ADVANCED SOFTWARE DESIGN
LECTURE 2
DOMAIN MODELLING
Dave Clarke
WHAT IS GOOD DESIGN?
GOOD DESIGN: DESIGN THAT COPES WITH CHANGE.
EMBRACE CHANGE
HOW CAN WE TELL WHETHER A DESIGN IS GOOD?
READING: DOMAIN MODELLING
MODELLED AND UML
UML IS ABOUT COMMUNICATION NOT DOCUMENTATION
**DESIGN NOTATION**

Diagram showing relationships:

- **LibraryMember**
  - memberId : Integer
  - Borrow/Return: 0..1

- **Copy**
  - Title: String
  - Borrow/Return: 0..1

- **Book**
  - Title: String
  - Is a copy of: 1..*

- **Journal**
  - Title: String
  - Series: Integer
  - Volume: Integer

- **MemberOfStaff**
  - Borrow/Return: 0..1
DOMAIN MODELLING: GOAL

To gain a better understanding of the domain within which we are working

To capture the key *domain abstractions* (classes)

To capture real-world relationships (associations) between classes
GENERAL GUIDELINES

Model just domain artefacts, not software artefacts

Getting it perfect isn’t a requirement – not cost effective

When classes and their behaviour are modelled in more detail, resulting class model will deviate from domain model.
IN-CLASS DESIGN EXERCISE
Wordfeud

Your Turn

Playing with simon downie
English game (Intl) started 18 Oct 22:35
simon downie played FEZ for 25 points

Playing with Alex
English game (Intl) started 19 Oct 07:03
Alex played EH for 28 points

Playing with barkeman
English game (Intl) started 18 Oct 22:39
barkeman played AX for 14 points

Their Turn

Playing with DelaGul
English game (Intl) started 18 Oct 21:35
You played OXY for 14 points

Playing with vacvac
English game (Intl) started 16 Oct 22:35
You played UTE for 32 points

Finished Games

You lost against venom007
English game (Intl) started 18 Oct 22:33
You played DANT for 11 points

You won against GrannyT1
English game (Intl) started 18 Oct 21:24
GrannyT1 resigned

You lost against simon downie
English game (Intl) started 15 Oct 12:58

You played JUS for 23 points

Playing with simon downie
English game (int) started 18 Oct 22:35
simon downie played FEZ for 25 points

Playing with Alex
English game (int) started 19 Oct 07:03
Alex played EH for 28 points

Playing with barkeman
English game (int) started 19 Oct 22:39
barkeman played AX for 14 points

Playing with DelaGul
English game (int) started 18 Oct 21:35
You played OXY for 14 points

Playing with vacvac
English game (int) started 16 Oct 22:35
You played UTE for 32 points

Their Turn

E N L A T E R
Statistics are grouped by board type and dictionary language.

**Overall statistics**
Any board, any dictionary

**Standard board**
- English (International)

**Random board**
- Dutch
- English (International)
- English (US)
- Swedish
Overall statistics
Any board, any dictionary

Games won 596
Games lost 211
Games tied 2
Games resigned 1
Games timed out 0

Highest game score 930
Highest move score 648

Longest word ENCOURAGES
Average Word Score 13
Average Move Score 23
Average Game Score 364
Most bingos in one game 2
Total number of bingos 43

A bingo is a move using all seven tiles.

Statistics are calculated for games started February 16, 2013 or later.
English (International)
Random board

Games won 586
Games lost 200
Games tied 2
Games resigned 0
Games timed out 0

Skill rating 1824

Highest game score 930
Highest move score 648
Highest scoring word VENDIS (648 points)
Longest word ENCOURAGES
Average Word Score 13
Average Move Score 23
Average Game Score 366
Most bingos in one game 2
Total number of bingos 41
### Player Statistics

<table>
<thead>
<tr>
<th>Statistic</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Longest word</td>
<td>ENCOURAGES</td>
</tr>
<tr>
<td>Average Word Score</td>
<td>13</td>
</tr>
<tr>
<td>Average Move Score</td>
<td>23</td>
</tr>
<tr>
<td>Average Game Score</td>
<td>366</td>
</tr>
<tr>
<td>Most bingos in one game</td>
<td>2</td>
</tr>
<tr>
<td>Total number of bingos</td>
<td>41</td>
</tr>
</tbody>
</table>

A bingo is a move using all seven tiles.

### Friend Statistics

<table>
<thead>
<tr>
<th>Friend Name</th>
<th>Wins</th>
<th>Draws</th>
<th>Losses</th>
</tr>
</thead>
<tbody>
<tr>
<td>astunner</td>
<td>6</td>
<td>0</td>
<td>3</td>
</tr>
<tr>
<td>authentic journey</td>
<td>2</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>barkeman</td>
<td>16</td>
<td>0</td>
<td>14</td>
</tr>
<tr>
<td>Batman Olsson</td>
<td>0</td>
<td>0</td>
<td>1</td>
</tr>
<tr>
<td>Bellechai</td>
<td>2</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>Cyan13</td>
<td>11</td>
<td>0</td>
<td>5</td>
</tr>
<tr>
<td>Drunk-on-life</td>
<td>6</td>
<td>0</td>
<td>2</td>
</tr>
<tr>
<td>elz14c</td>
<td>18</td>
<td>0</td>
<td>9</td>
</tr>
</tbody>
</table>

Statistics are calculated for games started February 16 2013 or later.
DESIGN

How would you develop this application?
ACTIVITY 1:
ELICIT CORE DOMAIN
ELEMENTS
IDENTIFYING DOMAIN ELEMENTS

Select key domain abstractions.

First underline nouns and noun phrases – gives candidate classes

tangible or ‘real world’ things – book, copy, course

roles – library member, student, director of studies

events – arrival, leaving, request

interactions – meetings, intersection
DISCARDING CANDIDATES

**redundant** – encompassed by some other compass, perhaps differing by attributes: *loan* vs *short term loan*. Select most general.

**vague** – e.g., *item*

**an event or operation** – something done by system. If it has no state, discard it.

**meta-language** – language used to define things. *requirements, system, ...*

**outside the scope of the system** – e.g., *library, week, ...*

**an attribute** – something simple, an attribute of another class. *name of library member*
ACTIVITY 2:
ELICIT CORE ASSOCIATIONS
ASSOCIATIONS: RELATIONSHIPS BETWEEN DOMAIN ELEMENTS

Clarify our understanding of the domain by describing objects in terms of how they work together.

Sanity check the coupling in our system – too make sure we are following good principles in modularising our design (later).
FINDING ASSOCIATIONS

Classes A and B are associated if:

- an object of class A sends a message to an object of class B
- an object of class A creates an object of class B
- an object of class A has an attribute whose values are objects of class B or collections of objects of class B
- an object of class A receives a message with an object of class B as an argument

In short: if class A has to know something about some object of class B
ACTIVITY 3: REFINE DOMAIN MODEL, MULTIPLEITIES
MULTIPLICITIES

Express more refined relationships between classes

• an exact number
• a range of numbers
• an arbitrary, unspecified number

Give guidance to implementation.
ACTIVITY 4: REFINE DOMAIN MODEL
ATTRIBUTES
ATTRIBUTES

State of objects (excluding associations)

Generally only of primitive type: ints, strings, enums, ... or library classes

Associations are not modelled as attributes – premature decision about navigability