Systems Integration

Models

Pere Palà

iTIC   `http://itic.cat`

v1.1 November 2015

Source: A significant part is from Mark W. Maier and Eberhardt Rechtin's *The Art of Systems Engineering 3rd Ed*

Inspiration

By relieving the mind of all unnecessary work,
a good notation sets it free to concentrate
on more advanced problems,
and in effect increases the
mental power of the [human] race.

Alfred North Whitehead

Introduction

Models

- Primary means of communication with clients, builders and users
- Help for construction (development and refinement)
- Help describe and diagnose its operation
- Architect has to be fluent with all the languages spoken: specifiers, designers, manufacturers, certifiers, distributors and users.
- Model defining the critical acceptance parameters
- Model of overall structure of the system
- Classification: Models by role or by content

Roles of Models

- Communication with client, users and builders
- Maintenance of system integrity
- Design assistance
- Guiding of decomposition and aggregation of functions, components and objects
- Performance prediction
- Provision of acceptance criteria for certification

Models by View

Perspective of View	Description
Purpose or objective	What the client wants
Form	What the system is
Behavioral or functional	What the system does
Performance objectives or requirements	How effectively the system does it
Data	The information retained in the system and its interrelationships
Managerial	The process by which the system is constructed and managed

Objective and Purpose Models

- Match desirability of purposes with practical feasibility
- Architect's first step: identify objectives and priorities
- Precise/abstract objectives
- If trade-off decisions are architectural drivers they must be maintained: they will be repeatedly revisited!
- Restate and iterate initial unconstrained objectives in user's language. Translate them into modeling language
- Behavioral objectives, performance objectives...
- Cathedrals in Middle Ages: *Communicate the glory of god and reinforce the faithful*

Models of Form

- Represent physically identifiable elements of the system to be built
- Aesthetic feel, dimensionally accurate model to ensure mechanical interfacing
- Scale models (buildings, cars, racing yachts, software prototypes...)
- Block diagrams. Models of form: correspondence with physically identifiable elements of the system
  - Interconnect: Specific physical elements connected by physically identifiable channels
  - Flow: Same as before, specifying information flow
  - Structure charts
  - Manufacturing process diagrams
- Flow control usually not shown but important

Behavioral Models

- What the system does (opposed to what it *is*)
- Unlike a building: looking at a scale model gives no clue
- Balance on the level of detail: too little provides no understanding / too much may be incomprehensible (or too late)
- Example: user interfaces of software systems
- Threads: sequences of events and actions
  - What the system must produce
  - Which sequence does not occur
- What should happen with this input at this moment? Track till the output
- Which inputs are required to get this output? Track till the input
- Use cases: sample dialogue between system and an actor

Behavioral Models /2

- Data and event flow networks
  - Enormous set of threads: collapse into compact model
  - Usually graphical hierarchy with finite state machine descriptions
- Mathematical Systems Theory
  - Definition of boundary: observable quantities
  - Mathematical tools
  - Continuous systems. Linear, well understood. Nonlinear ?
  - Temporally discrete systems (sampled). Similar understanding
  - Discrete event systems. Existing frameworks: state machines and Petri Nets. Incomplete

Behavioral Models /3

- Autonomous agents, neural nets, chaotic systems
  - Few types of multiply replicated, relatively simple, components
  - Ant colonies
- Public choice and behavior models

Performance Models

- Predict how effectively an architecture satisfies some objective (functional or not)
- Cost is not behavior of the system, but an important objective!
  - Analytical models
  - Simulation
  - Judgmental
- Formal methods: develop systems that provably perform as specified
  - Identify inputs and outputs. Identify relations between inputs and outputs
  - Decompose iteratively into components and code at the lower level
  - Compose back
- Used in software and digital design
- Impractical (controversial) for large systems
- Mathematical statements. Difficult. Difficult to understand by client

Data Models

- What data is retained? Relationships?
- Data models based on database developments
- Example: ITS. Intelligent light, needs simple data. Data fusion from many sources required for full deployment
- Based on Entity-Relationships diagrams

Managerial Models

- Project: technical exercise and milestones, budgets, schedules...
- Socio-technical systems: planning deployment more difficult than assembling hardware
- Familiar tools of project management
- Client needs a solid cost and schedule estimate