Simplicity and Problem Understanding

Michael Jackson
jacksonma@acm.org

ICSE ’09
Vancouver
May 19, 2009

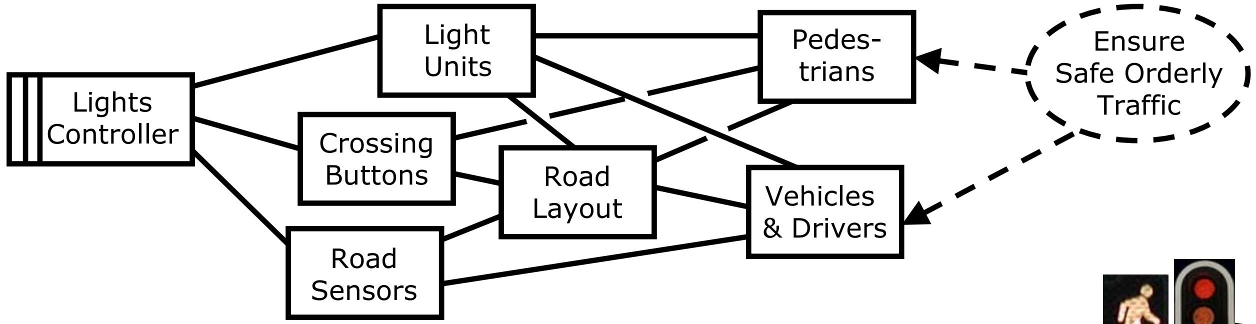
- Software-intensive systems
  - Radical, not normal
- Functional requirements
  - Structure
  - Elaboration
- Development as a problem
  - To understand, not construct
- Understanding
  - A process, not a state

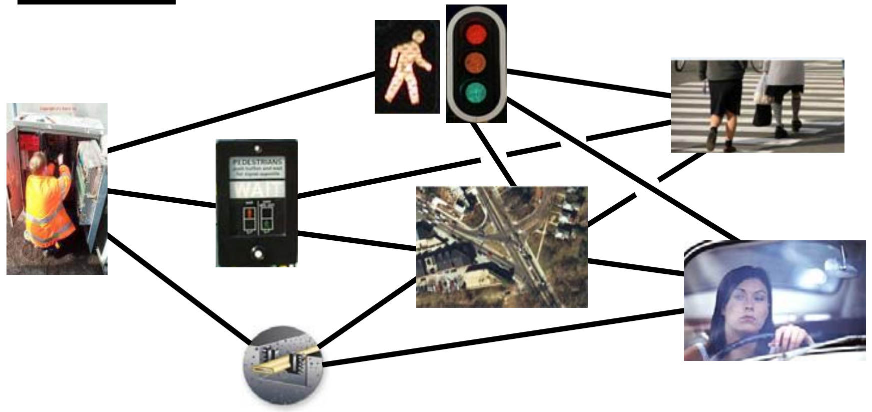
Software-intensive systems

- For lifts, library, traffic, pacemakers, banks, …

- Systems are contrivances
  - Purposeful assemblies of problem and software domains
- Most systems are complex
  - Features, feature interaction, automation, non-formal heterogeneous problem world, system interoperation, …

A traffic-control system

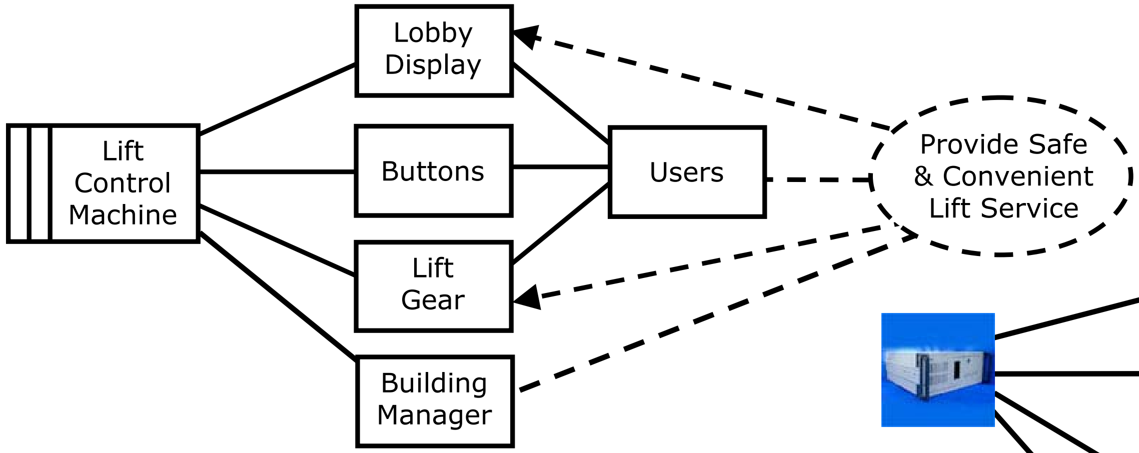


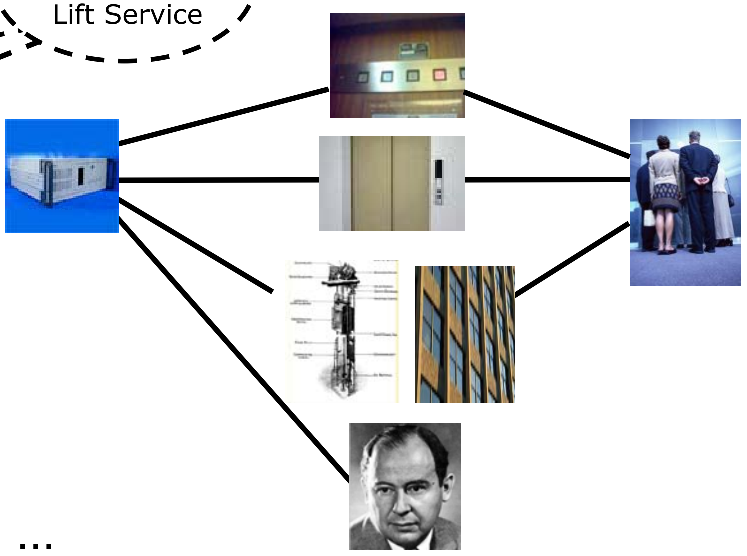


- Problem domains interact …
  - … with each other and with the software
- The purpose is a desired behaviour of the problem world
  - The software must achieve the purpose

Systems and problems

A lift-control system

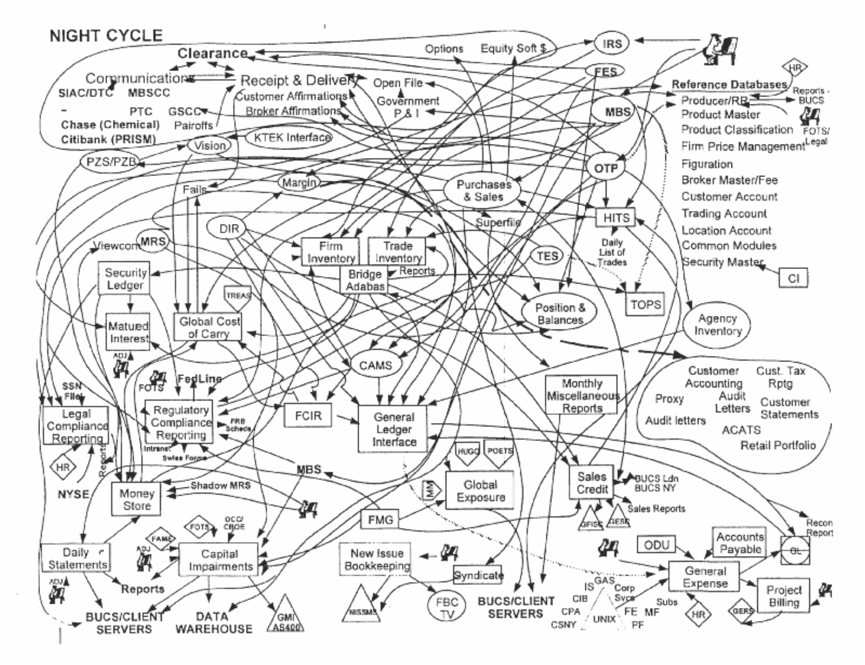




- Understanding complexity needs …
  - Formalisation, reasoning and calculation about domain properties and behaviours
  - In a well-chosen structural framework

Systems and problems

What kind of structural framework?



Automated overnight clearing system for US bank transactions

- Is a structural framework just the top level of abstraction from detail?
    - One refinement path from top to bottom?
- Or is it something different?

Understanding machines (ie contrivances)

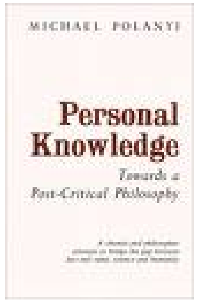
A particular kind of understanding

"We have seen that a tool, a machine or a technical process is characterised by an operational principle, which differs altogether from an observational statement."

"… the operational principle of a machine … [specifies] how its characteristic parts—its organs—fulfil their special function in combining to an overall operation which achieves the purpose of the machine. It describes how each organ acts on another organ within this context."

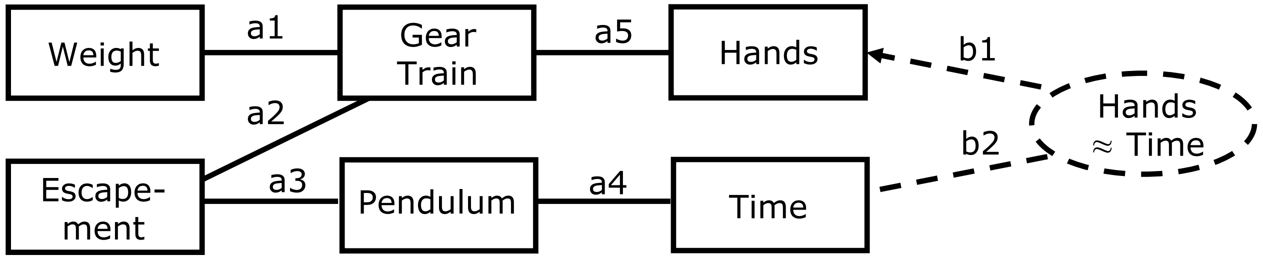
"… The first thing to realize is that a knowledge of physics and chemistry would in itself not enable us to recognize a machine."

Michael Polanyi;
Personal Knowledge, 1958

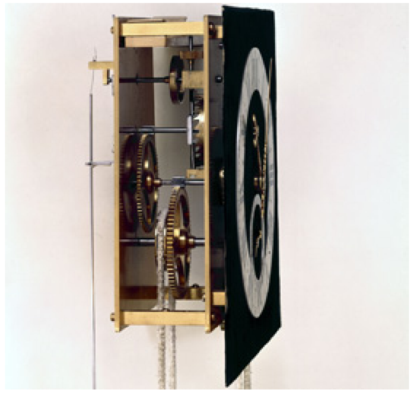


Understanding machines (ie contrivances)

An operational principle: the pendulum clock

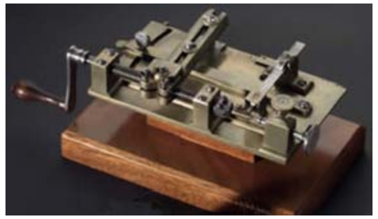


“The falling weight drives (a1) the gear train which drives (a2) the escapement; on each swing the pendulum releases (a3) one tooth and receives (a3) an impulse; hands, driven by (a5) gear train, count swings; roughly constant swing period (a4) ensures hands (b1) count elapsed time (b2) as required.”



Understanding machines (ie contrivances)

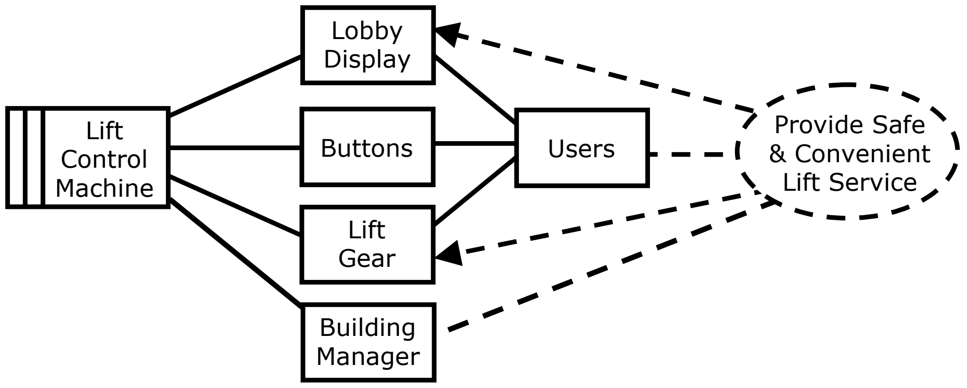
Scientific knowledge and contrivances

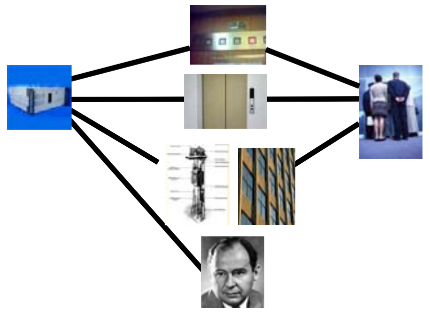


- Operational principle:
  - What is it? What's it for?
  - How the parts interact to achieve the purpose
  - A framework for detailed analysis
- Mathematical and scientific knowledge:
  - Underlying laws
  - Context necessary for success
  - Part properties necessary for success
  - Possible improvements
  - Specific failures and causes
  - Feasibility of a proposed contrivance

Software-intensive systems

Complexity and the operational principle

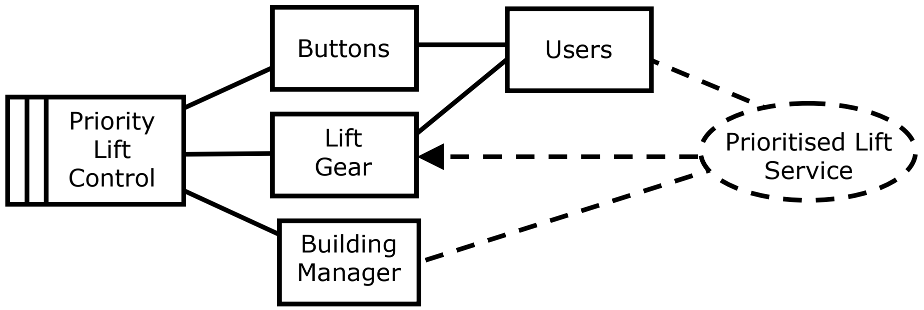


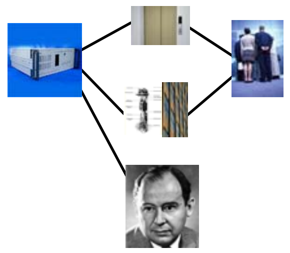


- No single operational principle
  (a requirement is not an operational principle)
- Decompose into simpler parts
  - Contrivances with simple operational principles
- What kind of decomposition?
  - Requirement or instrumental decomposition
  - In both cases, free decomposition
  - In both cases, problem decomposition

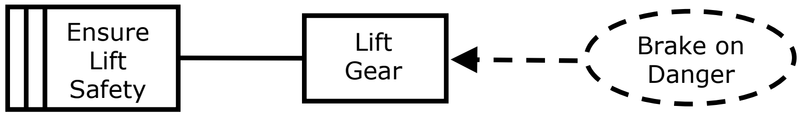
Requirement decomposition

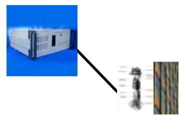
Provide lift service as prioritised by the building manager



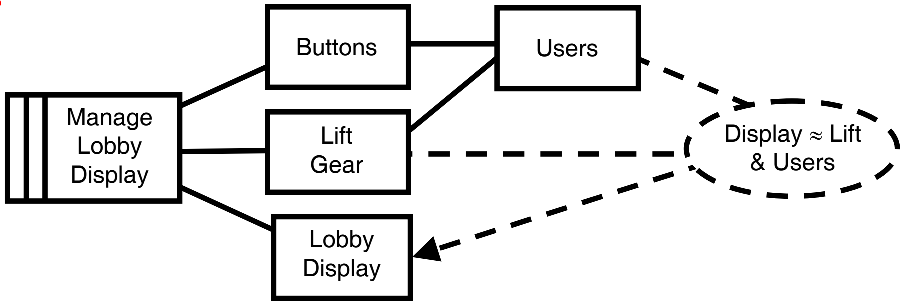


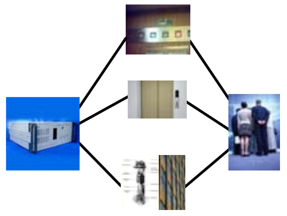
Brake on danger when fault found in lift equipment





Operate lobby display of outstanding requests and lift positions for users' information

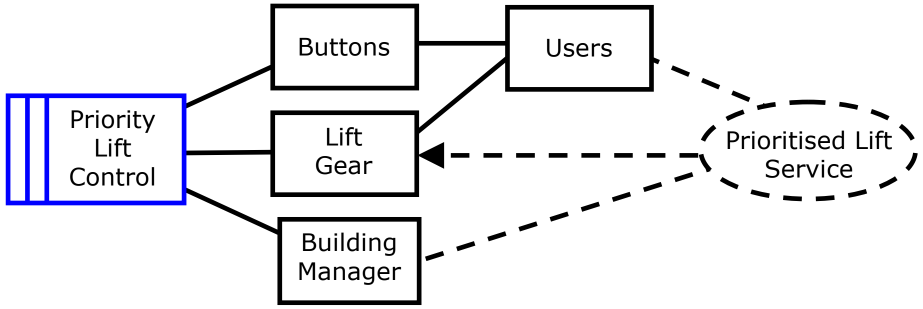




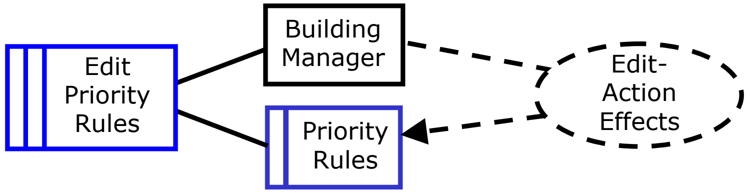
Decomposition

Instrumental decomposition

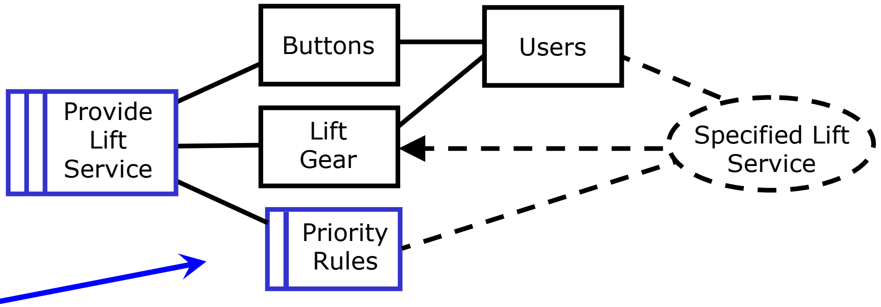
Provide lift service as prioritised by the building manager



Edit service priority rules



Provide prioritised lift service



- Priority Rules: a local variable of Priority Lift Control machine

Free decomposition

- The focus
  - Problem not solution
  - Identify subproblems
- Sources of complexity
  - Intrinsic + interaction
  - Ignore interactions
- Parts are simple
  - Simple operational principles

Simplicity

Simple operational principle

- Some unities of contrivance
  - One level of purpose
    - eg: Air traffic control
  - A closed operational purpose
    - eg: Railway management
  - One role for each problem domain
    - eg: Text-editing centre
  - Synchronous operation
    - eg: Accounting periods
- A general principle
  - '1-pass' explanation of operational principle

Simplicity

Local context of a subproblem

- Some unities of context
  - One level of domain abstraction
    - eg: Car park barrier commands
  - Consistent domain properties
    - eg: Healthy / faulty lift equipment
  - One operational phase
    - eg: Aircraft taxi, take-off, climb, …
  - Local context is constant
    - eg: Book loans and membership
- A general principle
  - Subproblem assumes its context

Top-down and bottom-up analysis

- Top-down decomposition
  - Requirement decomposition identifies sub-requirements
  - Instrumental decomposition simplifies operational principle
- Subproblem analysis
  - Uses operational principle …
  - .. as framework for sufficient analysis
- Bottom-up recombination
  - Addresses composition concerns
  - May modify result of subproblem analysis
  - May add new subproblems (eg scheduler)

Recombination

Composition concerns

- Some composition concerns
  - Interleaving
    - eg: Edit priority rules vs Lift Service
  - Requirement elaboration
    - eg: Book loans vs member status
  - Requirement conflict
    - eg: Inter-library vs member loan
  - Switching
    - eg: Lift Service to Emergency Action
  - Domain sharing
    - eg: Phone display: camera vs gps vs email …

Envoi

Concluding remark

- A principle of Descartes

  "...to conduct my thoughts in such order that, by commencing with objects the simplest and easiest to know, I might ascend by little and little, and, as it were, step by step, to the knowledge of the more complex; assigning in thought a certain order even to those objects which in their own nature do not stand in a relation of antecedence and sequence."

Rene Descartes;
Discourse on method; Leyden, 1637

- A radical project is a learning project

Thank you