ADVANCED SOFTWARE DESIGN
LECTURE 5
MODELLING BEHAVIOUR
Dave Clarke
GOALS

• To provide UML notations for expressing system behaviour.

• To provide guidelines for using these notations.

• Key word: Interaction
CONTENTS

• What is behaviour?

• Relevant UML diagrams and their relationship

• Guidelines
WHAT IS BEHAVIOUR?

BEHAVIOUR IS INTERACTION BETWEEN OBJECTS.
INTERACTION DIAGRAMS
INTERACTION DIAGRAMS

• Visualise the interactive behaviour of the system.

• Describe the message flow in the system.

• Describe structural organisation of the objects.

• Describe interaction among objects.

• Different types of models capture different aspects of the interaction.
KINDS OF DIAGRAMS

• Use Case Diagrams (already considered)
• Sequence Diagrams
• Communication Diagrams
• State Machine Diagrams
• Activity Diagrams
• Interaction Overview Diagrams
• Timing Diagrams (not considered)
SEQUENCE DIAGRAM

• defines sequence of events, in particular, order in which messages occur

• a natural refinement of Use Case diagrams
KEY INGREDIENTS

found message (sender not specified)

execution specification

lifeline

message

named object

:A

doOne

dotwo()
FRAMES: OPTIONS

essentially an if statement
FRAMES: ALTERNATIVES

if-then-else

alt \[ x < 10 \] calculate
[ else ] calculate

\[ A \quad B \quad C \]
FRAMES: LOOPS

frames can be nested

also par (parallel) and region (critical sections)
SOME OTHER FEATURES

self call

selfCall

clock

:ClockStarter

startClock

clock

:Clock

creation

active object

run

asynchronous
EXAMPLE:
BUYING A COFFEE AT STARBUCKS
COMMUNICATION DIAGRAM

• illustrate object interactions in a graph or network format
• objects are placed anywhere in the diagram
• captures essence of wall sketch
EXAMPLE

makePayment(cashTendered) → Register

1:makePayment(cashTendered) → Sale

1.1:create(cashTendered)

:Payment

direction of flow

sequence number

nesting – subcall of 1

conditionals also possible
LARGER EXAMPLE
STARBUCKS (AGAIN)
COMPARISON
SEQUENCE VS COMMUNICATION DIAGRAMS

• Sequence diagrams:
  • easier to see flow, just follow arrows (vs numbers)
  • large number of notation options
  • rigid structure – consumes horizontal space

• Communication diagrams:
  • easier to sketch free form, more space efficient, and easier to modify on the fly
  • difficult to see sequence of messages
  • fewer notation options
STATE MACHINE DIAGRAM

• illustrates the behaviour of an object in reaction to an event
• based on interesting events and states of an object
• show the lifecycle of an object
• use only for state-dependent objects
• limited use in business information systems
• common in process control, device control, protocol handlers, and telecommunications
GUARDS AND ACTIONS

- Active: off hook / play dial tone
- Idle: on hook

Action (fires when transition is taken)
Guard (inhibits transition)

Valid subscriber
NESTED STATES

![State diagram showing the transitions between states: Idle, Off hook / Play dial tone, Dialing, Active, Talking, Connecting, and Digit complete.]

- **Idle**: The state where the system is not in use.
- **Off hook / Play dial tone**: Transition from Idle to Play Dial Tone.
- **Valid subscriber**: Transition from Idle to Dialing.
- **Dialing**: Transition from Idle to Dialing with a digit.
- **Connecting**: Transition from Dialing to Connecting with a complete digit.
- **Active**: Transition from Connecting to Talking.
- **Talking**: Transition from Connecting to Talking with a connected digit.

Events:
- **Digit**: Used in transitions Dialing -> Connecting and Connecting -> Talking.
- **Complete**: Used in transition Connecting -> Talking.
**EXAMPLE**

- **WaitingForSale**
  - makeNewSale to **EnteringItems**
  - makeCashPayment to **Authorising Payment**
  - authorised to **WaitingForPayment**

- **EnteringItems**
  - enterItem to **WaitingForSale**
  - endSale to **WaitingForPayment**

- **WaitingForPayment**
  - makeCreditPayment to **Authorising Payment**
  - makeChequePayment to **WaitingForSale**
STARBUCKS EXAMPLE
REVISITED
WHERE IS THE STATE?
ACTIVITY DIAGRAM

• rich notation to show a sequence of activities, including parallel activities.

• popular for business workflows and processes, and use cases.

• partitioned into different actors

• expresses control flow, but not actual message contents.

• can incorporate external systems (e.g., databases)
EXAMPLE

partitions. different parties involved

object node. object produced or used by actions

Sunday 3 November 13
ANOTHER EXAMPLE

Customer

Shop and Fill Cart

Cart

Cashier

Enter Cart Items

NextGen POS

Calculate Taxes and Discounts

[ cash payment ]

[ else ]

Submit Authorisation Request

Authorisation Service

Authorise Payment

Create Receipt

Receipt

Hand Over Items

Submit
STARBUCKS?
• Provide a big-picture overview of how a set of interaction diagrams are related in terms of logic and process flow.
FRAMES

Encloses some kind of interaction diagram.
Name describe relevant component.
An interaction defined elsewhere can be referred to inside an Interaction Overview diagram.

```
sd Add Item to Shopping Cart

:User

:Shopping Cart
```

```
ref Add Item to Shopping Cart
```
EXAMPLE 2
GUIDELINES
GUIDELINES

• Novice modellers do not pay enough attention to interaction diagrams

• Do dynamic object modelling with interaction diagrams, not just static modelling with class diagrams.

• Without modelling behaviour, you cannot really understand the system.

• Interaction Diagrams can express use cases in concrete form that relates to classes
GUIDELINES

- Walkthrough dynamic models against use cases.
- Adjust static models whenever dynamic models expose weaknesses.
- Ensure reuse among different dynamic models touching the same class – e.g., reusing a method for different scenarios.