Problem Frames

~

a short talk

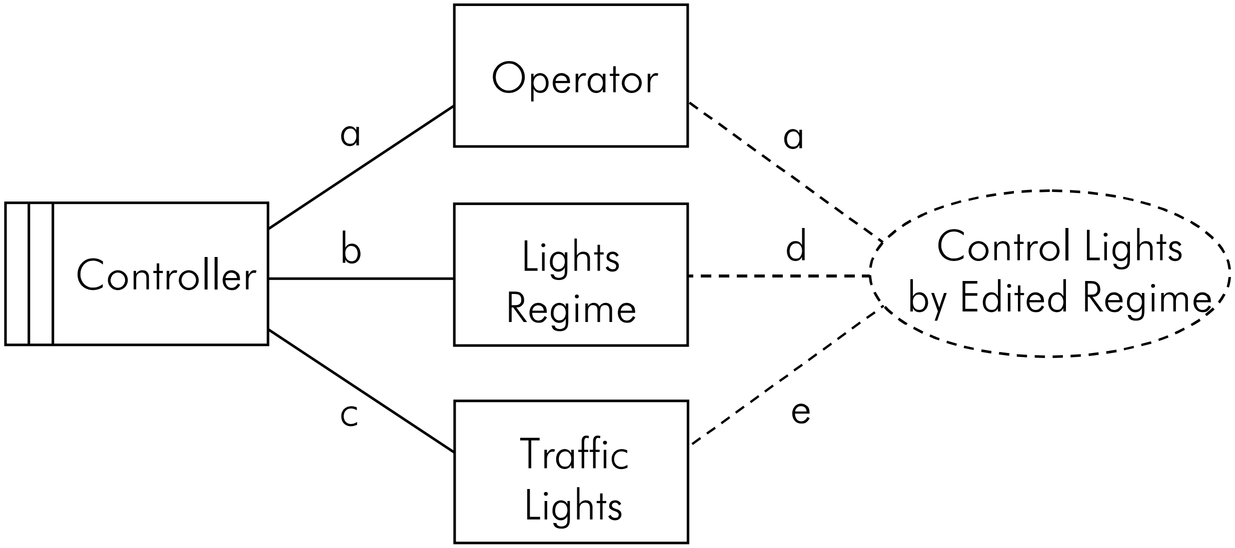
ISRE01 Workshop
25 April 2001
Imperial College

Michael Jackson

jacksonma@acm.org

Software Development Problems

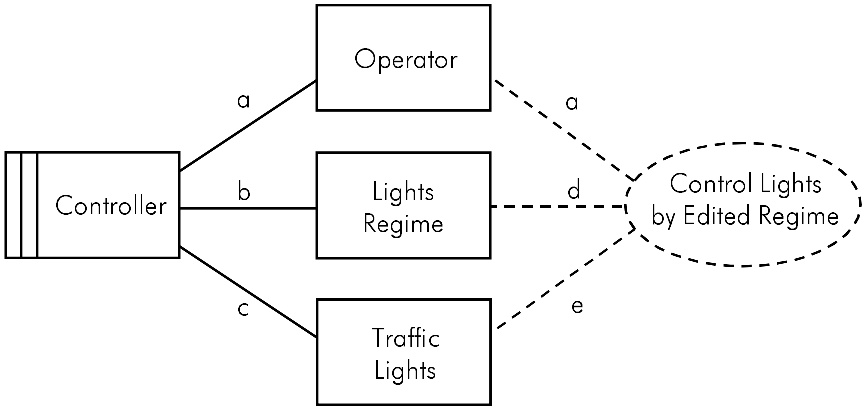
- Developing the software part of the solution to a problem
- The machine and the problem world



- Interfaces of shared phenomena
- The problem world is physical domains — human, lexical, causal
- Requirements are remote from the machine-world interface
- Some problem domains are remote from the machine-world interface

Classifying Software Development Problems

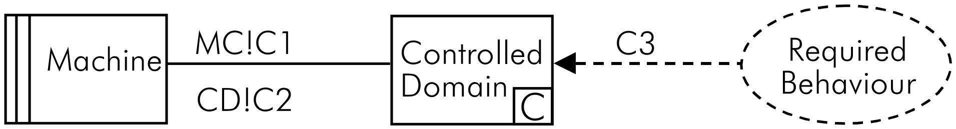
- Different problems have different characteristics, raise different concerns
  - eg: avionics, telephone switching, banking, compiling, word-processing
- We need problem classification
  - A structure for learning, recording, applying lessons of experience
- But realistic problems are heterogeneous



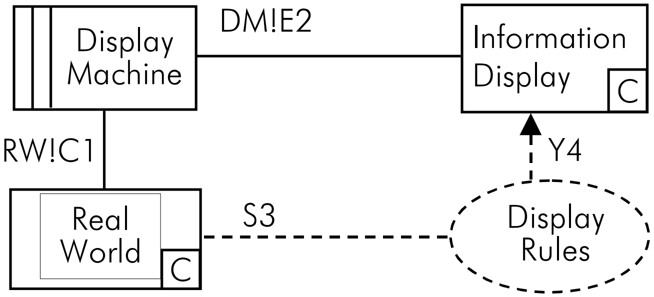
  - Editing the regime is one kind of problem, controlling the lights is another
- So we need (a) to classify subproblems, and
  (b) to decompose and compose problems as structures of subproblems

Some Elementary Problem Frames

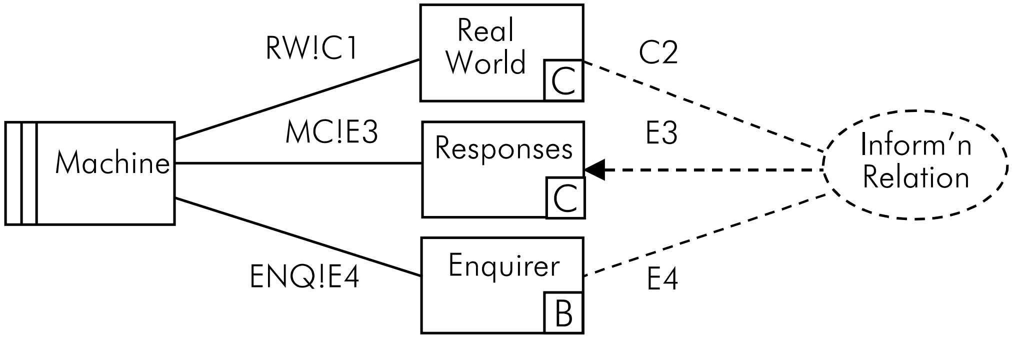
Required Behaviour Frame



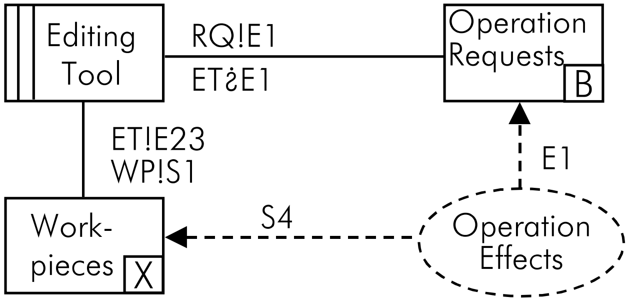
Information Display Frame



Simple Enquiry Frame



Simple Workpieces Frame

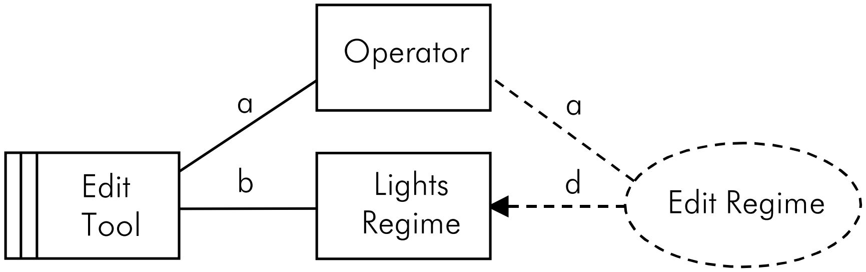


Problem Frames

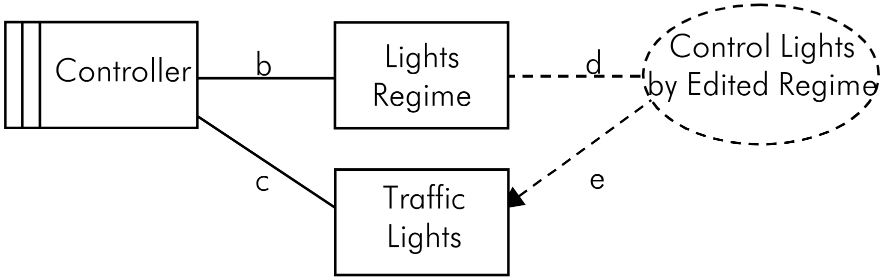
- A problem frame has:
  - A problem diagram with generalised names for domains and phenomena
  - A FRAME CONCERN
    - Constructing the argument necessary to show that the problem is solved
    - Making the descriptions necessary to support the argument
  - Typical PARTICULAR CONCERNS
    - eg: in a Behaviour problem, we must rely on causal properties of the Controlled Domain to satisfy the requirement. What if the Controlled Domain does not exhibit these properties reliably?
- A problem must fit a frame exactly
  - Number and configuration of problem domains
  - Control properties at interfaces
- There are defined variants of elementary frames
  - eg: Simple Behaviour with Operator
  - eg: Described Behaviour

Decomposing a Realistic Problem

- A realistic problem is decomposed as a superposition of simple subproblems
  - Each subproblem must fit a known problem frame



A Simple Workpieces problem



A Described Behaviour problem

- A realistic problem is decomposed as a superposition of simple subproblems
  - Each subproblem must fit a known problem frame
  - Each subproblem is a projection of the whole problem
- The subproblems are treated in isolation
  - Their composition to give the whole problem is an additional task

Subproblem Composition

- The Problem Frames approach departs from a traditional trade-off
  - Traditionally: uniform composition, complexsubproblems
    - eg: all composition is procedure call, or message passing, or CSP, or …
    - But each subproblem must be solved in the composition context
      - Essentialy, the composition concern is distributed among thesubproblems
  - With frames: heterogeneous composition, simplesubproblems
    - The Workpieces subproblem is solved in isolation
    - The Described Behaviour subproblem is solved in isolation
    - The composition concernis addressed when the subproblems are known
- Lights Control example
  - Composition concern has (at least) these two aspects
    - Mutual exclusion, because Lights Regime domain is assumed static for the Controller
    - Scheduling, because not all pairs of valid Regimes can be concatenated
      - eg: $<\text{Stop}, \overrightarrow{\text{Go}}, \text{Stop}, \overleftarrow{\text{Go}}> \,\hat{}\, <\text{Stop}, \overleftarrow{\text{Go}}, \text{Stop}, \overrightarrow{\text{Go}}>$ is OK

        but $<\text{Stop}, \overrightarrow{\text{Go}}, \text{Stop}, \overleftarrow{\text{Go}}> \,\hat{}\, <\overrightarrow{\text{Go}}, \text{Stop}, \overleftarrow{\text{Go}}, \text{Stop}>$ is not OK

Principles of the Problem Frame Approach

- Focus on the problem world
  - The traditional focus on the machine has been very harmful
- Focus on phenomena, not abstractions
  - The approach is inappropriate for purely mathematical problems
- Guided decomposition
  - It's no use decomposing an unfamiliar problem into unfamiliar subproblems
- Frames provide a structure for recording and exploiting lessons of experience
  - Experience with problems fitting particular frames
  - Experience with compositions of particular pairs, triples, etc of frames
- Separating concerns: subproblem vs composition
  - Composition is a separate concern
  - Composition while subproblems are not yet understood is premature