204362 – Object-Oriented Design

Object Interaction – Sequence Diagrams
(Part II)

Adapted for 204362
by Areerat Trongratsameethong

Bennett, McRobb and Farmer, *Object Oriented Systems Analysis and Design Using UML*
4th Edition, McGraw Hill, 2010

In This Lecture You Will Learn:

- how to model object interaction using an interaction communication diagram.
- how to model interactions using interaction overview diagrams;
- how to model interaction using an interaction sequence diagram;
- how to use timing diagrams.
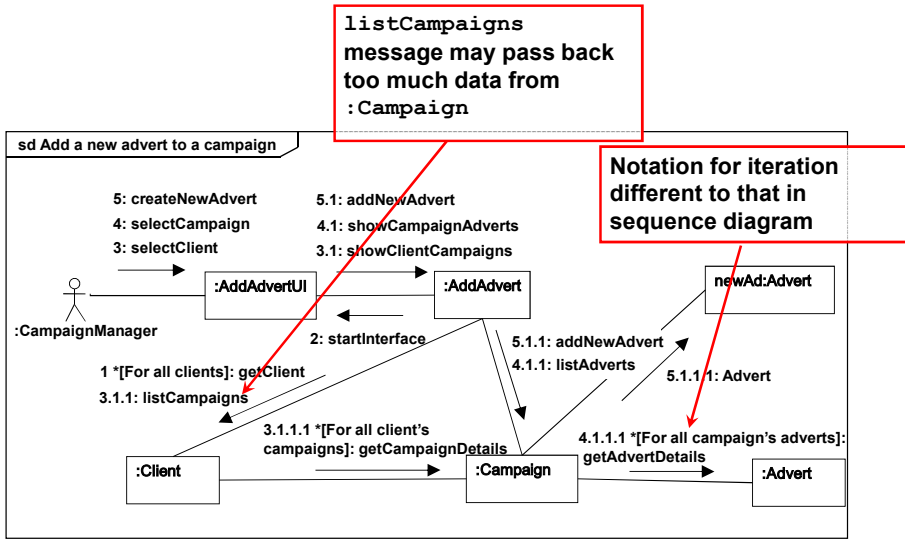

Object Interaction – Communication Diagram

Communication Diagrams

- Hold the same information as sequence diagrams.
- Show links between objects that participate in the collaboration.
- No time dimension, sequence is captured with sequence numbers.
- Sequence numbers are written in a nested style (for example, 3.1 and 3.1.1) to indicate the nesting of control within the interaction that is being modelled.
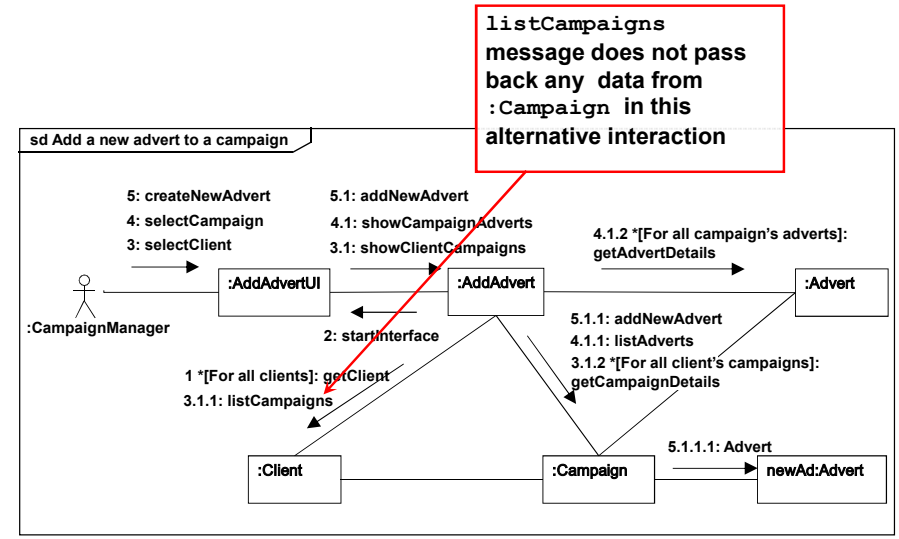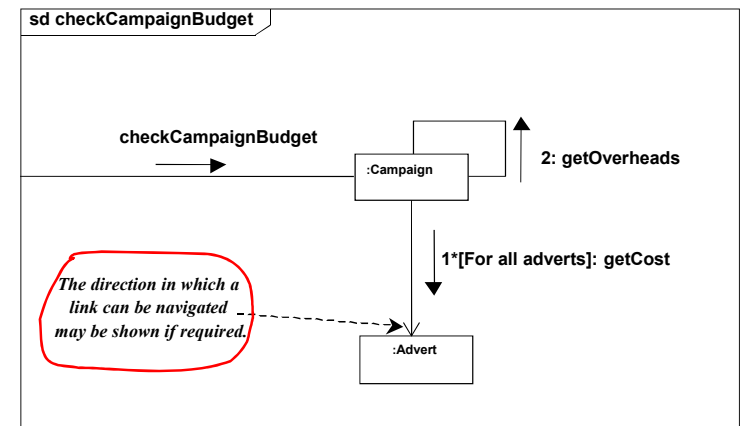
Communication Diagrams

listCampaigns message may pass back too much data from **:Campaign**

Notation for iteration different to that in sequence diagram

sd Add a new advert to a campaign

5: createNewAdvert
4: selectCampaign
3: selectClient

5.1: addNewAdvert
4.1: showCampaignAdverts
3.1: showClientCampaigns

:AddAdvertUI   :AddAdvert   newAd:Advert

:CampaignManager

2: startInterface

5.1.1: addNewAdvert
4.1.1: listAdverts

5.1.1.1: Advert

1 *[For all clients]: getClient
3.1.1: listCampaigns

3.1.1.1 *[For all client's campaigns]: getCampaignDetails

4.1.1.1 *[For all campaign's adverts]: getAdvertDetails

:Client   :Campaign   :Advert

5

Communication Diagrams

listCampaigns message does not pass back any data from **:Campaign** in this alternative interaction

sd Add a new advert to a campaign

5: createNewAdvert
4: selectCampaign
3: selectClient

5.1: addNewAdvert
4.1: showCampaignAdverts
3.1: showClientCampaigns

4.1.2 *[For all campaign's adverts]: getAdvertDetails

:AddAdvertUI   :AddAdvert   :Advert

:CampaignManager

2: startInterface

5.1.1: addNewAdvert
4.1.1: listAdverts
3.1.2 *[For all client's campaigns]: getCampaignDetails

1 *[For all clients]: getClient
3.1.1: listCampaigns

5.1.1.1: Advert

:Client   :Campaign   newAd:Advert

6

Message Labels

Type of message	Syntax example
Simple message.	`4: addNewAdvert`
Nested call with return value. *The return value is placed in the variable* `name`.	`3.1.2: name = getName`
Conditional message. *This message is only sent if the condition* `[balance > 0]` *is true.*	`5 [balance > 0]: debit(amount)`
Iteration	`4.1 *[For all adverts]: getCost`

Navigating Links

sd checkCampaignBudget

checkCampaignBudget

:Campaign

2: getOverheads

1*[For all adverts]: getCost

The direction in which a link can be navigated may be shown if required.
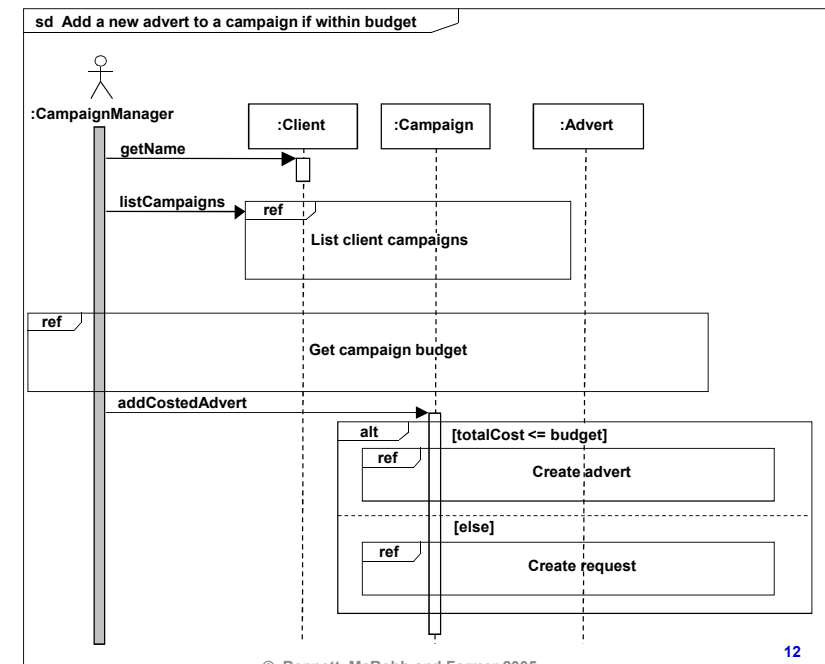
:Advert

8

Slide 9

**Object Interaction –
Interaction Overview Diagrams
Timing Diagrams**
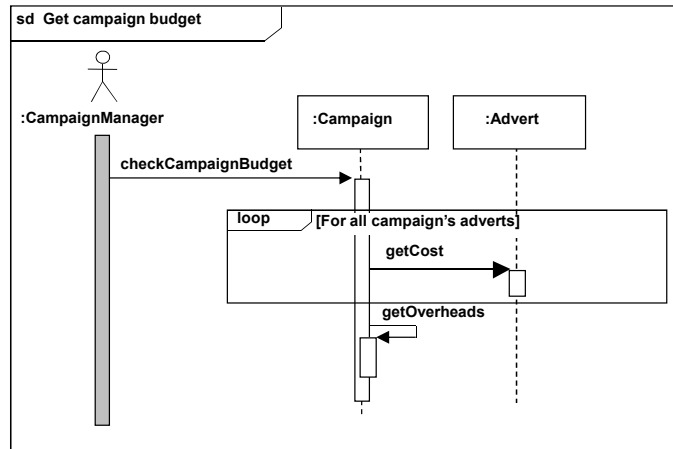
Slide 10

Interaction Overview Diagrams

- Variants of activity diagrams
- Focuses on the flow of control in an interaction
- Nodes in the diagram may be interactions or interaction occurrences
- Interaction needs to be broken down into its key elements.

Slide 11
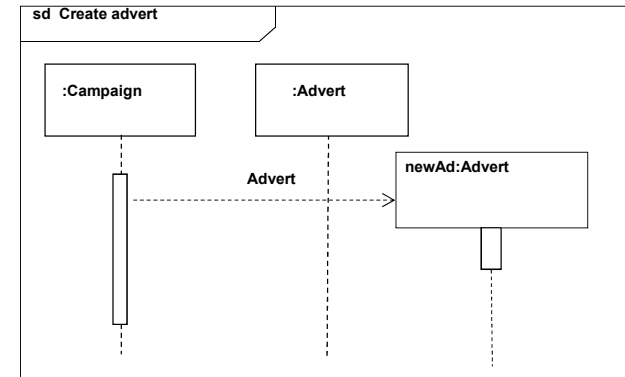
Interaction Overview Diagrams

- An alternative version of the sequence diagram `Add a new advert to a campaign if within budget` is shown on the next slide and is used to develop an interaction overview diagram

Slide 12

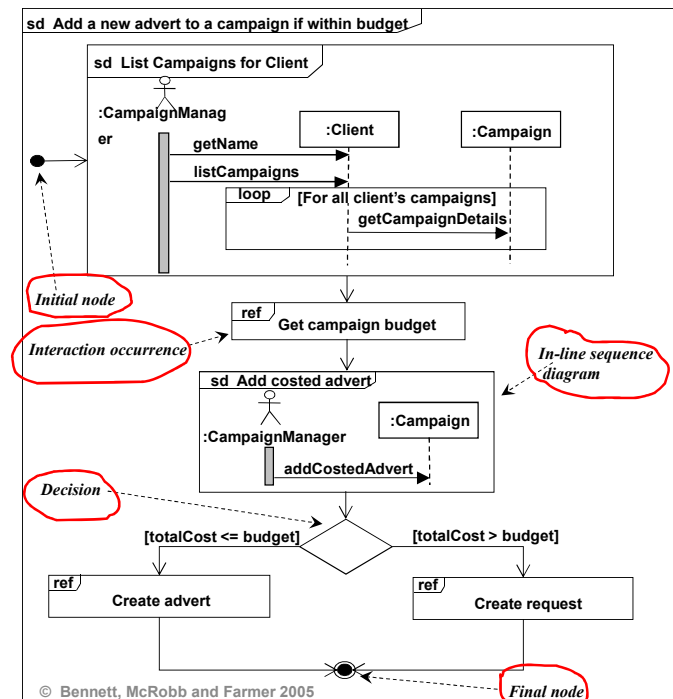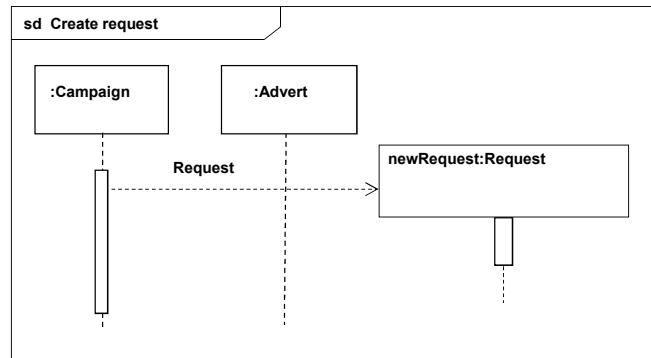sd Add a new advert to a campaign if within budget

:CampaignManager   :Client   :Campaign   :Advert

getName
listCampaigns
ref List client campaigns
ref Get campaign budget
addCostedAdvert
alt [totalCost <= budget]
ref Create advert
[else]
ref Create request

Interaction Fragment Used

sd Get campaign budget

:CampaignManager   :Campaign   :Advert

checkCampaignBudget

loop   [For all campaign's adverts]

getCost

getOverheads

Interaction Fragment Used

sd Create advert

:Campaign   :Advert

Advert   newAd:Advert

Interaction Fragment Used

sd Create request

:Campaign   :Advert

Request   newRequest:Request

sd Add a new advert to a campaign if within budget

sd List Campaigns for Client

:CampaignManager   :Client   :Campaign

getName

listCampaigns

loop   [For all client's campaigns]

getCampaignDetails

Initial node

ref   Get campaign budget

Interaction occurrence

In-line sequence diagram

sd Add costed advert

:CampaignManager   :Campaign

addCostedAdvert

Decision

[totalCost <= budget]   [totalCost > budget]
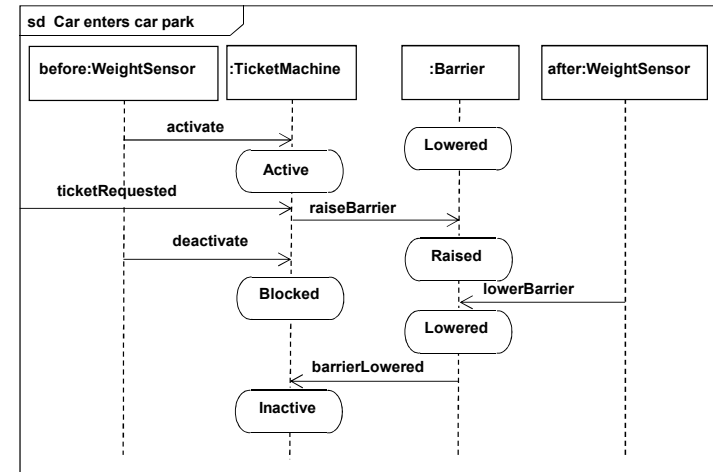
ref   Create advert   **ref**   Create request
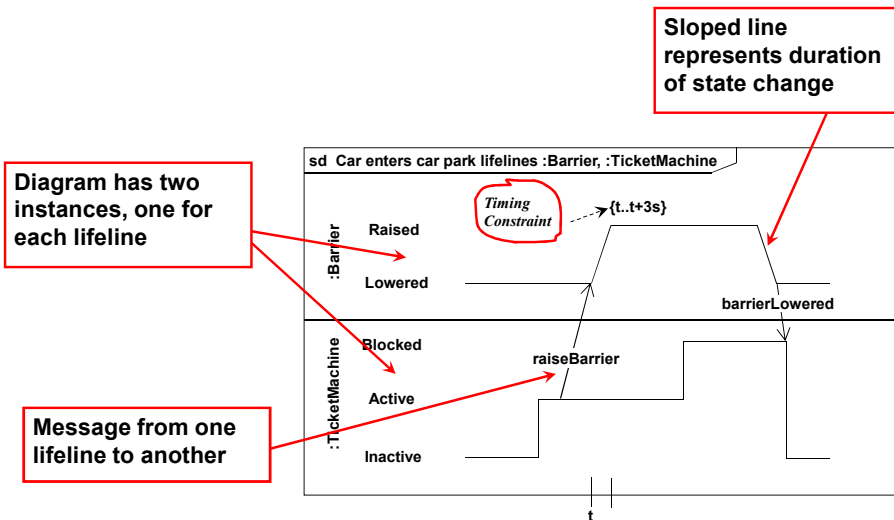
Final node

Timing Diagrams

- A new feature in UML 2.0
- Show how time constraints affect interactions between lifelines
- The sequence diagram `Car enters car park` is the basis for the subsequent timing diagram

Timing Diagrams

sd Car enters car park

before:WeightSensor | :TicketMachine | :Barrier | after:WeightSensor

activate — Active
Lowered
ticketRequested
raiseBarrier
deactivate — Raised
Blocked
lowerBarrier
Lowered
barrierLowered
Inactive

Timing Diagrams

Sloped line represents duration of state change

Diagram has two instances, one for each lifeline

sd Car enters car park lifelines :Barrier, :TicketMachine

Timing Constraint ---> {t..t+3s}

:Barrier
Raised
Lowered
barrierLowered

:TicketMachine
Blocked
raiseBarrier
Active
Inactive

Message from one lifeline to another

t

Model Consistency

- Timing diagrams must be consistent with the relevant sequence diagrams and state machines.

Summary

In this lecture you have learned about:

- how to model object interaction using an interaction communication diagram.
- how to model interactions using interaction overview diagrams;
- how to model interaction using an interaction sequence diagram;
- how to use timing diagrams.

References

- UML Reference Manual (OMG, 2009)
- Bennett, Skelton and Lunn (2005)
- UML Reference Manual (OMG, 2009)
- Bennett, Skelton and Lunn (2005)

(For full bibliographic details, see Bennett, McRobb and Farmer)