Bournemouth University

Business Processes and Requirements

Professor Keith Phalp

http://dec.bournemouth.ac.uk/staff/kphalp/

The course rationale

- Customers want systems to support their business processes.
  - (We can argue about the b word).
- Developers build systems for clients
- "Oh dear. The system doesn't seem to meet the client's needs". (It must be someone's fault – next slide).
- This is a *requirements* problem, and ***very*** common.
- One reason is that the developers didn't understand the problem: or what they wanted or needed.
  - That's where the business process modelling comes in, and some other assorted ideas.
- Another problem is that they didn't really consider requirements (so we will review what requirements *should* be).
- Finally we consider how best to get the process knowledge to inform the specification, that is how to map from process model to specification.
  - This last bit is something where students have struggled in assignment (usually part 3 of the coursework), BUT where there have been good student PROJECTS.

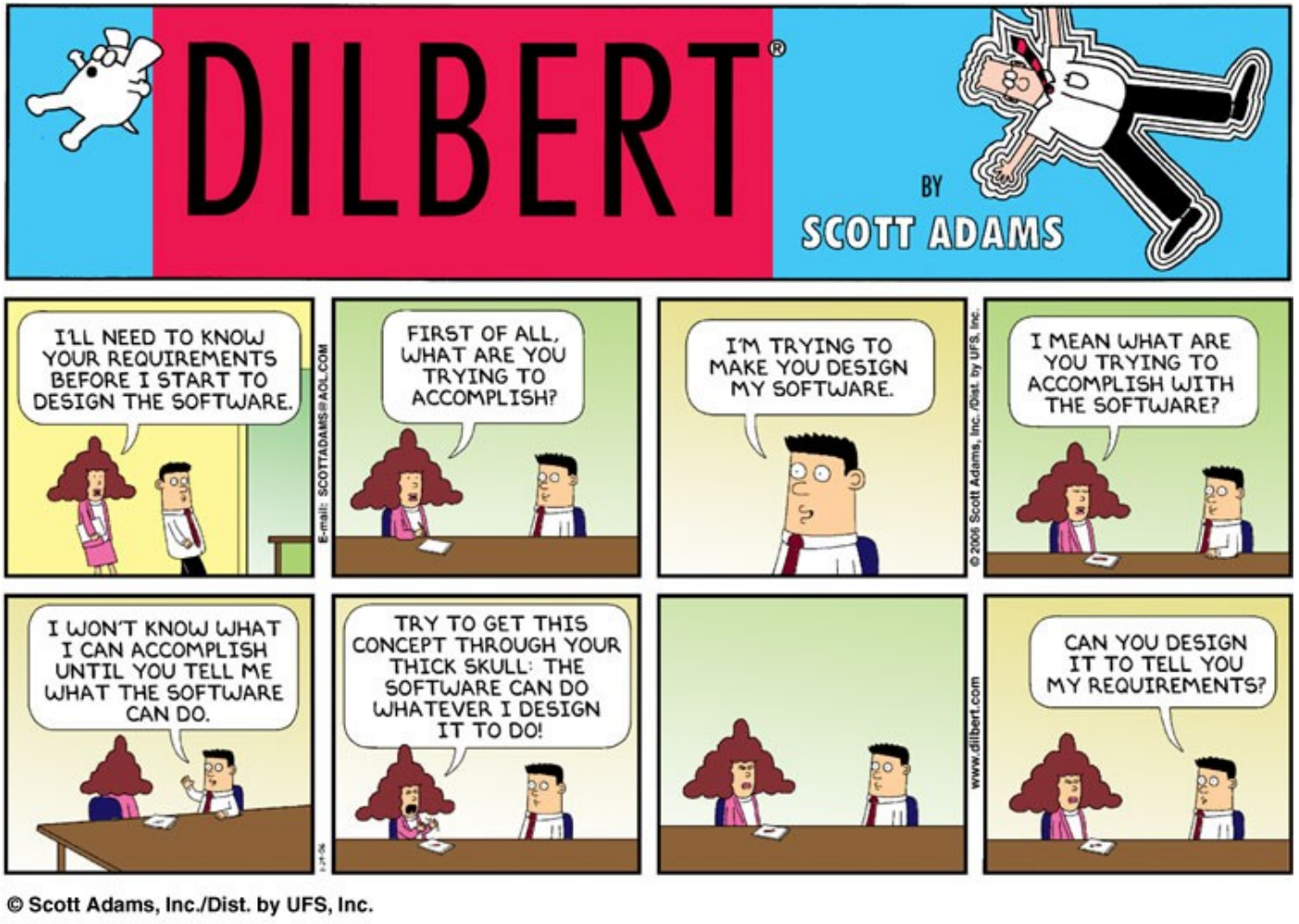
Topics & Assessment (Autumn 2012)

My aim is that topics will relate to the parts of the assignment that you will deliver. Specific outcomes.

- *Should be able to do an 'analysis' of a given scenario – noting issues and ambiguities.*
- *Produce 'process model' of a scenario.*
- *Suggest (and model) improved process.*
- *Be aware of some of the research issues (this is specifically focussed on moving from process model to specification, and the ideas of business and IT Alignment).*
- Some of the later lectures are based around such research.
- Start with some awareness of the requirements field (and some contentious perspectives).

- Attendance and success are also aligned, even down to topics!

Who is at fault?



Bournemouth University

Some themes today

- We need to unpick this problem. So we will look at:
- What did we mean by requirements.
- Does everyone agree? (Of course agreement is unlikely, even among us).
- How do current approaches fare (and past and proposed and future).
- What might we do to help:
  - What is found to be useful in practice
  - Of course it's process modelling.
  - How do we put our ideas together.

- Notes and further details: http://dec.bournemouth.ac.uk/ESERG/kphalp/teaching.htm

Bournemouth University

I know I said this but.... we will look at...

- Thinking about requirements and what requirements should and should not be (week one).
- Business process modelling (and why we want to do some).
- Process modelling notations (and role models)
- In particular: Role activity Diagrams (RADs)
- Some thoughts about specification (and further problems with use cases).
- Mapping from process models to use cases (some alternative ways to do this, and a pragmatic approach).
- Experiences of business analysis and requirements.
- Will review ordering as we go – to support assignment, preferences etc.
- Will give glimpses of all of these this week, and then cover in greater detail later (so you have an idea as to what is coming).

Other things to consider..

- What do I know?
- Impact: is this stuff used?
- Other things up front...
- I want to know your experiences (e.g., what methods you have used, and your thoughts on those methods).
- It's not about getting a single 'right' answer, it's about ensuring that all issues are covered, and that you match, are consistent across models and descriptions.
- This should be a dialogue / discussion, and the quality of discussion and reflection, rather than an absolute answer.

Methods – an aside

- How many used a definite requirements method on placement?
- How many used a definite design method?
- Was the method standard (published) or company specific.
- Was it well known, e.g., Rational Unified Process?
- List of methods used for
    - a) requirements or specification
    - b) design
    - c) general software development process
    - d) other...

    - What does this tell us about methods claims?

Bournemouth University

What do I know

- My experiences.
- Software Process modelling (91...)
- Business Process Modelling (94...)
- Consultancy, UK & Europe (plus short courses)
- Methods used more widely
- Process Oriented Requirements Engineering (taught here since 97 in various forms..) - many gone on to use in practice.
- Use case guidelines (2000 onwards...)
- Strategy and Process Oriented Requirements Methods, widely published from around 2005 onwards (BSCP 2006).
- Much of this part of the units covers methods that I have used directly (such as RADs), invented or co-authored (BSCP, mappings to use cases), and which has been used across many industries.
- Recent supervision of PhD which embedded requirement with MDA.

Impact

- **Process approaches (BSCP)**
- Enterprise Analysts Pty Ltd
- Promise Point: boutique management, Japan
- NICTA Ltd Consultants
- NTT Data (largest SI in Japan), use of tool (based on method): estimated to save potentially Y10Ms of rework costs on one project alone
- NRI (2nd largest SI in Japan)
- Commonwealth Bank Australia
- Centrelink (Australian Benefits Agency): IT refresh across whole country delivered real business benefits, saving potentially $Ms in rework
- NICTA (new management system) -> saved over $150K (on initially budgeted $250K project) Prickie.com (saved $100Ks for company directors)

- **Use case guidelines various papers 2000-2012**
- Used in practice and taught to business analysts at Macquarie Bank, Sydney Australia
- Were used by Sharepeople to promote IT Dept expansion when acquired by American Express

Analysis, Requirements, Spec & External Design

Tasks :-	**analysis**	**specification**	**design**
Systems :-	**problem domain**	**interface**	**solution system**
Outputs :-	**analysis/ requirements document**	**specification document**	**design document**

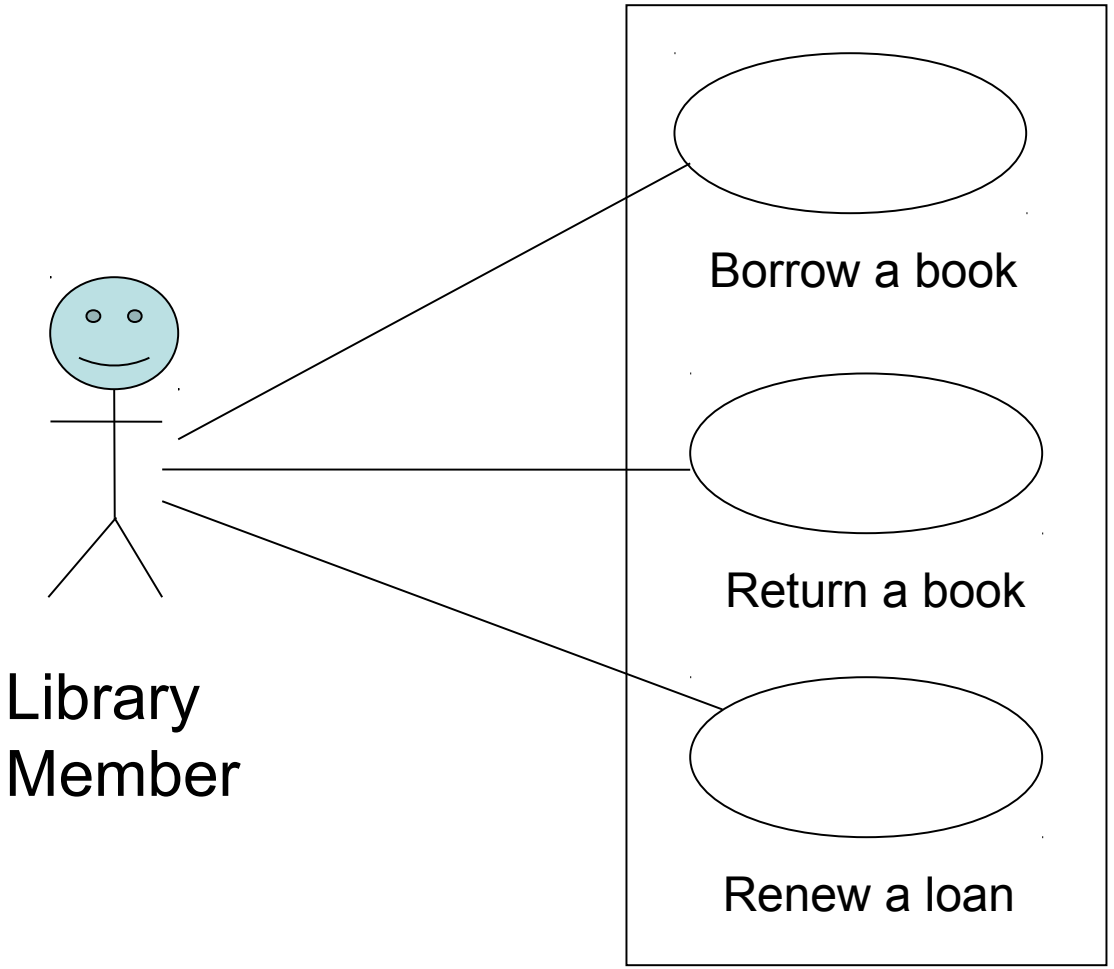
Bournemouth University

What requirements isn’t and implications.

- The main thing is to realise that we need to understand that *requirements is about the problem domain*, and is *not the same as specification* (which we will also define).
- Most people mix these up, but:
  - Most people start from specification
  - There is no such thing as a requirements specification.
  - Use cases are really specification (if you don't believe me – wait a few slides).

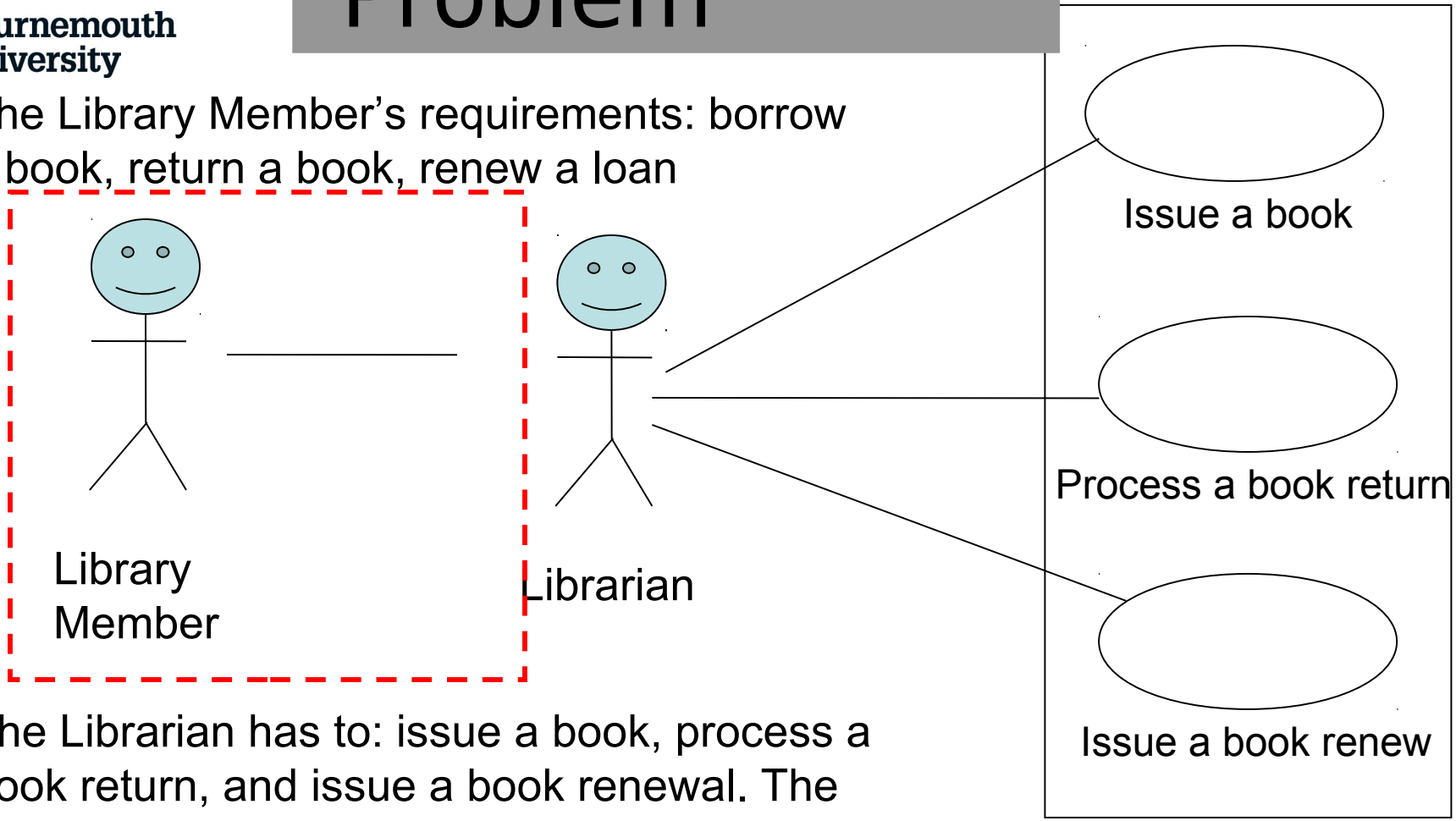
- Let’s start with some things we know well :)
- OR: Oh NO not Use Cases again....
- CAVEAT – have published extensively on use cases – so I may be overstating the arguments against them (you decide).

A Use Case Example



The UC Problem

The Library Member's requirements: borrow a book, return a book, renew a loan



The Librarian has to: issue a book, process a book return, and issue a book renewal. The Library Member's requirements aren't quite the same as the Librarian's tasks.

Bournemouth University

Use Case problems

So what happens when a librarian's job is to loan books out and collect returns?

What happens to the Library Member when we cannot show actors one step removed from the machine?

Do we just forget about them or fudge them?

But isn't a library's purpose to provide books and services to its library members?

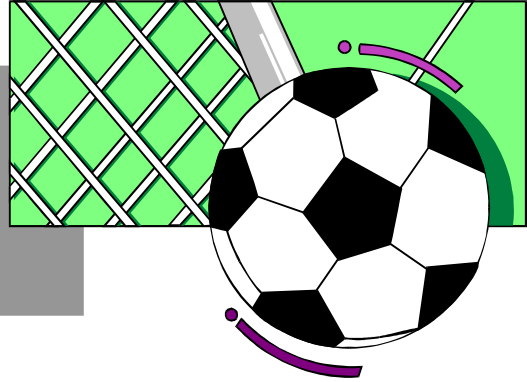
And if we ignore this fact, what are we essentially doing?

We're ignoring the problem context, andwe're ignoring the requirement: **and that's bad**

Dependencies in Descriptions (big problem to come)

Note. We will (of course) adopt use cases anyway!

Two sporting use cases



- The match reached full-time
- The referee blew his/her whistle
- The ball crossed the goal-line
- The goal was not given

Alternatives

4. The goal was given

- The match reached full-time
- The referee blew his/her whistle
- The ball crossed the goal-line
- The goal was given

Alternatives

4. The goal was not given

Validation & Context. Someone who ‘knows the the game’.

Bournemouth University

Historically

- "Traditional" analysis (1950's)
- Structured Analysis (late 1960's)
- Modern" Structured Analysis (late 1980s
- Object Oriented Analysis (1990s
- Problem Domain Oriented Analysis (2000)
- Process Oriented Requirements Engineering: from mid to late 1990s, but latterly issues resurfaced within
  - Requirements Engineering for Business Needs and IT Alignment (2005 onwards).
  - CIM Phase of MDA – and assorted MDD
  - Visual Development Environments (VIDE) – 2006 onwards..

Analysis (SA & OOA)

"There's a big temptation to believe that you can describe the application domain and the machine all together, in one combined description. . . . . .

But if you only make one description, you'll surely be tempted to put things into it that describe only the machine, and to leave out things that describe only the application domain. After all, you have to describe the machine sooner or later, don't you ?

You can see the results clearly in many object-oriented modelling descriptions. Often they are accompanied by fine words about modelling the real world. But when you look closely you see that they are really descriptions of programming objects, pure and simple. ***Any similarity to real-world objects, living or dead, is purely coincidental.****"*

Michael Jackson [JACK95]

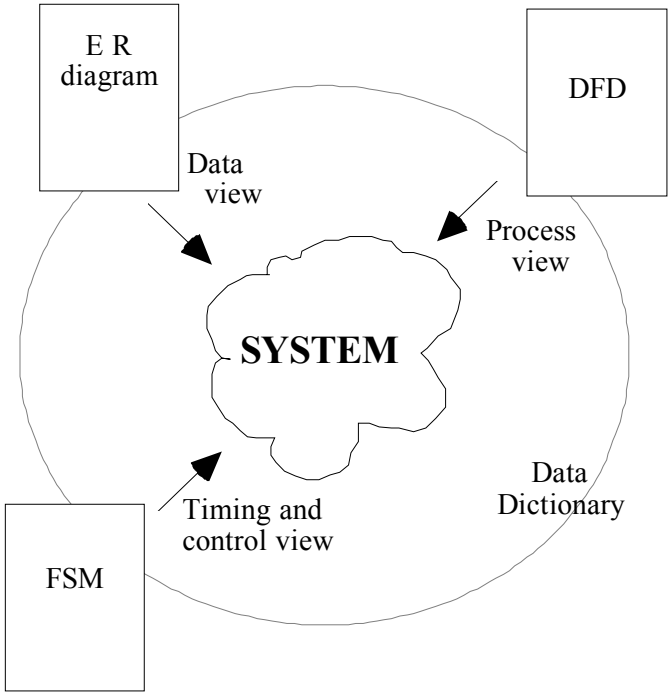
Structured Analysis

Familiar with SA/SD?

Move to graphical modelling notations; partitioned, levelled and minimally redundant.

- Structured Systems Analysis (Gane & Sarson in 1977).
- Structured Design (Yourdon and Constantine 1979).
- Tom deMarco's: Structured Analysis and System Specification from Yourdon Press in 1979.
- Many years taught Yourdon's "Modern Structured Analysis", 1989.

- *Too many models???*



Yourdon SA / SD Overview

Essential model ‘the essence of the system’.

This contains an Environmental Model (describing the environment) and a Behavioural model.

Environmental model consists of:

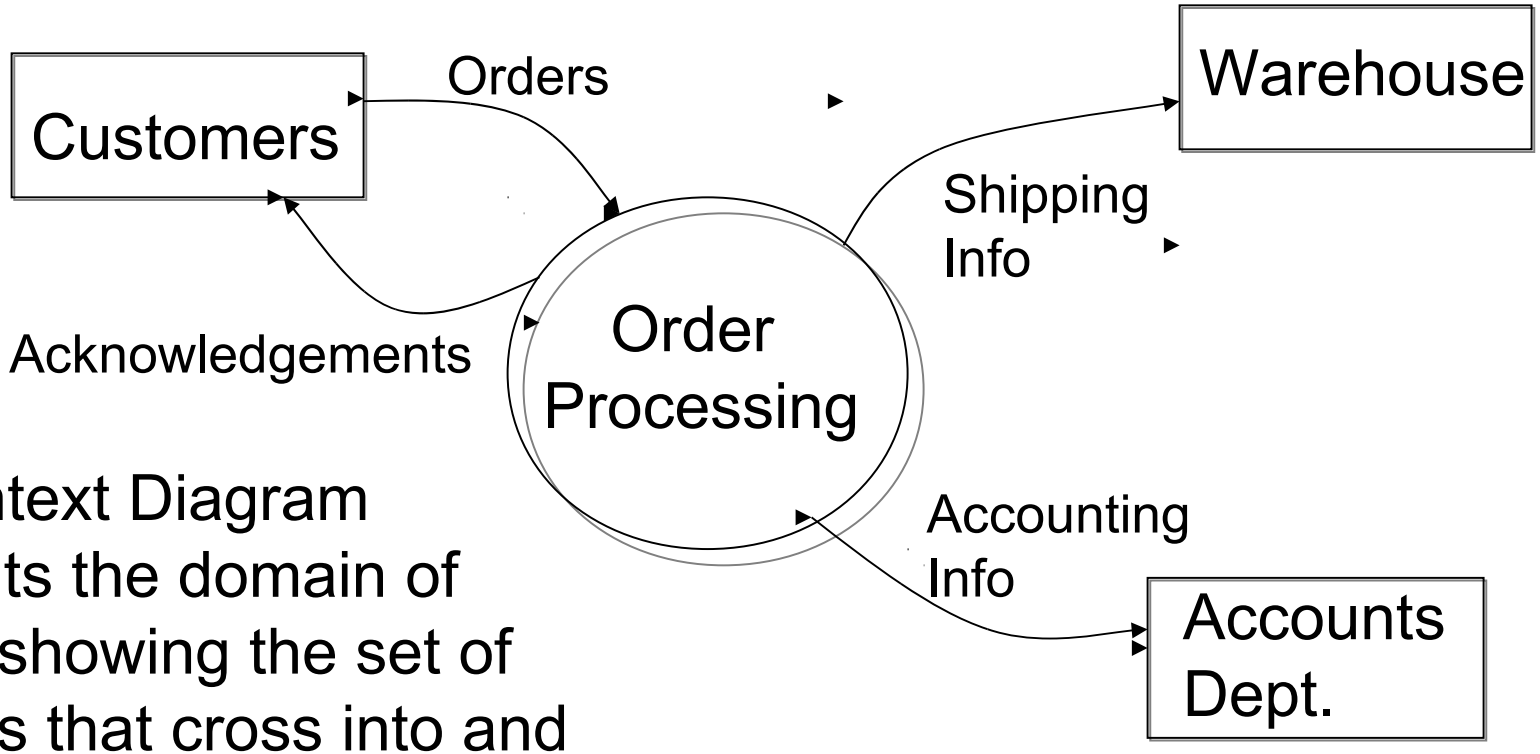
- **Context Diagram**, Statement of Purpose & Event List.

BUT “analysis paralysis” (bigger problems, ever “enhanced” method), dubious value added.

Heavy bias towards DFDs: often amounts to procedural specification. *(Sometimes want that)*

Problems where no pre-existing system.

Context Diagrams



"The Context Diagram documents the domain of study by showing the set of data flows that cross into and out of the domain."

Tom DeMarco, *Structured Analysis and System Specification*, 1978.

Problem and System

DeMarco's 'domain of study' is the system.

"Boxes (representing sources and sinks) are used rather sparingly in Data Flow Diagrams, and usually not in a very rigorous fashion. *Since they represent something outside the area of our major concern*, they exist only to provide commentary about the system's connection to the outside world." - DeMarco.

In Structured Analysis, the Context Diagram shows the context of the *system*, not the context of the *problem*.

Contrast with Problem Frames (later), or process models.

NB: Still very useful to delineate OUR area of interest or system scope.

Object Oriented Analysis

- Seamless Development
  - Object-oriented programming promised the idea of encapsulation and this presented a way to represent things in the real world.
  - A software object could be represented as a design object which was a representation of an analysis object.
  - The assumption was that to represent the real world as an object was simple.

Objects in Analysis?

- Often objects in the ‘problem domain’ are endowed with properties which real world objects would never exhibit.
  - When did you last send a message to a paycheck?
  - What reply would you get back if you sent a message to an aeroplane?
  - What methods does a tax return perform in response to messages it receives?
  - When the sun rises, does it send a message to each bird to tell it to start singing?

  - this last one is taken from a common critique – but, arguably, in a way, the Sun does send a message.

OO Again (or is it UML?)

- Bray notes:
  - The lift controller. Analysis has a ‘floor object’, which has methods within it.
  - YRRS: Has methods within people in the problem.
- In addition OOA (and the UML in general) does not allow understanding of behaviour, process, or dependencies among events, *(use cases)*, all of which are important in understanding user processes.

Bournemouth University

Objects are fixed

Each object belongs to a fixed class, determined when the object is created.

But the world is not like this:

pupils become teachers

Students become graduates: well some of them ☺

Bills become laws

Partnerships become corporations

Doctors become lawyers

Cotton mills become offices or hotels

Caterpillars become butterflies

Bournemouth University

Inheritance

Each object inherits properties and behaviour from just one class at the next level up in the tree. That's single inheritance. (Unless we have C++).

But the world is not like this:

The logistics manager wants to classify the company equipment as production plant, office equipment and distribution vehicles.

The finance director classified it as owned, rented and leased.

The two classifications can't coexist in the same single-inheritance hierarchy.

Active and Passive

Objects are reactive rather than active. If you don't send a message to an object, it won't do anything.

But the world is full of individuals, like

People (who may initiate interactions)

Vessels in chemical plants

Government departments

who all do things spontaneously.

All these restrictions have programming solutions - but they are not so good at representing the richness of the world.

Bournemouth University

Story So Far

SA/SD – concentration on system context.

Tendency to model existing system (rather than the ‘essence’ or ‘business need’, or even the process.

Still provides useful notational tool-kit

OOA. Real world objects? Not really.

Use cases: Oh dear, oh dear oh dear…

So what else is there – for requirements?

Bournemouth University

Implications for Requirements

- Need to understand the (whole) problem and the client goals, needs and processes.
- Need to identify the system boundary (and model interactions outside and across).
- Elicitation to directly produce requirements
- Use of process models (existing or new) to inform requirements.
- All before use cases and specification.

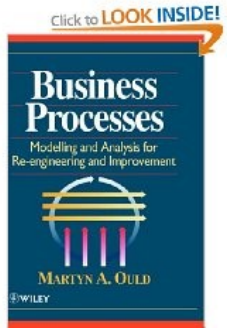
Role Activity Diagrams

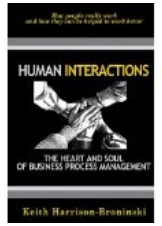
Original concepts in paper by Ould & Roberts (1986), book by Ould 1995 (a great read).

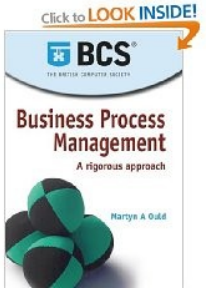
- Initially, promoted by Praxis & Coordination Systems (Roberts), and the DTI sponsored IOPTClub.
- Variants and extensions, e.g., PROCESS project (Southampton Uni, 94-97) produced families of models (mapping to CSP) and enactable models (RolEnact 98).
  - Phalp, K.T., G.K. Abeysinghe, P. Henderson, and R.J. Walters, (1998), RolEnact: Enactable Models of Business Processes, Information and Software Technology, vol. 40, num 3.

- Recent resurgence of interest, with popular books by Keith Harrison Broninski and Martin Ould (both BCS).
- Still supported by many, e.g., see Venice Consulting (Martin Ould's site), for much of interest.

www.veniceconsulting.co.uk







Bournemouth University

What do process actors need to know?

‘For an individual (or group) in the organisation to carry out their activities, they need to know what activities they must take part in, in what order those activities must take place, what other individuals or groups they must interact with, and which actions are dependent upon those interactions’.

Handy, C. (1976), ‘Understanding Organisations’, Penguin.

Role Based Models

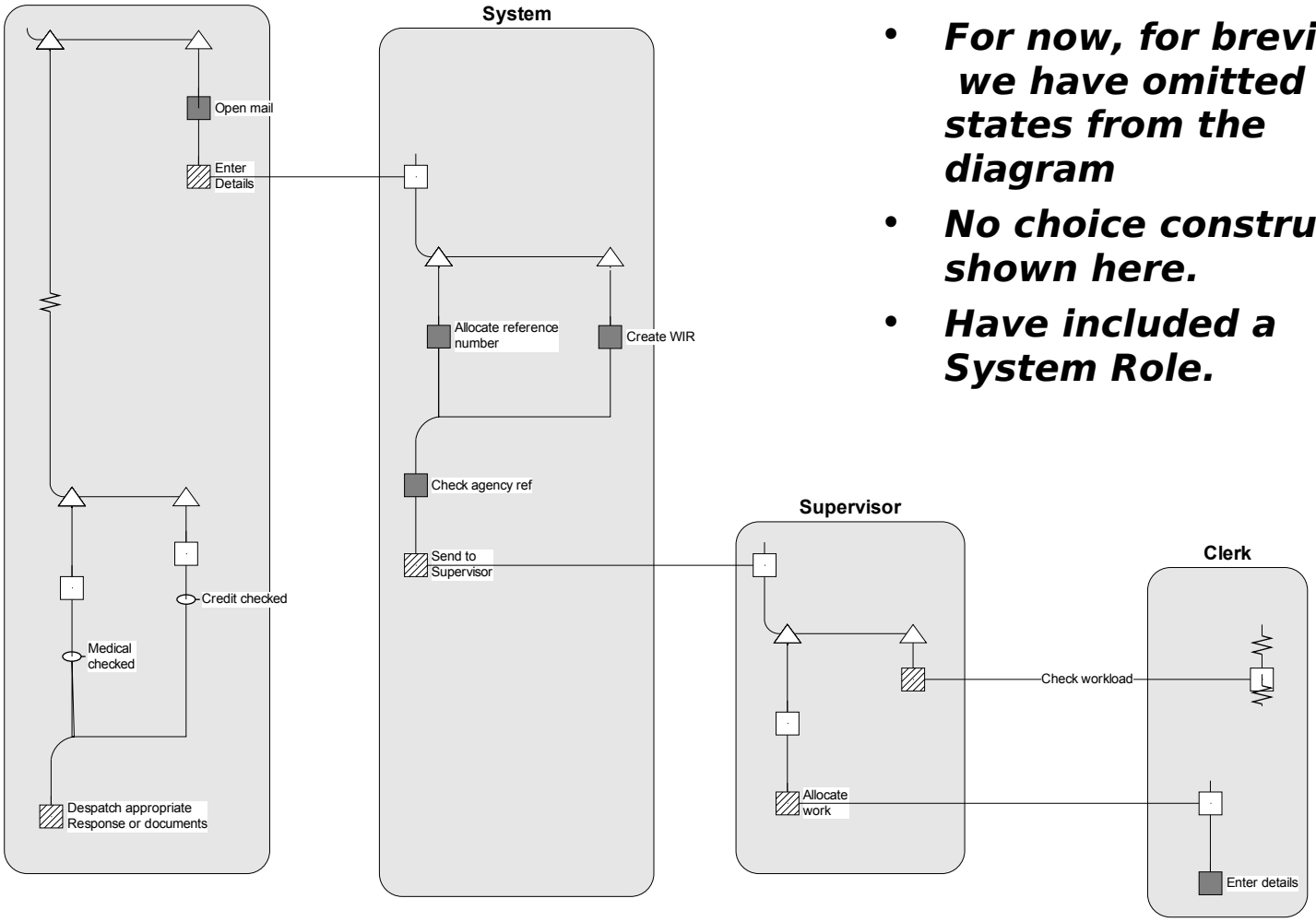
‘Role based models satisfy these requirements by grouping activities into ‘roles’, which describe the desired behaviour of individual groups, or systems’.

Ould, M.A. (1992), An introduction to Process Modelling using RADs, in IOPTCLUB Practical Process Modelling, Mountbatten Hotel, Monmouth Street, Covent Garden, London.

‘A role involves a set of activities which, taken together, carry out a particular responsibility or set of responsibilities’.

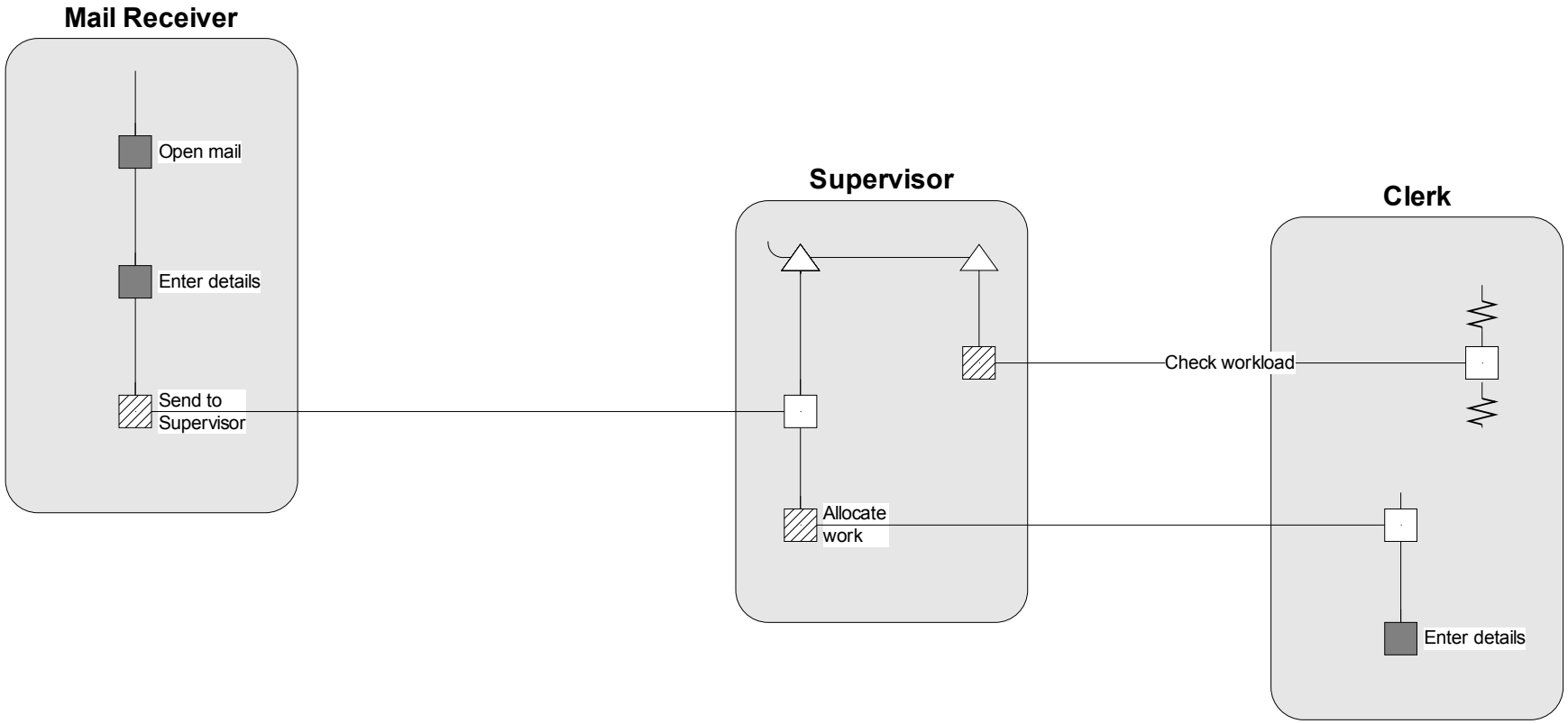
Ould, M.A. (1995), Business Processes modelling and Analysis for Re-engineering and Improvement, John Wiley & Sons.

Role Activity Diagrams



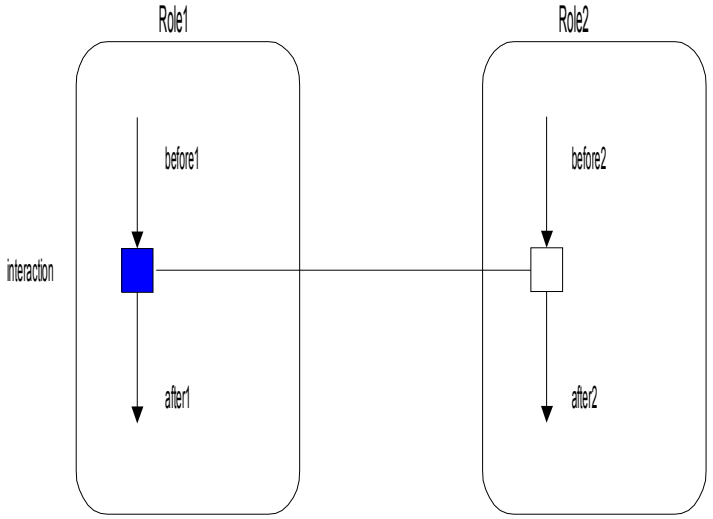
- ***For now, for brevity, we have omitted states from the diagram***
- ***No choice constructs shown here.***
- ***Have included a System Role.***

A ‘Purist’ RAD approach



The importance of *Interaction*

- Activity (or activities) carried out at the same point as another activity in another role (or roles).
- A shared event.
- The consequence of an interaction is that all of the roles involved move from their current state to their next state.
- Act as points of synchronisation (or control) of the process.
- Interactions are synchronous.



Issues in moving to specification

ISSUES

Process models and software models have different perspectives and languages.

For say RAD to use case, they can be considered almost orthogonal.

Activities assigned to roles versus actors (roles) assigned to processes.

Hence difficult to preserve mapping in moving from process model to software.

In addition, there may be information loss if our software constructs are not sufficiently powerful (rich enough).

SOLUTIONS (lots of different ones of course)

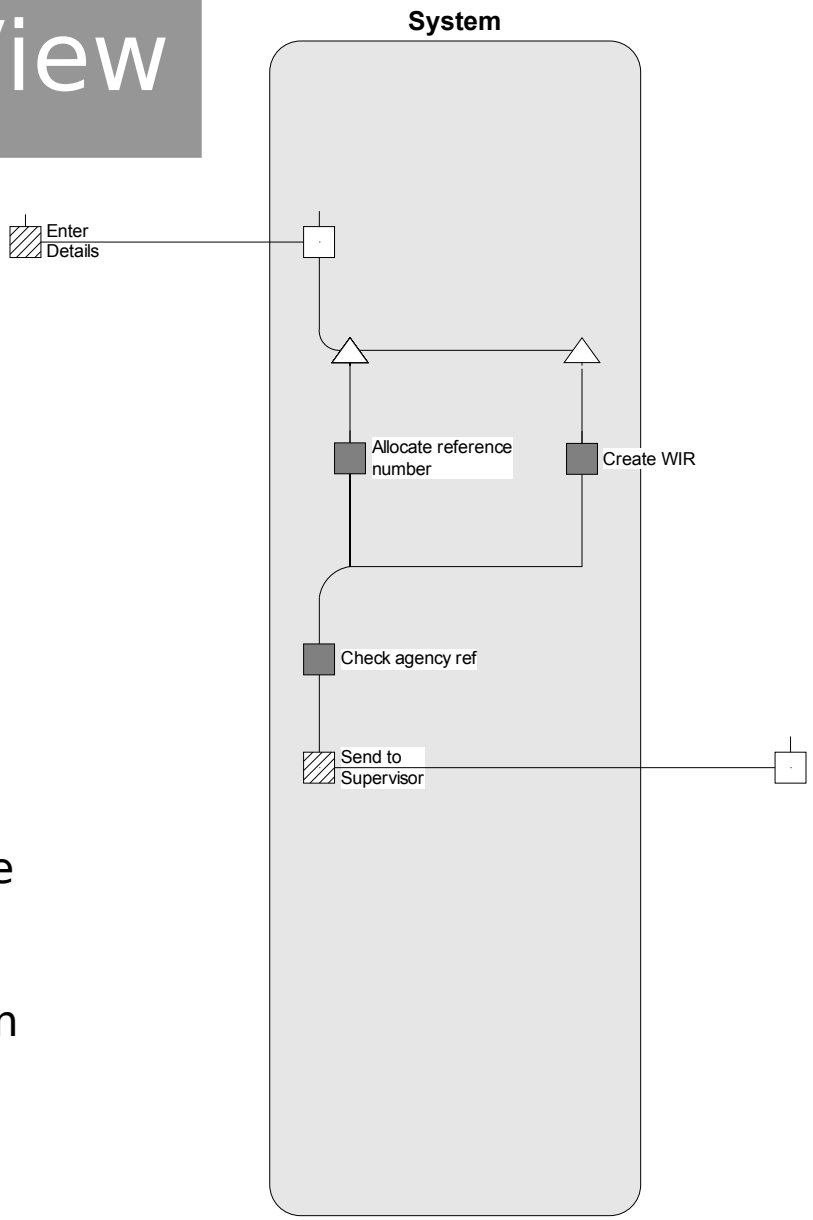
A very thorough approach, which can be found in BSCP (Business Strategy Context Process) paper, moves from strategic view, using goal models, problem frames, and process models (Role Activity Diagrams).

S. Bleistein, K. Cox, J. Verner and K. Phalp (2006
B-SCP: a requirements analysis framework for validating strategic alignment of organisational IT based
, Information and Software Technology, 48 (9), pp.846-868.

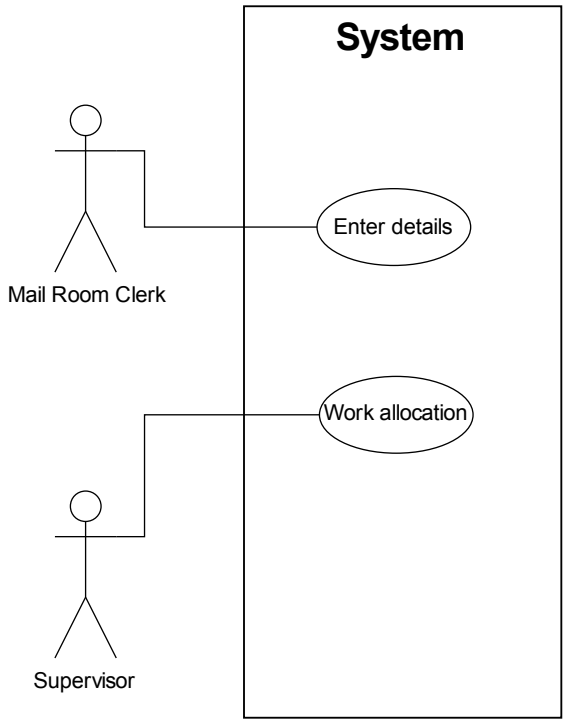
Present here aspects of a lightweight view (often used in teaching), more pragmatic and accessible (simple notational devices).

Software View

- This is against the purist approach, and a rather simplified (teaching) example.
- We (as software engineers) move towards specification.
- Need to ensure that we capture the system boundary (as with say a Yourdon Context Diagram).
- Need to ensure that, in moving to spec, we show cross boundary (problem to machine interactions).
- With a system RAD (usually will have different sub-system names) the interaction is between the roles (which will be actors) and the system role.
- This will correspond to use case communications or associations.



A simplified use case



- Hence, RAD acts as a way to consider the problem domain (inform requirements).
- RAD (with system roles) allows one to 'discover' or discuss the system boundary.
- Acts as a link between business view (intentions for system) and IT.

Practical

- Acts as a checklist for the specification.
- Gives a first cut list for the use case diagram communications.
- *Of course the meat of the use case is in the description, which brings further considerations...*

What have we covered?

Analysis and Requirements are both very misunderstood

Traditional approaches (UML, SASD, objects, use cases) have their place, but are not suited to requirements.

Need to use process models (note that we will justify the choice of process models, RADs, and role models in another lecture.

Briefly introduced RADs – so that we can get under-way with modelling asap.

Of course there are issues in moving from process model to use case.

Hence, devised and introduced mechanisms, of which a simple one is presented and can be used to move from RADs to Use Case.

NEXT TIME – Process models properly...

Role Activity Diagrams

Overview of the assignment