From Process Model to Problem Frame - A Position Paper

Karl Cox

Computer Science, University of New South Wales,

National ICT Australia

karlc@cse.unsw.edu.au

Keith Phalp

Bournemouth University, UK

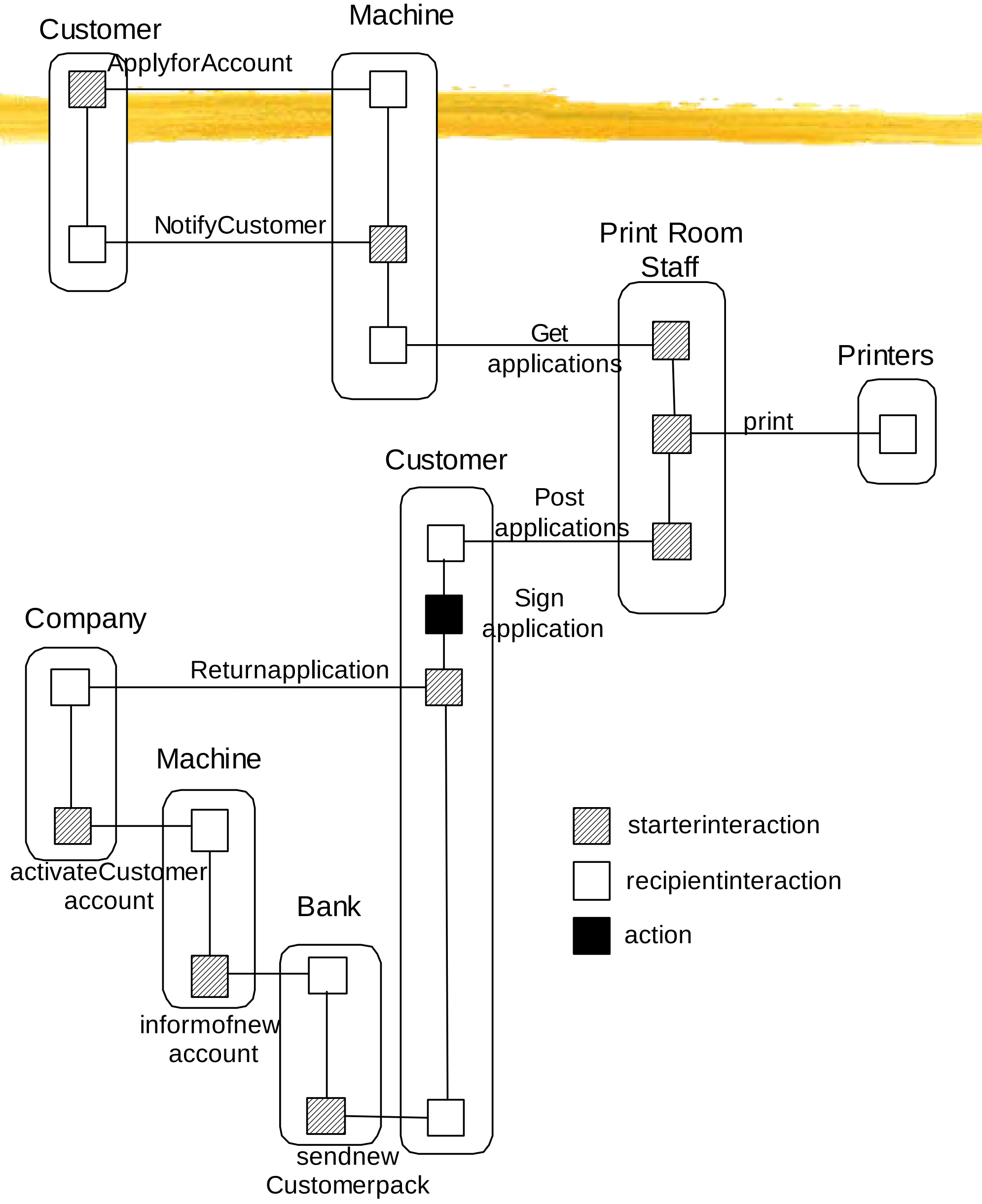
kp@bmth.ac.uk

Rationale

- There is still difficulty in bridging the business - software gap
- Companies that model software product processes might wish to change their ‘legacy’ systems for different software systems.
- Problems frames are an approach to understanding what elements are involved in the change and will provide for a best-fit solution structure.

Our Proposal

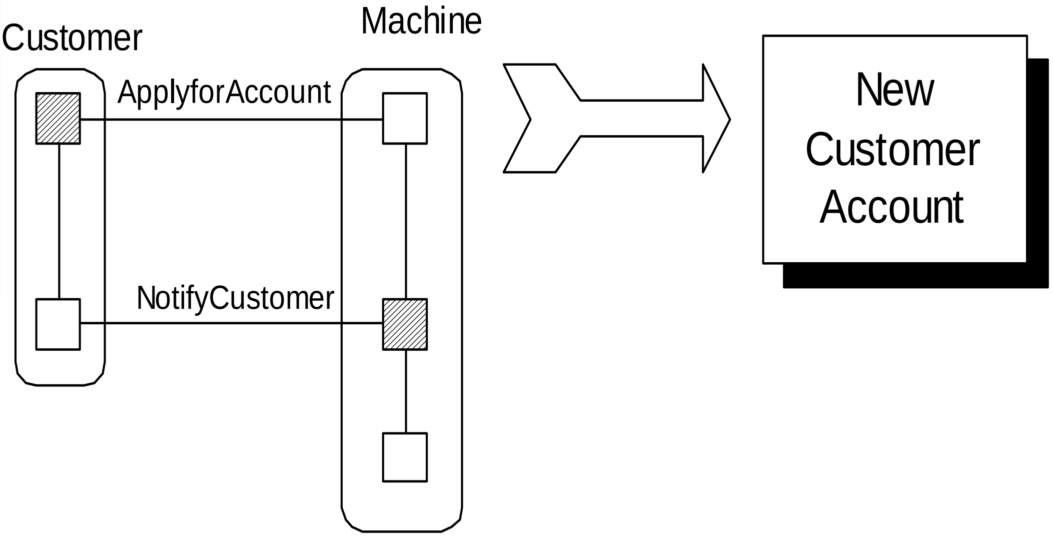
Step	Action
1	Describe Role Activity Diagram



Our Proposal

Step	Action
1	Describe Role Activity Diagram
2	Identify outcomes of interactions

The interaction ‘Apply for Account’ has the outcome of creating a new Customer Account

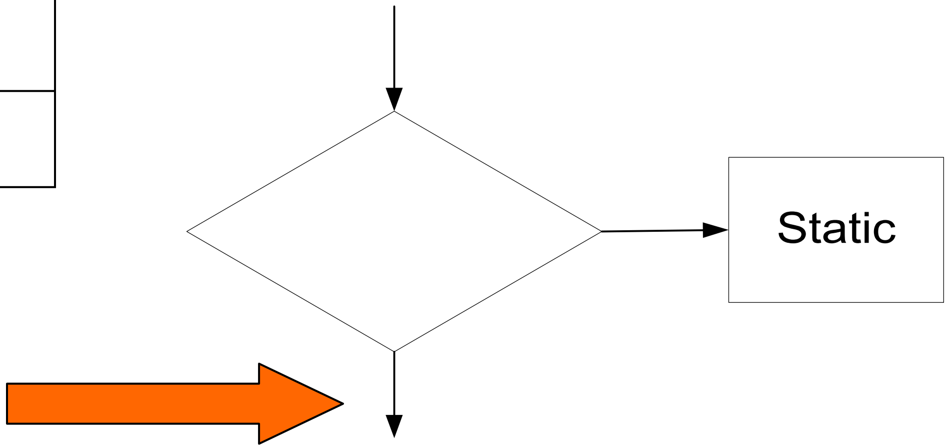


Our Proposal

Step	Action
1	Describe Role Activity Diagram
2	Identify outcomes of interactions
3	Identify potential domains from outcomes

What kind of domain is it?

We use Ian Bray’s domain taxonomy here.



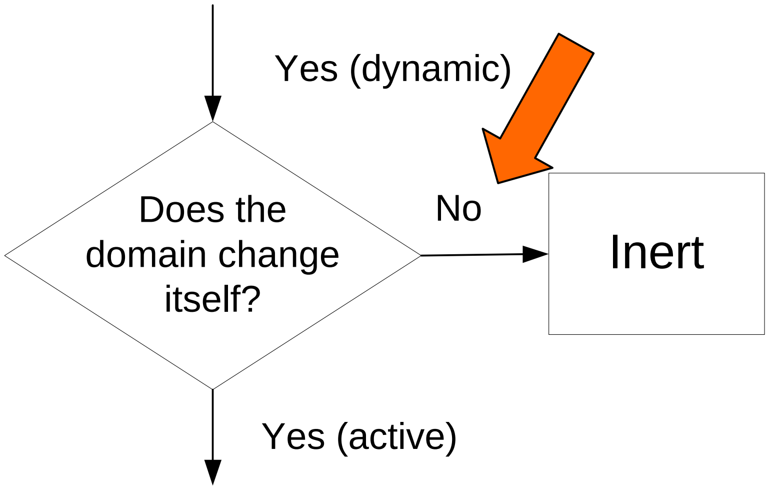
Our Proposal

Step	Action
1	Describe Role Activity Diagram
2	Identify outcomes of interactions
3	Identify potential domains from outcomes

Our search stops here. The Customer Account in **inert**. It changes over time but only at the behest of its owners (Customer and Bank)

So the domain changes with time – it is not static.

In our case, the Customer Account will be manipulated over a period of time.



Yes (dynamic)

Does the domain change itself?

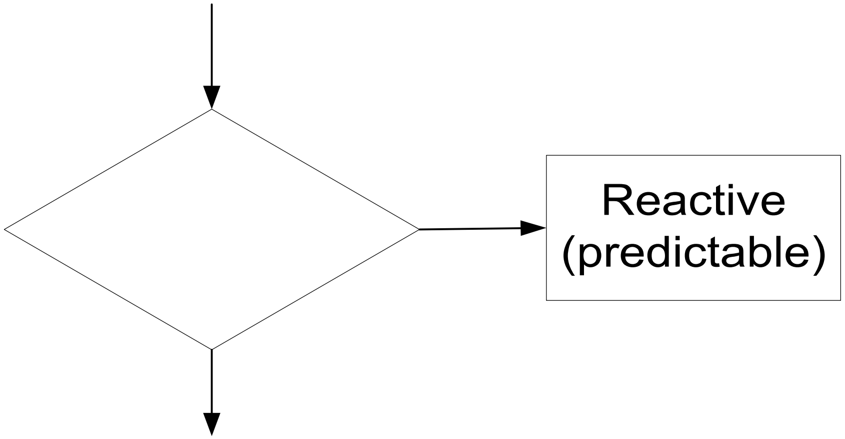
No

Inert

Yes (active)

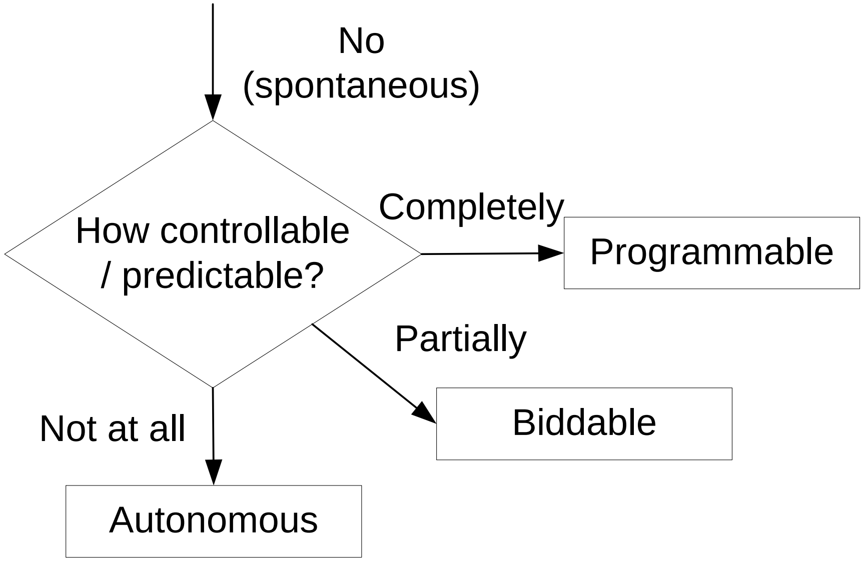
The rest of the taxonomy

Step	Action
1	Describe Role Activity Diagram
2	Identify outcomes of interactions
3	Identify potential domains from outcomes



The rest of the taxonomy

Step	Action
1	Describe Role Activity Diagram
2	Identify outcomes of interactions
3	Identify potential domains from outcomes



We then ask our inert Customer Account domain where it resides: Primarily it is housed within a computer.

This makes it a design domain – but we must check the requirements.

Our Proposal

Step	Action
1	Describe Role Activity Diagram
2	Identify outcomes of interactions
3	Identify potential domains from outcomes
4	Identify potential rules that govern interactions

YES, we have an inert, design domain

The Requirements!

(sort of)

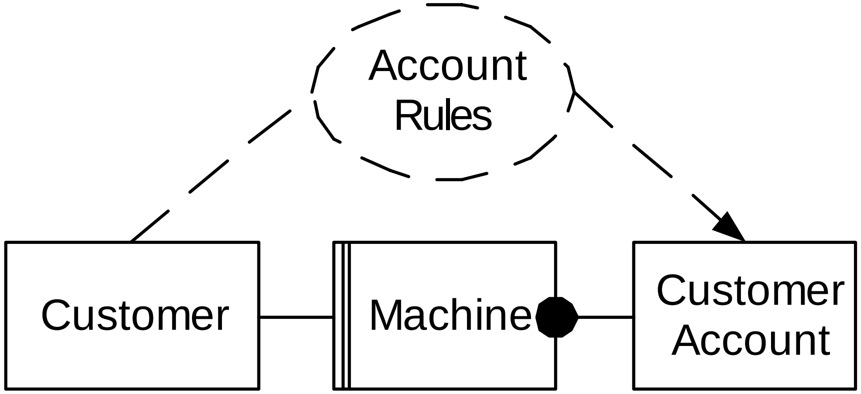
R1. The Customer must enter financial information online to create the new account.

R2. Credit status must be checked before the account is created.

R3. Finance laws govern the nature of the details the Customer must provide and the Company must adhere to ...

Our Proposal

Step	Action
1	Describe Role Activity Diagram
2	Identify outcomes of interactions
3	Identify potential domains from outcomes
4	Identify potential rules that govern interactions
5	Identify problem frames



We have an inert design domain (Customer Account) that is governed by financial law as well as company requirements that is created by the Customer: A Workpiece Problem Frame.

Any other frames?

- Yes, in this re-engineering project we identified:
  - an Information Systems frame
    - The Customer can check stocks and shares prices - it’s a web browser in this sense
  - a Commanded Behaviour frame
    - The Customer can buy and sell stocks and shares and the machine must behave according to its behaviour rules (updating stock prices in real time and automatically - a required behaviour frame)
    - Maintenance can update / alter Customer files…

Validity Threats

- Any Empirical Evidence?
  - This is an exploratory idea though the RAD (and subsequent elicitation of problem frames) came from a real industry project we worked on.
- Who uses Role Activity Diagrams?
  - A lot of companies! But the point is we are in the very early phases of this research so choose the RAD as our ‘starter’ exemplar notation.
- What about other domain modelling approaches, analysis/design patterns?
  - Good point. We are getting to these!

Work to be done

- What's the point of this research anyway?
  - When systems evolve it is not always easy to know what kind of system is required in its place. Problem frames can help in this regard.
- So we plan to
  - Formalise the process for identifying the other problem frames.
  - Conduct empirical studies - case studies with industry to assess the efficacy of the approach in (web and legacy) systems evolution.