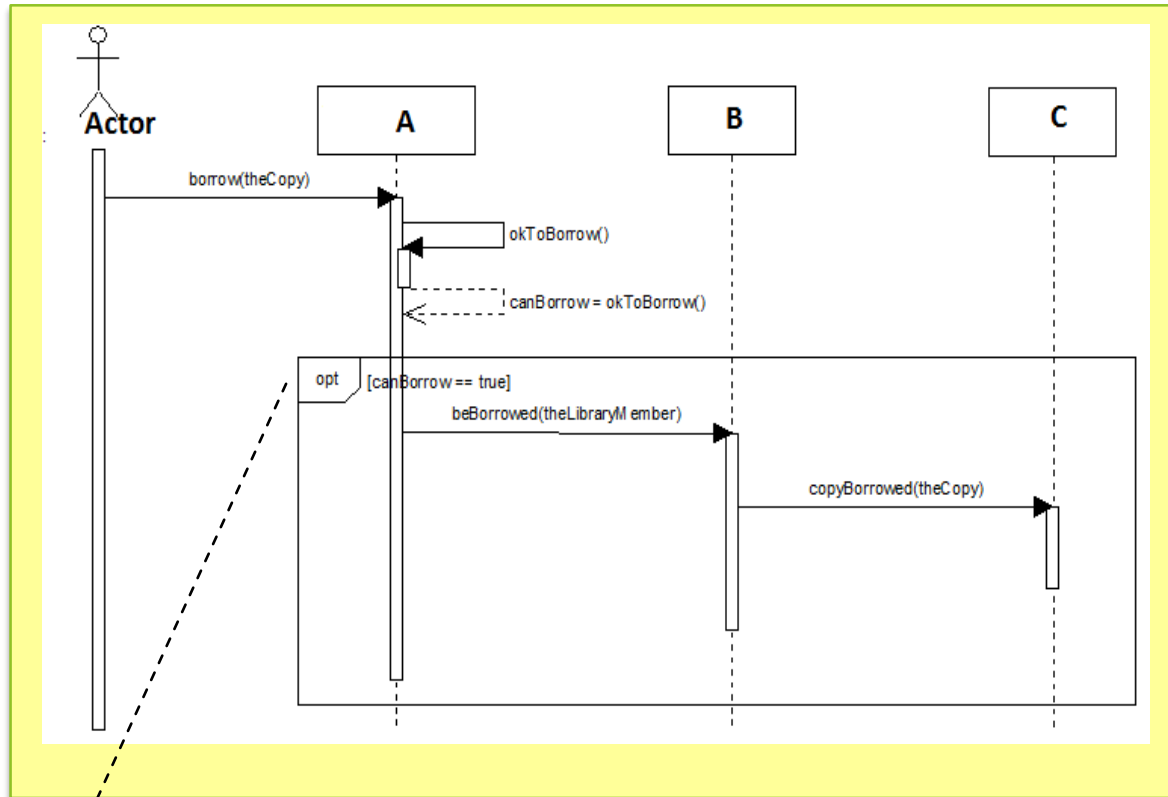
Alternatives, Loops and Break



Conditional Behaviour

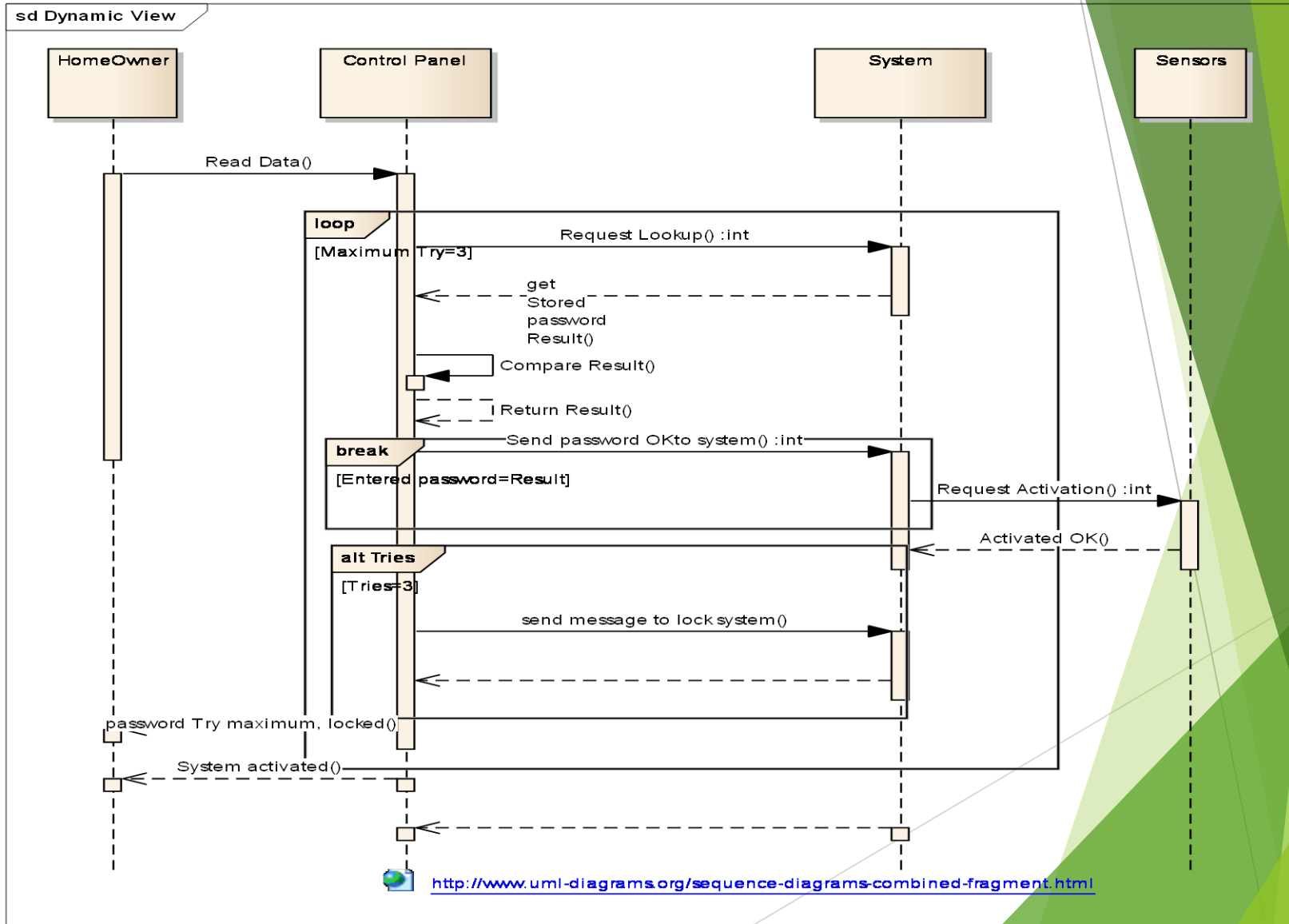


Conditional behaviour

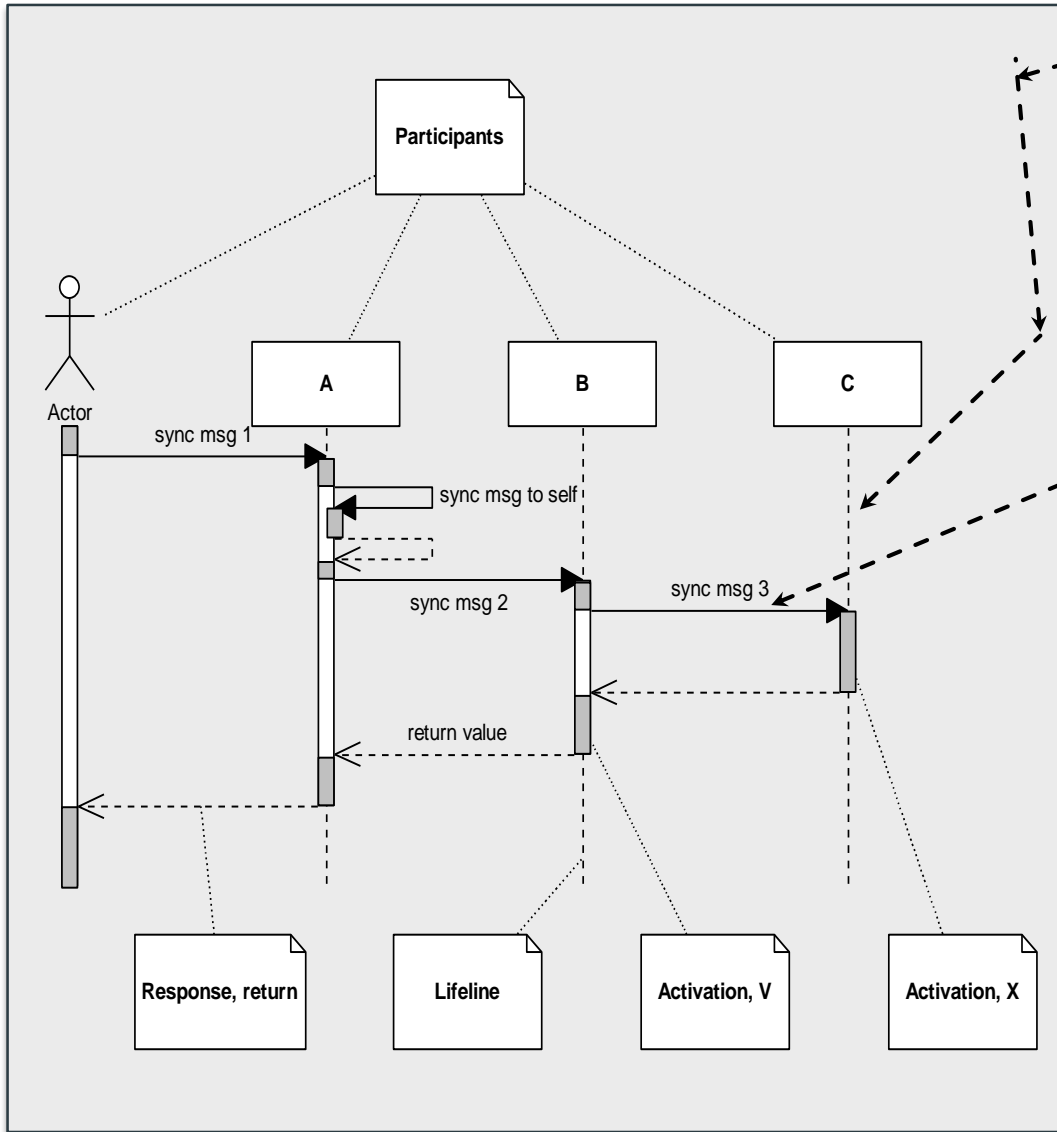


- ▶ Conditional behaviour enclosed within rectangle labelled “opt” (for “optional”)
- ▶ Guard condition written between square brackets near “opt” label
- ▶ The option combination fragment notation is similar to the alternation combination that it only allow one operand and there never can be an "else" guard
- ▶ Multiple action can be put in one fragment

Example of loop and Break

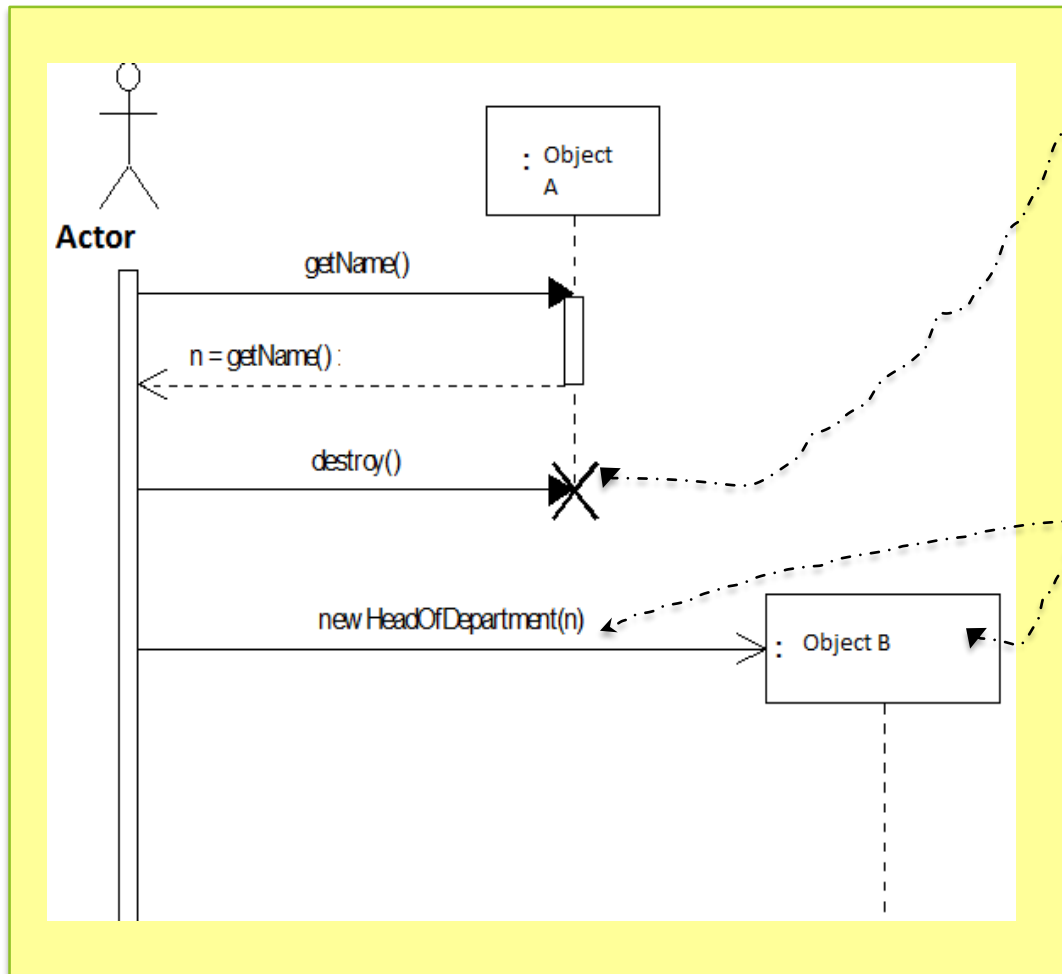


Synchronous Message



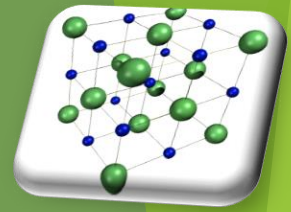
- ❖ Lifeline- Extending down the page from each participant is a dashed line
- ❖ Time is understood to move forward as we move down the diagram
- ❖ Message: A message sent from participant A to participant B is represented by an arrow with a solid line drawn from the lifeline of A to the lifeline of B (sync msg 2)
- ❖ Synchronous Message: If A has to stop computing while B carries out the operation invoked by the message sent to it by A, then this message is a *synchronous* message and control is transferred from A to B
- ❖ A synchronous message symbol:
—————→

Asynchronous Message



- ▶ Object destruction indicated by a synchronous message sent to the object whose lifeline is terminated with an X
- ▶ Construction of new object indicated by horizontal asynchronous message arrow directly to the head of the new participant
- ▶ Asynchronous message arrow has stick head
 - ▶ Synchronous message arrow has solid black triangular head
- ▶ Asynchronous message starts a new thread of control
 - ▶ Participant that sends asynchronous message continues computing after sending message

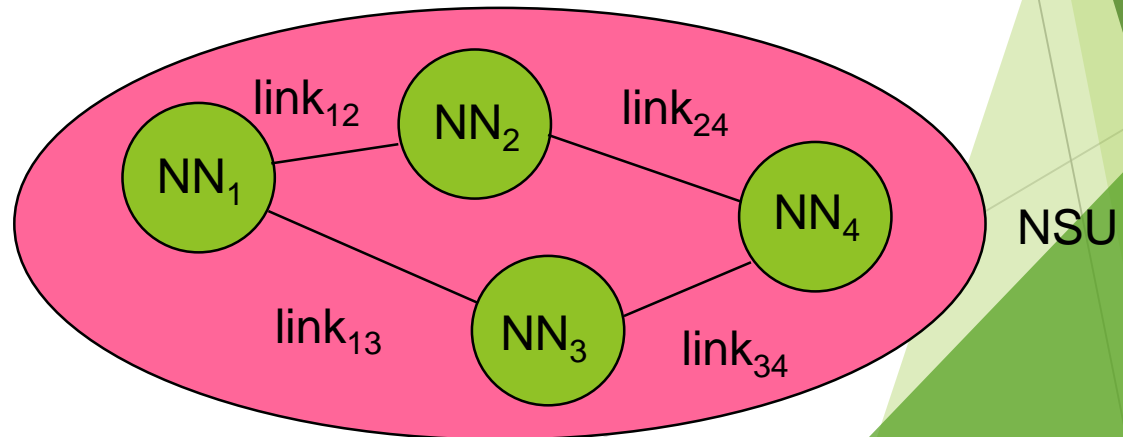
Navigation Design



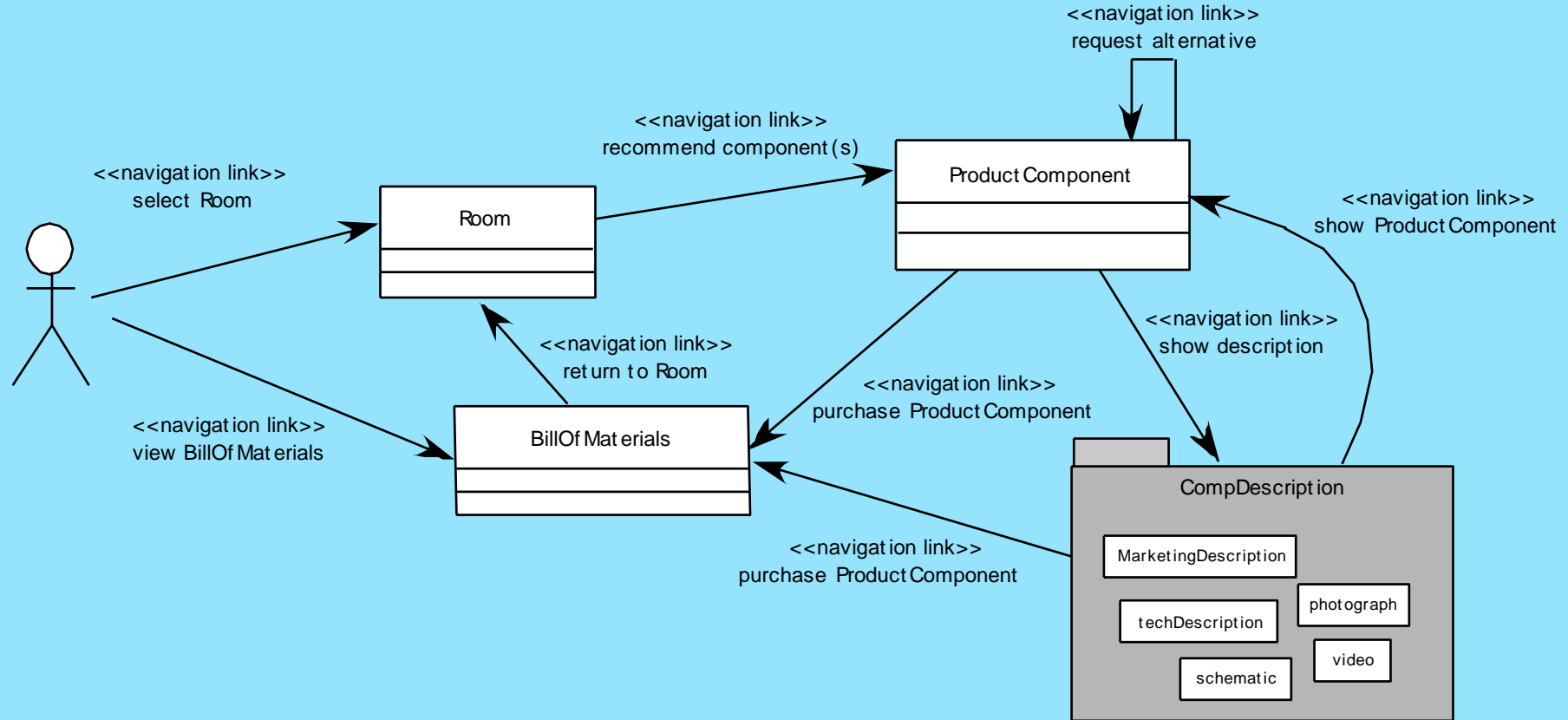
- ▶ Begins with analyzing the user hierarchy and related use-cases.
 - ▶ Each actor may use the WebApp somewhat differently and therefore have different navigation requirements.
- ▶ As interaction with web application occurs, a series of Navigation Semantic Units (NSUs) are encountered.
- ▶ A NSU describes the navigation requirements for each use case and each user role.
- ▶ The NSU shows how an actor moves between content objects or web application functions.

Navigation Semantic Units NSU

- ▶ Navigation semantic unit
 - ▶ Ways of navigation (WoN)— this represents the best navigation way or path for users.
 - ▶ E.g. best pathway to get from product page to checkout.
- ▶ It is composed of ...
 - ▶ Navigation nodes and the links that enable navigation between them.



Creating an NSU (example)



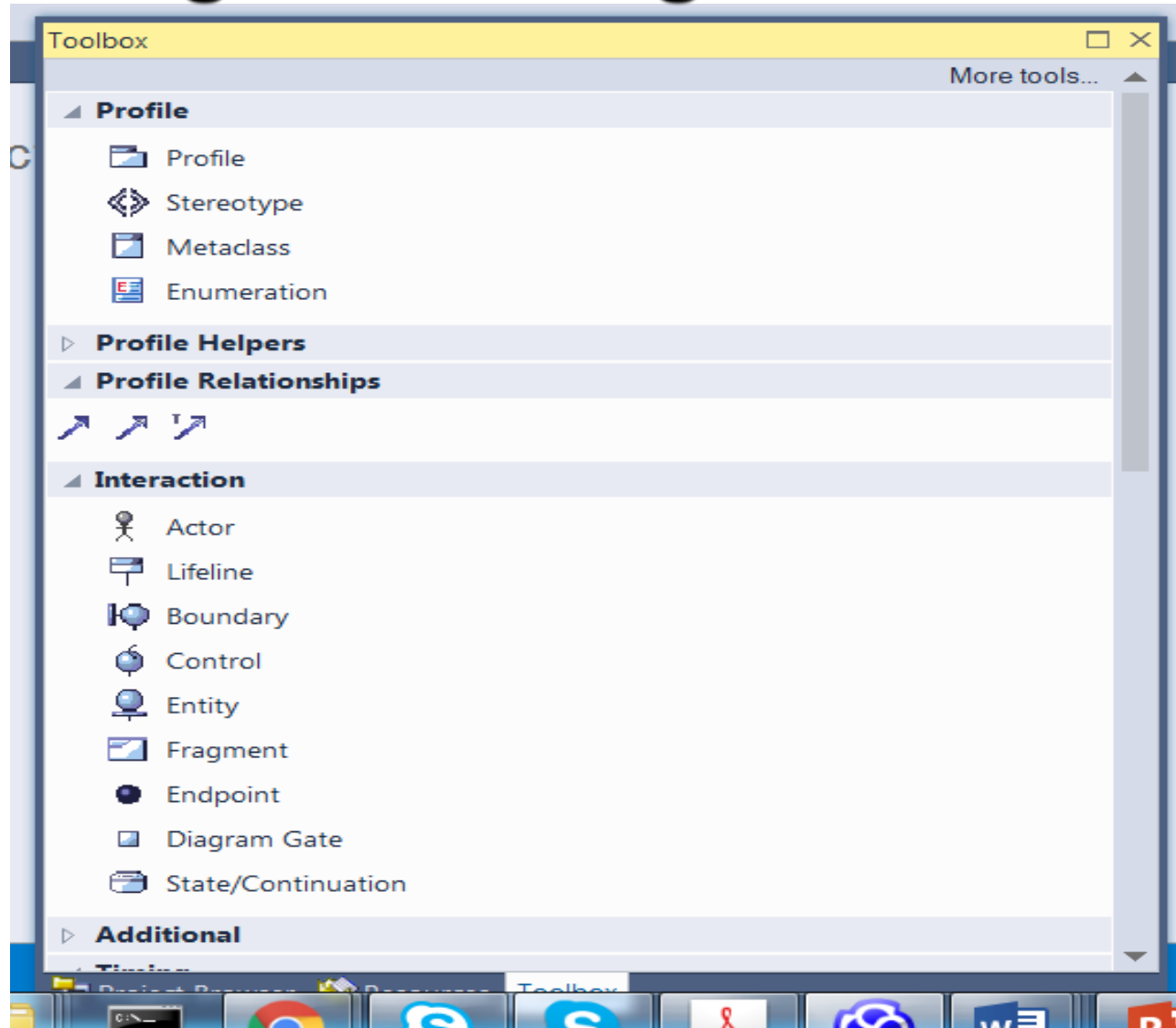
This navigation can be designed from a sequence diagram representing user interaction scenario in use cases

moves between content objects or web application functions.

Navigation Design: UML Profile

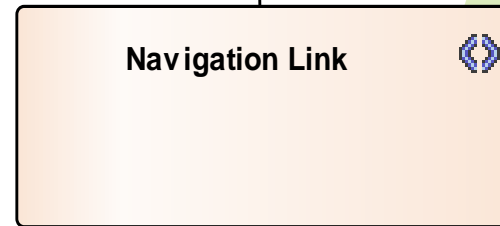
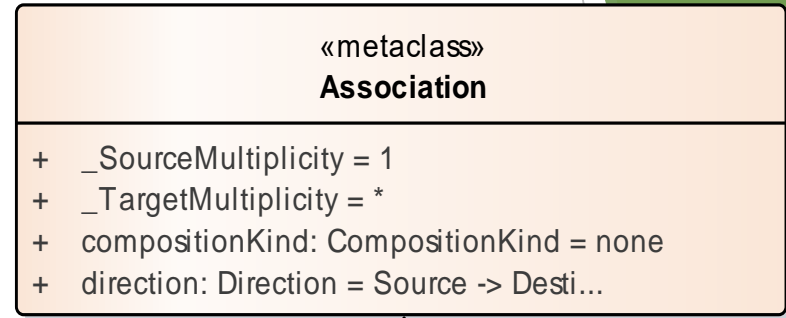
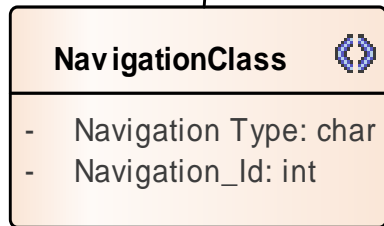
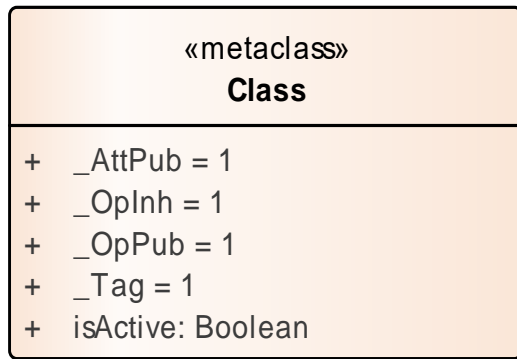
- UML Profiles provide a generic extension mechanism for building UML models in particular domains.
- They are based on additional Stereotypes and Tagged values
- These can be applied to Elements, Attributes, Methods, Links, Link Ends and more.
- A profile is a collection of such extensions that together describe
 - some particular modeling problem
 - and facilitate modeling constructs in that domain.
- More Reference: Enterprise Architect (sparx Systems)

Navigation Design: UML Profile in EA

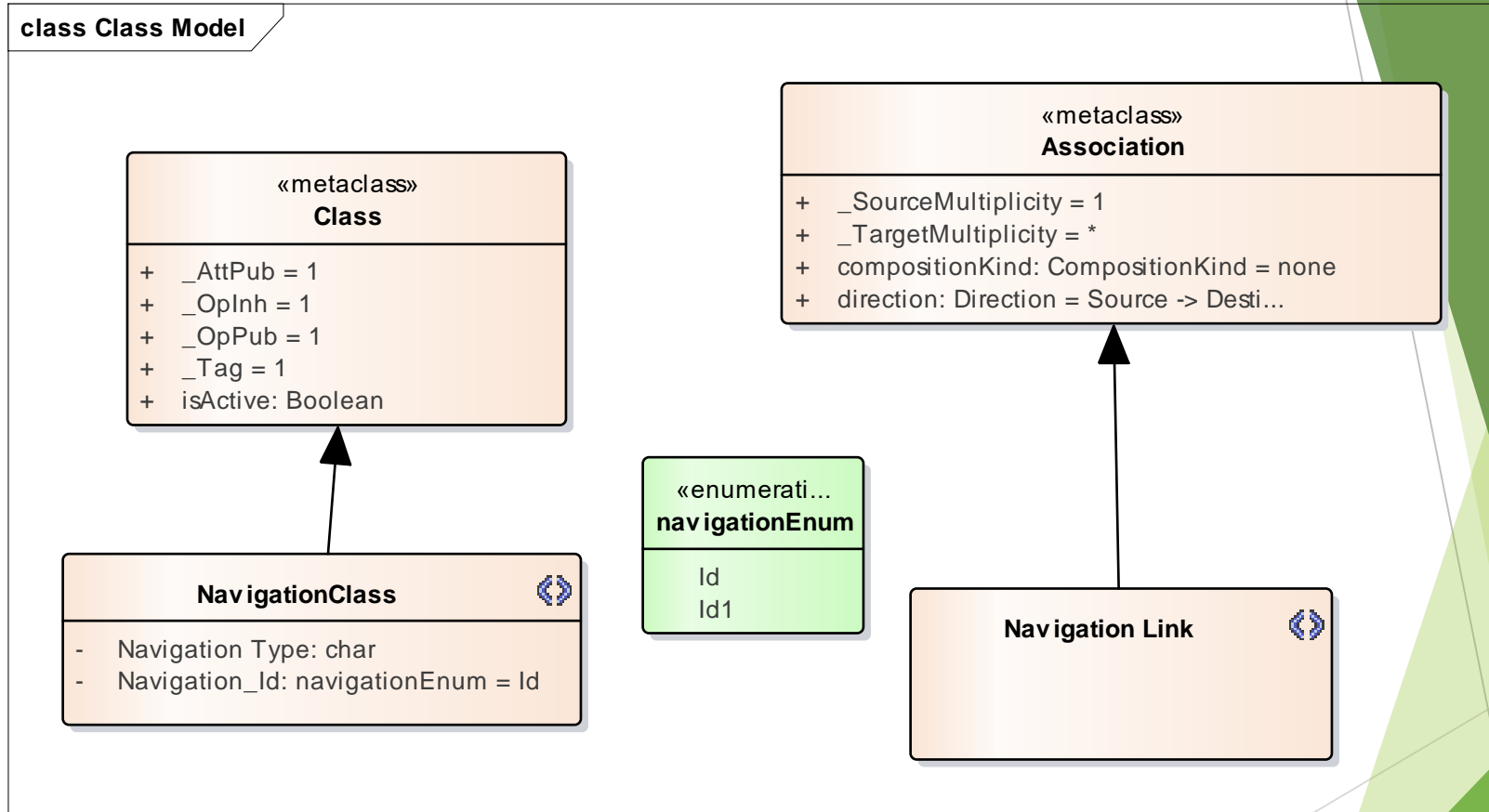


Navigation Design: UML Profile

class Class Model

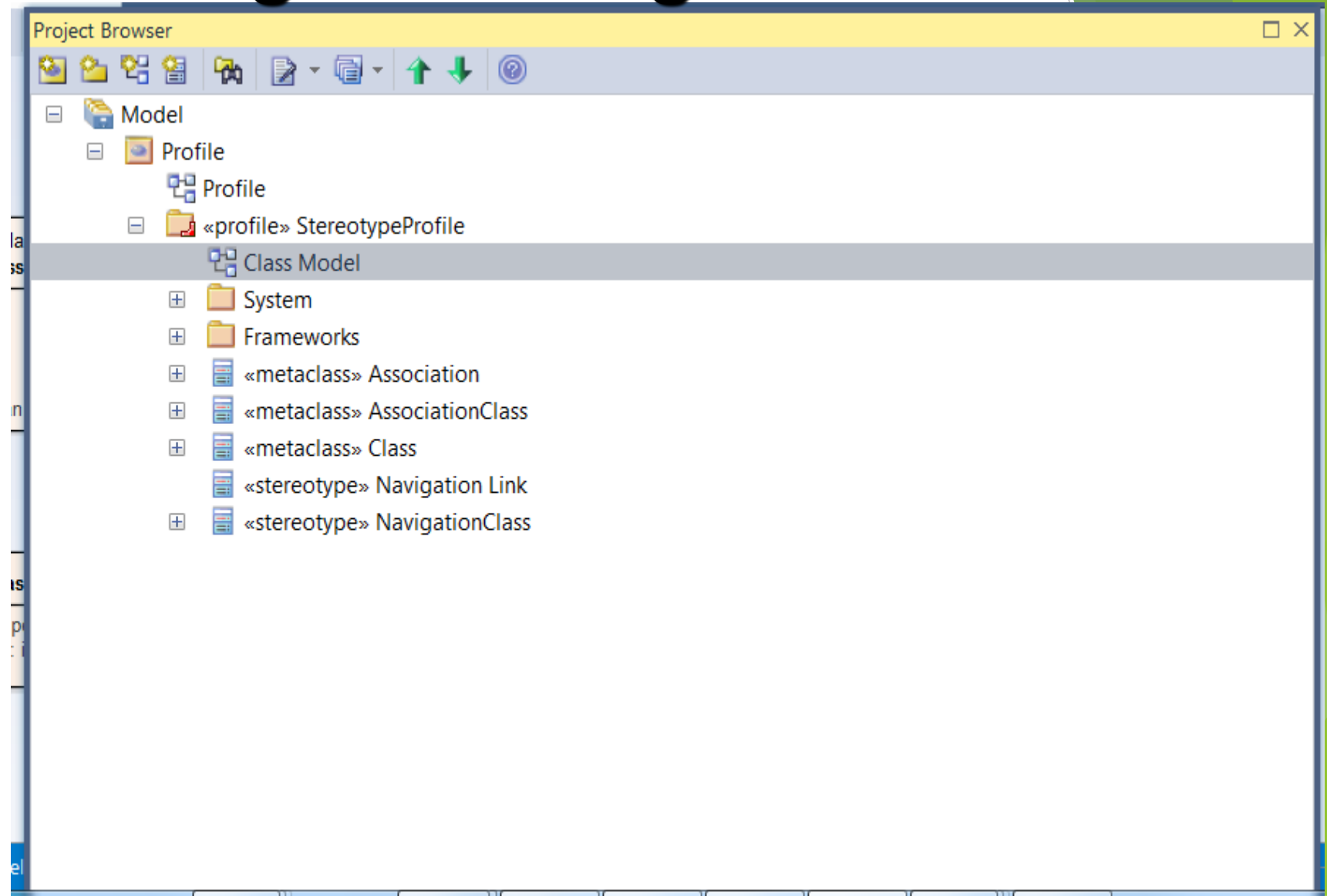


Navigation Design: UML Profile

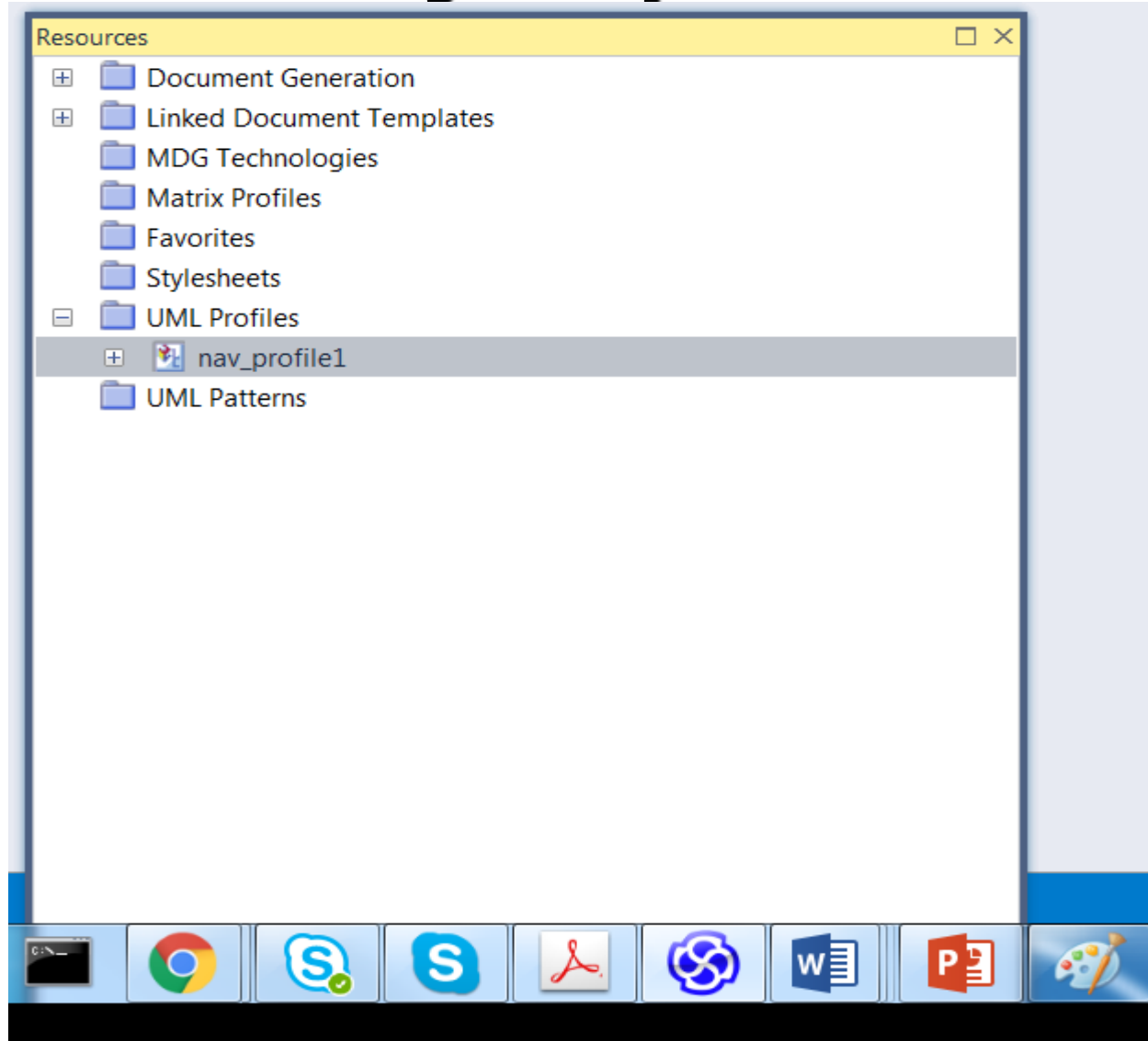


Enumerations is used to restrict the values available to stereotype tags (the attributes of stereotype).

Navigation Design: UML Profile

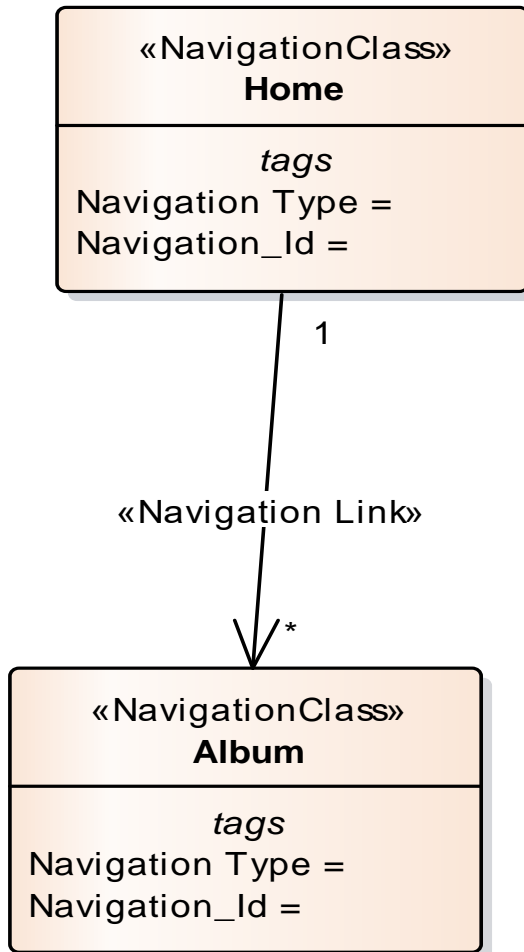


Use your profile in EA



Use your profile in EA

class Class Model



Toolbox

More tools...

nav_profile1

- NavigationClass
- Navigation Link

Interaction

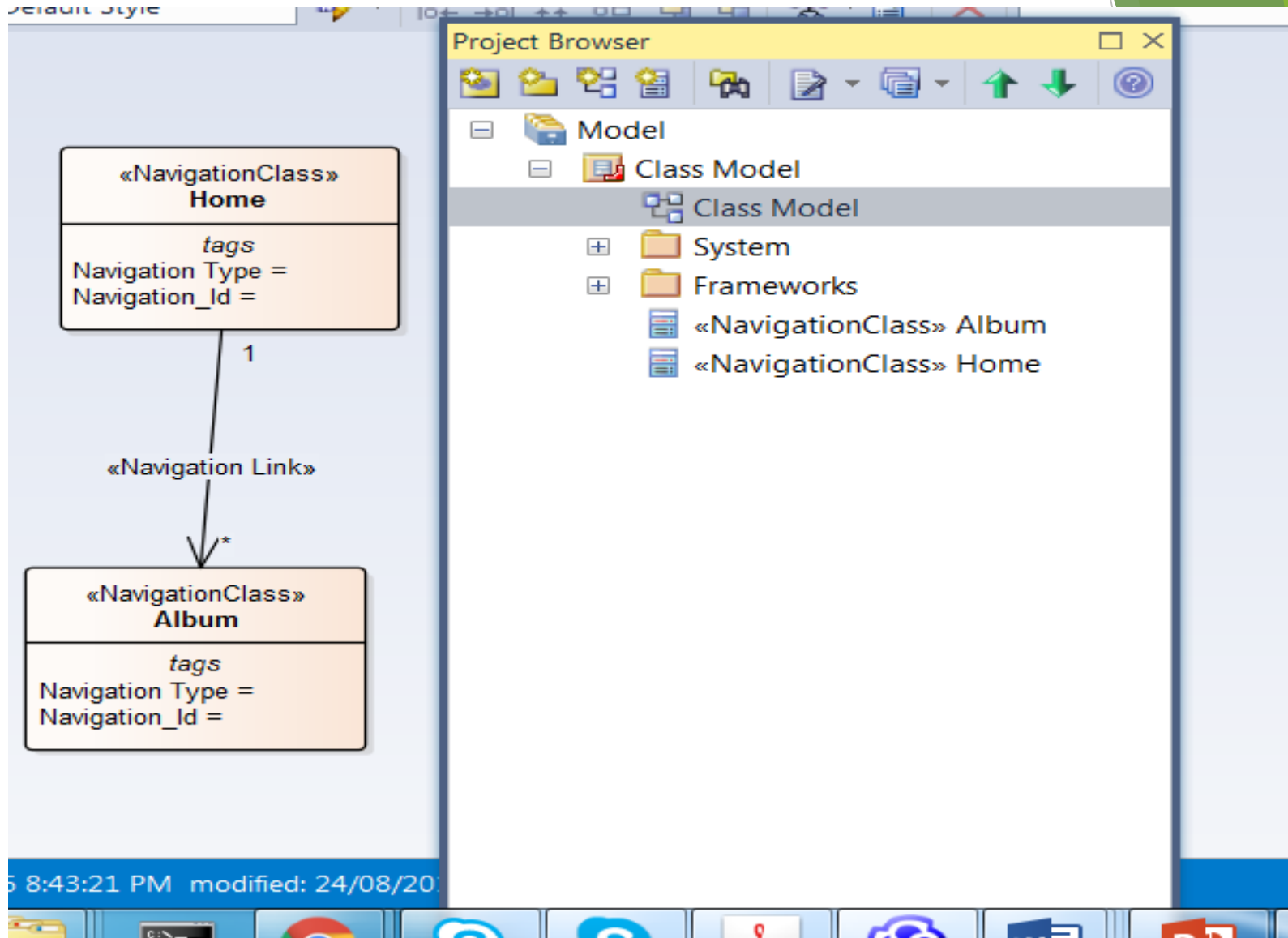
- Actor
- Lifeline
- Boundary
- Control
- Entity
- Fragment
- Endpoint
- Diagram Gate
- State/Continuation

Additional

Timing

- State Lifeline
- Value Lifeline
- Message Label
- Message Endpoint

Use your profile in EA

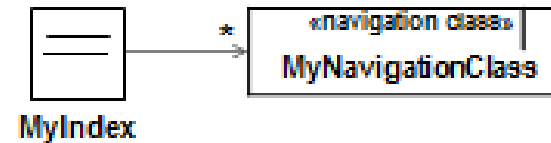
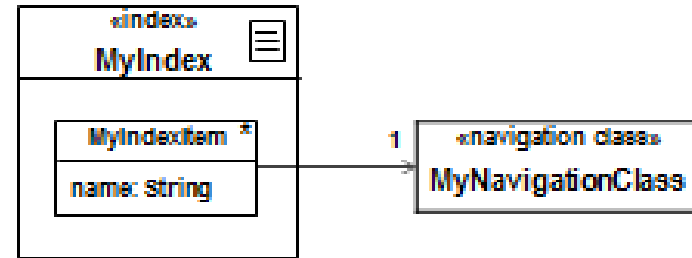


Navigation Design

Navigation Modelling Elements

Modelling element: index

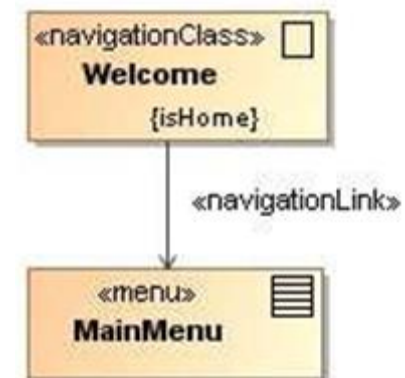
- specifies a class of composite objects
- contains an arbitrary number of index items
- each index item is an object which has a name and owns a link to an instance of a navigation class



- Menus are used to structure the outgoing links from a node

- usually associated to a navigation class by composition
- consists of a set of links to heterogeneous elements, such as indexes, queries, instances of navigation classes or other menus
- UML stereotype: **«menu»**

«query» for selection of instances of a navigation class



Navigation Design (First step in EA)

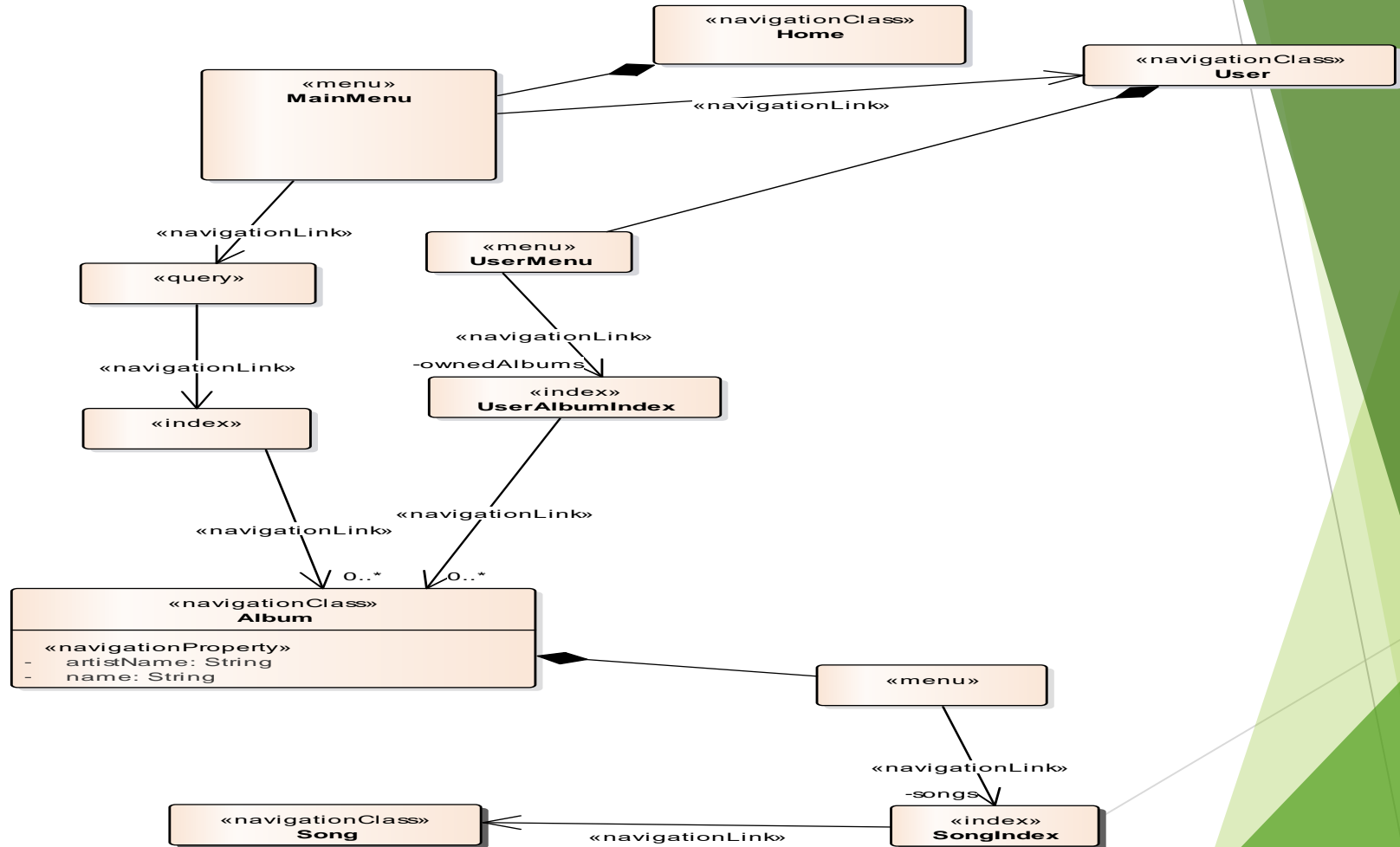
- Develop initial navigational model from your analysis
 1. Create a navigation class for each navigation relevant content class using your UML profile (This comes from information structure)
 - a. Navigation classes map to content classes
 1. Attributes of a navigation class maps to the attributes of content class)
 2. Map navigation links of navigation model to associations of content model using your information structure
 3. transform aggregations and compositions into navigation links as required
 4. transfer multiplicities, aggregations and compositions of content model to corresponding modelling elements of the navigation model
 5. add navigation links based on the scenarios of the use case model and information structure

Navigation Design (Second step) in EA

- Add menu, query, index
- Refine the class diagram
 1. add an *index* to all navigation links, which have multiplicity > 1 at the directed association end (one index item for each navigation object)
 2. Add a *query* to select a subset of navigation objects
 3. Add an additional navigation link from (query, index, menu) to target navigation class as required

Navigation design-Second Step

class Navigation-Step2



Navigation Design (Third Step) in EA

❖ Add User observable function

- ❖ Add the processes to your navigation when any user interaction /navigation requires the system to do some tasks,
- ❖ Add the process link between source to destination process and add return process link back to the target navigation class.



More on Sequence Diagram

- ▶ <http://www.ibm.com/developerworks/rational/library/3101.html>
- ▶ http://www.sparxsystems.com/enterprise_architect_user_guide/10/standard_uml_models/sequencediagram.html

Conclusion

- ▶ We looked at **information architectures**, **blueprints**, **wireframes** and **navigation** design models.
- ▶ **Content is crucial!** Without it our website is useless.
- ▶ The heart of a WebApp information design is the **information architecture we choose** to go with.
 - ▶ So how information will be **organised**, **accessed** and **managed**.
- ▶ There is a range of models we can use, however **blueprints** and **wireframes** are the best.
 - ▶ **Blueprint** gives us the baseline for information access
 - ▶ **Wireframe** forms the bridge to how we design the interaction

Readings

Thanks to Sparx Systems to allow us to use EA for free.

Essential Scrum—Kenneth S. Rubin

The agile age-Managing projects effectively using agile scrum---Brian vanderjack

R. S. Pressman and D. Lowe: *Web Engineering, A Practitioner's Approach*, McGraw-Hill, 2009.

- Chapter 10: Information Design

(concentrate on the topics covered in the lecture)

Papers and other reading materials in “Week 6 Readings” folder on CloudDeakin.

