Visual Modelling with UML

Unified Modelling Language

Aims

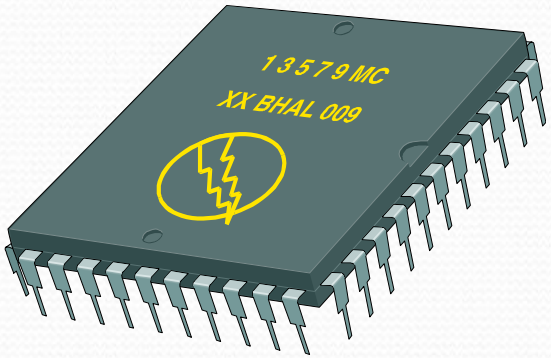
- Introduction to visual modelling using UML
- Modelling Requirements with Use Cases
- Capturing System Structure and Design with Class diagrams
- Capturing System Behaviour with Interaction diagrams

What is “Modelling”?

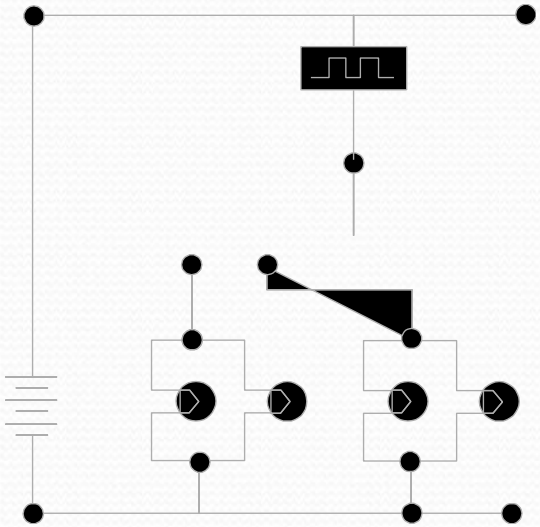
- Modelling
  - an abstraction to show the construction of something
  - a simplified representation of a phenomenon
  - an exhibition of relationships or connections between objects and actions
- Abstraction is performed
  - Lose some details so that an overview is produced

More than one model

- Who is viewing the model.
- Why they need to view it.



View for Customers

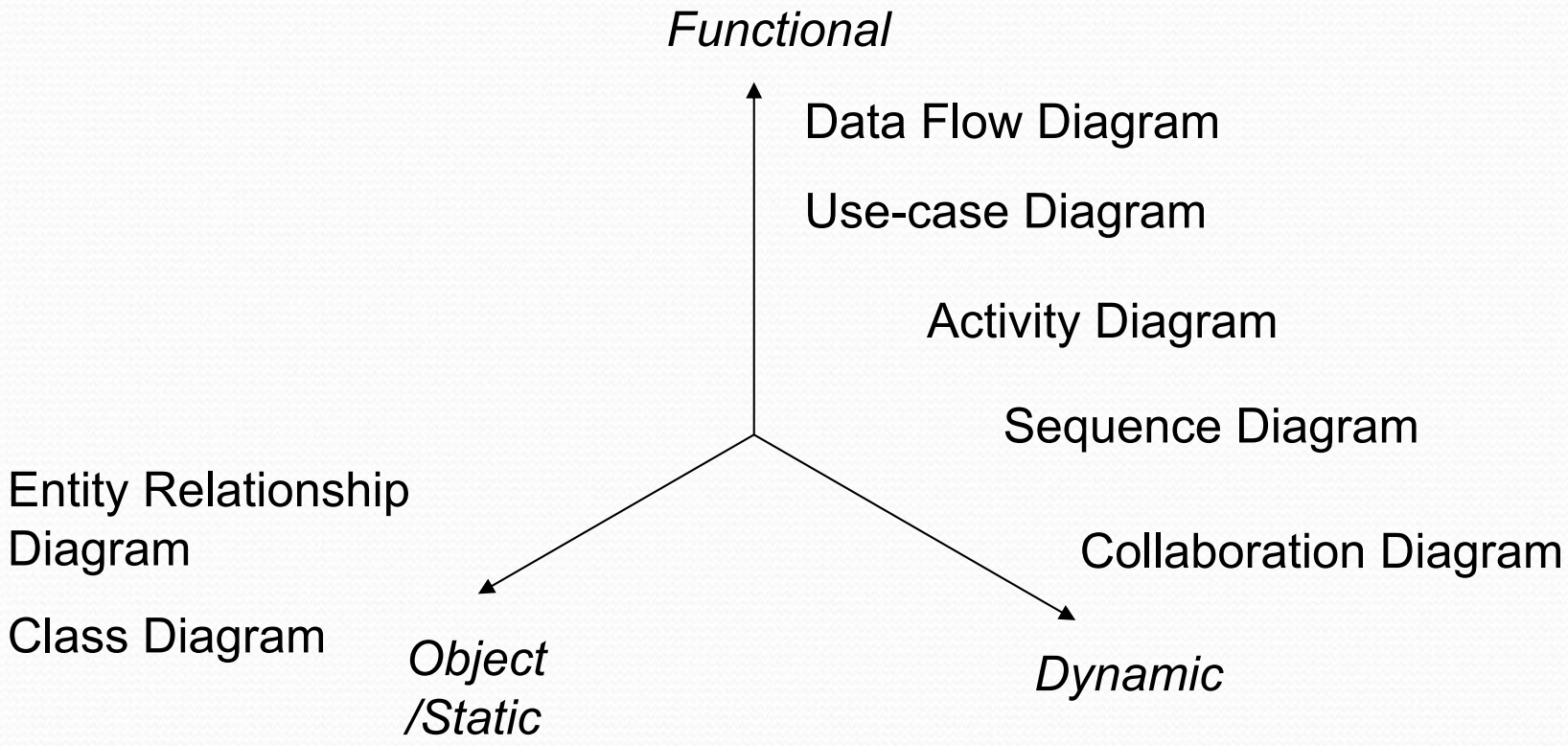


View for Designers

Why Model?

- Modeling achieves four aims:
  - Helps you to visualize a system as you want it to be.
  - Permits you to specify the structure and behavior of a system.
  - Gives you a template that guides you in constructing a system.
  - Documents the decisions you have made.
- You build models of complex systems because you cannot comprehend such a system in its entirety.
- You build models to better understand the system you are developing.

Views of Modeling



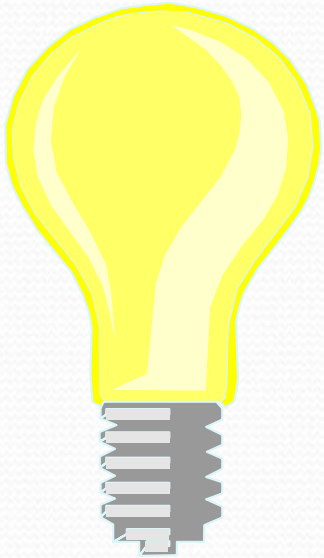
Three Viewpoints we can use

- A software system can be modeled from three viewpoints
  - **Object model** -- describes the static, structural, data aspects of the system
  - **Dynamic model** -- describes the temporal, behavioural, control aspects of the system.
  - **Functional model** -- describes the transformational, algorithmic aspects of the system.

- Note that the three models are not completely independent of each other

The UML Is a Language for Visualising

- Communicating conceptual models to others is prone to error unless everyone involved speaks the same language.
- There are things about a software system you can't understand unless you build models.
- An explicit model facilitates communication.

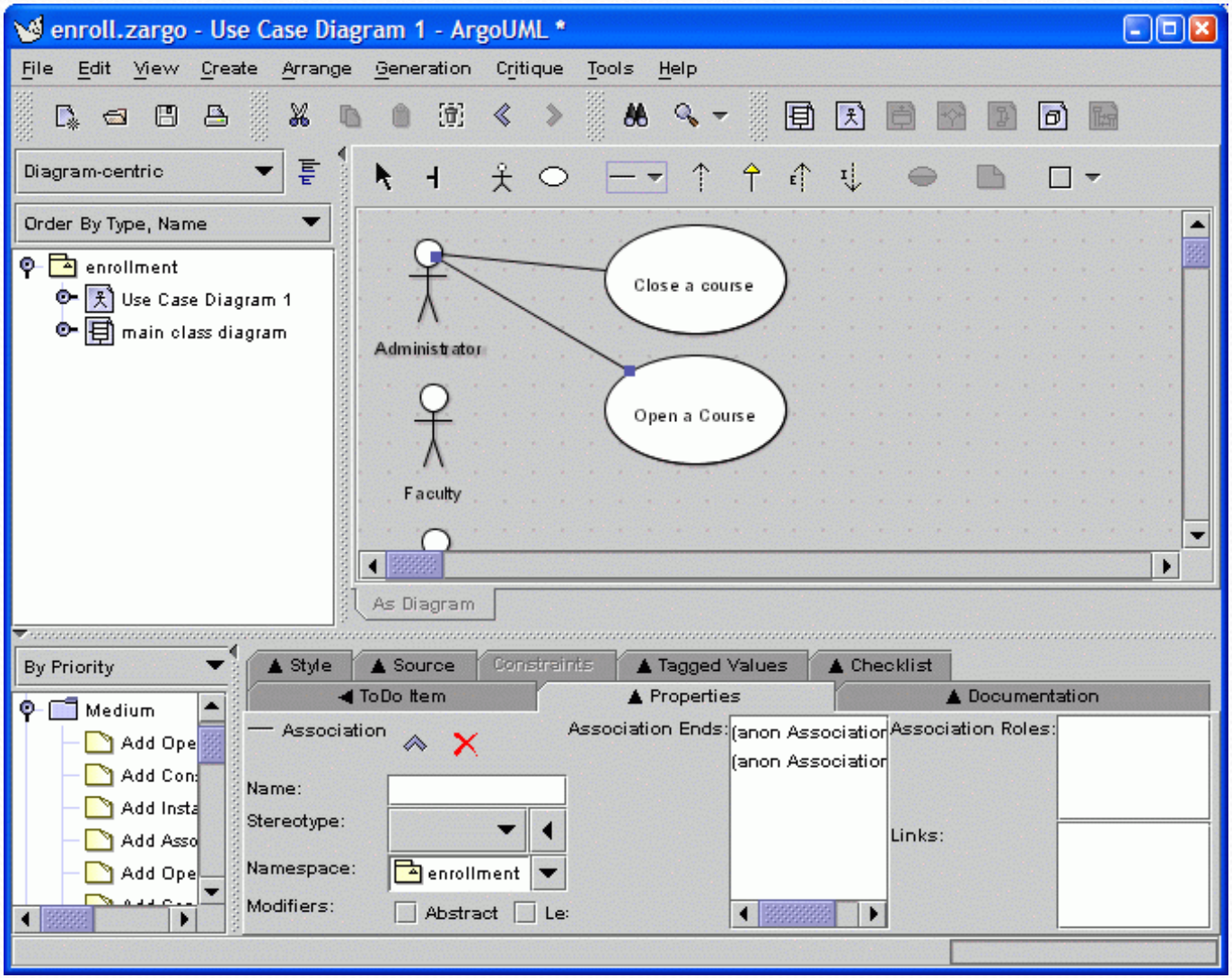


Today, UML has become the standard method for modelling software systems

Tools for drawing UML diagrams

- Several tools and templates for tools exist.
- ArgoUML
- Visio
- Visual Paradigm

Note that we need to use UML 2 notations



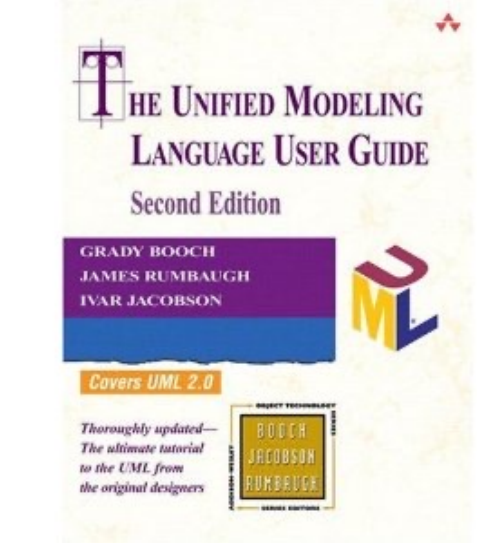
Remarks …

- There is no single correct model of a situation. A good model captures the crucial aspects of a problem and omits the others.
- The key objective is to achieve abstraction --- the selective examination of certain aspects of a problem.
- In this module, we will be looking at some basic techniques and apply them to small systems.

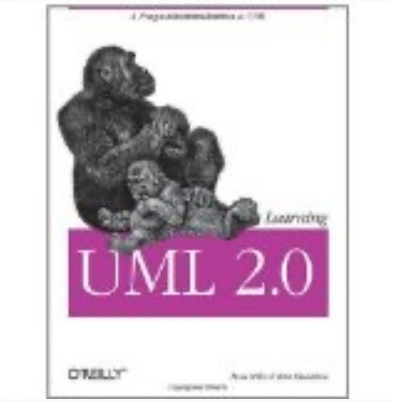
References

Many books in the library:

- The Unified Modelling Language User Guide, Addison, 2005
- Learning UML 2.0, O'Reilly, 2006 – electronic resource



Check the UML resource pages on the Object Management group site: www.omg.org



Note that we shall be using the

UML 2.0 notation