ICS 121 Lecture Notes

Architectural Design

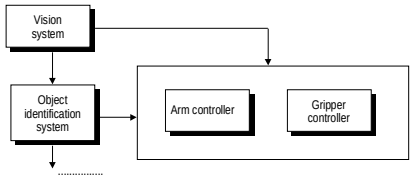
ICS 121

- Architectural Design
  - Decomposition of large systems into sub-systems that provide some related set of services + establishing a framework for sub-system control and communication
- Developing a software architecture is a problem of mastering complexity
  - Different design methodoligies produce different decompositions
- No generally accepted process of architectural design, but some subprocesses:
  - *System structuring*: Stucturing of the system into a number of sub-systems, where a sub-system is an independent software unit
  - *Control modelling*: Establishment of a general model of control relationships between the parts of the system
  - *Modular decomposition*: Each identified sub-system is decomposed into modules

Architectural Design: System Structuring - 1

ICS 121

- Decomposition of the system into a set of interacting sub-systems
- Most abstract model may be represented as a block-diagram (each box represents a sub-system)
- Example: Packing robot control



Architectural Design: System Structuring - 2

ICS 121

- More specific models show
  - how sub-systems share data
  - how they are distributed
  - how they interface with each other
- Three standard models:

❶ The *repository model:*

- Sub-systems must exchange information
  - All shared data is held in a central database (i.e. repository)
  - Each sub-system maintains its own database (i.e. message passing to exchange data)
- Advantage: Efficient way to share large amounts of data, activities (backup, security, access control, recovery from error) are centralized
- Problem: Integration of new sub-systems with different data models

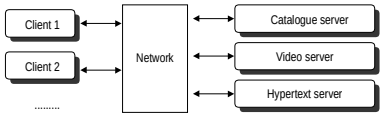
Architectural Design: System Structuring - 3

ICS 121

❷ The *client-server model:*

- Distributed model which shows how data and processing is distributed across a range of processors
  - Servers offer services to other sub-systems
  - Clients call on the services offered by servers
  - A network allows the clients to access the services offerd by servers
- No shared data model (each server must take responsibility for data management activities)
- Easy integration of new servers in the network
- Register of server names and services required

Example:



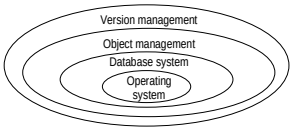
Architectural Design: System Structuring - 4

ICS 121

❸ The *abstract machine model:*

- Also called layered-model: Models the interfacing of sub-systems
- Organizes a system into a series of layers each of which provides a set of services
- Each layer defines an abstract machine
- Examples: OSI reference model, Ada programming support environment (APSE) [Buxton,1980]

Example: Version management system



Architectural Design: Control Models

ICS 121

- To work as a system, sub-systems must be controlled so that their services are delivered to the right place at the right time
- Control models are concerned with the control flow between subsytems
- Two general approaches:

❶ Centralized control: One sub-system has overall responsibility for control and starts/stops other sub-systems

❷ Event-based control: Rather than control information being embedded in a sub-system, each sub-system can respond to externally generated events (from other sub-systems or the environment)

Topic 8 Architectual Design

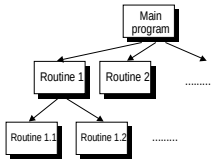
ICS 121 Lecture Notes

Architectural Design: Control Models - 2

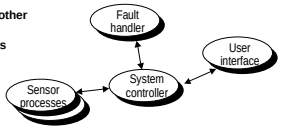
ICS 121

❶ **Centralized Control:**

(a) ***Call-return model***:
- Familiar top-down subroutine model
- Control starts at the top of a subroutine hierarchy
- Only applicable to sequential systems



(b) ***Manager model***:
- One system component is designated as a system manager and controls starting/stoping/coordination of other system processes
- Applicable to concurrent systems



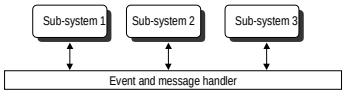
Architectural Design: Control Models - 3

ICS 121

❷ **Event-based control**
- Event-driven systems are driven by externally generated events (in contrast to the values of some system state variables)
- Event is not only a binary signal, but may have a range of values
- Timing of the event is outside the control of the process which handles that event

Two examples of event-driven control models:

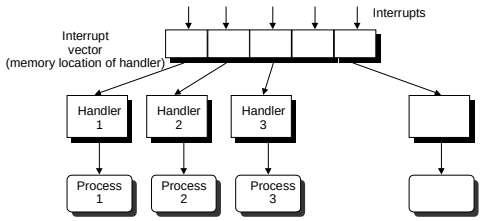
(a) ***Broadcast model***:
- An event is, in principle, broadcast to all sub-systems
- Any sub-system which is designed to handle that event repsonds to it (registration with event handler required)



Architectural Design: Control Models - 4

ICS 121

(b) ***Interrupt-driven models***:
- External interrupts are detected by an interrupt handler and passed to some other component for processing
- Exclusively used in real-time systems (fast response time)



Architectural Design: Modular decomposition

ICS 121

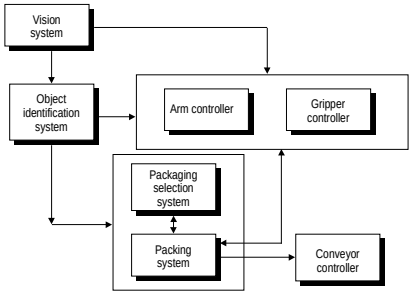
- After decomposition of the system into sub-systems, sub-systems may be decomposed into modules
- No rigid distinction between system decomposition and modular decomposition
- Two important models for decomposing sub-systems into modules:
  - ***Data-flow model***: The system is decomposed into functional modules which accept input data and transform it to output data
  - ***Object-oriented model***: The system is decomposed into a set of communicating objects
- Hierarchies support the decomposition of sub-systems into modules

System Structuring: Example

19a

ICS 121

- Packing robot control



C2 - Properties

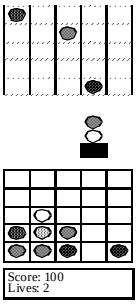
ICS 121

- A C2 architecture is a hierarchical network of concurrent components linked together by connectors (message routers) according to specific rules.
- A component is aware of services only from components "above" it.
- The definition allows
  - substitution of components
  - reuse
  - extensibility

Topic 8 Architectual Design

C2 KLAX Example

ICS 121



KLAX Chute
Tiles of random colors drop at random times and locations.

KLAX Palette
Palette catches tiles coming down the Chute and drops them into the Well.

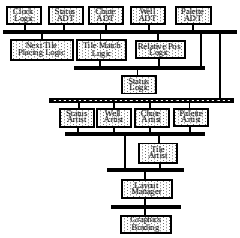
KLAX Well
Horizontal, vertical, and diagonal sets of three or more consecutive tiles of the same color are removed and any tiles above them collapse down to fill in the newly-created empty spaces.

Score: 100
Lives: 2

KLAX Status

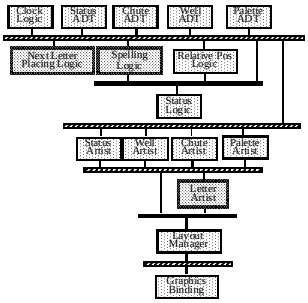
C2 KLAX Architecture

ICS 121



C2 Modified KLAX Architecture

ICS 121



Topic 8
Architectual Design