16.882/ESD.34J - System Architecture

ESD Section

Fall 2001

Edward F. Crawley, Professor of Aeronautics and Astronautics, Department Head
Olivier L. de Weck, Assistant Professor of Aeronautics and Astronautics and ES

Learning Objectives

General

Students will be able to apply the principles, methods, and tools of system architecting such that they will be able to

- structure and lead all the early, conceptual phases of the system development process, and
- support an ongoing project through it's subsequent system engineering and design phases.

To prepare students for their first, second, and third jobs after the end of their studies.

Systems and Products

Students should be able to

- explain what a system is and how behavior emerges
- demonstrate "system thinking" (holism, balance, focus)
- explain what a product is, how it creates value and competitive advantage
- identify the common features of a generic Product Development Process (PDP), and understand variants of the PDP for a given product
- identify where the architecting process sits in the PDP, and its role in establishing value and competitive advantage

Architecture

Students will be able to

- define system architecture
- identify the architecture of systems, critique them, and learn from them
- create architectures for new or improved systems
- produce deliverables of the architect needed to define the architecture of a system

Architecting

Students will be able to execute the role of a system architect.

Creative / Critical Process Thinking

Students will be able to

- compare existing architecting approaches
- create new approaches
- analyze old and new approaches, and synergize a "best" approach
- think creatively and "out-of-the-box" when necessary
- develop a personal set of guiding principles for successful architecting

Strategy

Therefore, the course is structured into four phases:

1) Introduction (9/7-9/17)
    - the definitions used in the subject
    - the Product Development Process
    - the concepts in product architecture

2) Analysis of System Architecture (9/21-10/19)
    - analysis of architectures
    - frameworks for thinking holistically,
    - processes upstream and downstream of architecting
    - methods of critical evaluation of JAMs ("just another method") and contemporary tools

3) Synthesis of System Architecture (10/22-11/26)
    - concept maps function to form
    - alternative and "out of the box" alternatives thinking for yourself
    - approaches to creativity, ambiguity, and complexity

4) Advanced Topics and Conclusion (11/30-12/10)
    - underlying and enduring principles of system architecture, and
    - regulatory influences and technology infusion, and
    - advanced topics (platforms, reuse, legacy systems, supply chain)

A detailed **course schedule** is **handed out separately** in the first lecture.

Definitions

Architecture (alt):

The structure, arrangements or configuration of system elements and their internal relationships necessary to satisfy constraints and requirements. (Boppe)

Architecture (alt):

The arrangement of the functional elements into physical blocks. (Ulrich & Eppinger)

Architecture (alt):

The embodiment of concept, and the allocation of physical/informational functionality and definition of structural interfaces among the elements. (Crawley)

Holistic:

of the whole. To think **holistically** is to encompass all aspects of the task at hand, taking into account the influences and consequences of anything that might interact with the task.

Principles, Processes and Tools:

Architects use **principles**, **processes** and **tools**. **Principles** are the underlying and long enduring fundamentals. **Processes** are the organization of methods and tasks to achieve a concrete end, which should be solidly grounded on principles. **Tools** are the contemporary ways to facilitate process.

System: A system is a physical or virtual object that performs a function which cannot be fulfilled by its constituent parts alone and that distinguishes itself from its environment by a system boundary.

Product: Products are systems that have value.

Interface: The points at which elements of form connect.

The **Product Development Process**:

The inclusive process of creating a new or modified product, bringing it to "market" and supporting its life-cycle.

A separate glossary of **engineering systems-ESD definitions** (draft) will be handed out during the first lecture. These definitions are not all directly related to System Architecture, but provide additional information.

The **Architect and Architecting:**

Architecture exists for all products and systems. **Architecting** is a function. It may or may not be associated with a person explicitly called "the **architect**".

Architecting is the most abstract, highest level function in product / system development process. **Architecting**

- is done by the smallest number of people (sometimes less than one),
- has some of the greatest impact on eventual success,
- factors in the greatest number of considerations,
- is not primarily concerned with detailed or quantitative data.

An **architect** must be able to think holistically, and:

1. Define boundaries, and establish goals and functions,
2. Create the concept which maps function to physical / logical elements
3. Define decomposition, abstraction hierarchy and inter-element interfaces.

An **architect** is not a generalist, but a specialist in simplifying complexity, resolving ambiguity, and focusing creativity.

The approaches that architects follow during their work are varied. Generally speaking:

- no single method will work,
- out of the box thinking often bears fruit,
- must (in principle) be able to deal with and consider everything,
- must concentrate on and trade the essential things.

Roles of the Architect

1. The architect defines the boundary of the "closed system" which constitute the design of the system and its implementation process.
   Specifically, the architect defines the goal(s) and function(s) by:

   - interpreting corporate strategy,
   - interpreting corporate marketing strategy, and competitive analysis
   - listening to "customers" or their representative
   - infusing technology available
     - in platforms,
     - at the company, and
     - from other sources,
   - interpreting regulatory and pre-regulatory influences, and
   - is sensitive to product liability and intellectual property issues.

2. The architect creates the concept for the system.

- proposes and develops options,
- identifies key metrics and drivers,
- conducts highest level trades, and optimization
- thinks holistically about the entire product life cycle in terms of
  - design
  - implementation (sourcing and manufacturing)
  - operation
  - product and process
  - risk management
  - sustainability
- anticipates failure modes and plans for mitigation and recovery

3. The architect allocates functionality and defines interfaces and abstractions

- decomposes form and function
- allocates functionality to elements
- defines interfaces between subsystems,
- configures the subsystems - creates the structure of the system while considering:
  - flexibility vs. optimality
  - modularity vs. platform
  - vertical vs. horizontal strategies, and
  - in-house vs. outsourcing design and manufacturing

Deliverables of the Architect

The architect will deliver

- A clear, complete, consistent and attainable (with 80%-90%confidence) set of goals (with emphasis on functional goals)
- A functional description of the system, with at least two layers of decomposition
- A concept or concepts for the system
- A design for the form of the system, with at least two layers of decomposition
- A notion of the timing and operator attributes, and the implementation and operation plans
- A document or process which ensures functional decomposition is followed, and the form at interfaces is controlled

Staff

Subject Faculty:

Edward Crawley	Lecturer	33-207	253-7510	crawley@mit.edu
Olivier de Weck	Lecturer	33-406	253-0255	deweck@mit.edu

Prof. Crawley's Administrative Assistant:

Kathi Grace	33-207	253-3251	kathi@mit.edu

Prof. de Weck's Administrative Assistant:

Fran Marrone	33-409	253-4885	franm@mit.edu

Teaching Assistant:

Jay Jootar	E60-266	253-4670	jootar@mit.edu

General Lecture Times:

DAY: Monday,
TIME: 2:30pm-4:30pm
ROOM: **33-116**

DAY: Friday,
TIME: 11:30am-1:30pm
ROOM: **33-116 or 9-057**

Office Hours:

Edward Crawley, by appointment (contact his secretary).
Olivier de Weck, **Wednesdays 1:00-2:00 p.m**. in 33-406 (please email prior).

Books

Required Textbooks (available at the COOP):

1. Rechtin E., Maier M.W., *The art of Systems Architecting,2nd ed.*, CRC Press, Boca Raton, FL, 2000.
2. Ulrich K.T., Eppinger S.D., *Product Design and Development, 2nd ed.*, McGraw-Hill Inc. New York, NY, 2000.

Additional References:

3. Shishko R., *NASA Systems Engineering Handbook*, NASA June 1995, SP-6105.
4. Rechtin E., *Systems Architecting*: *Creating and Building Complex Systems*, Prentice Hall, Englewood Cliffs, NJ, 1991, ISBN 0-13-880345-5
5. Oliver, D. W., Engineering complex systems with models and objects, ISBN 0-07-048188-1
6. Suh-THE PRINCIPLES OF DESIGN, Oxford, 1990, ISBN 0-19-504345-6
7. Leveson-SAFEWARE: SYSTEM SAFETY & COMPUTERS, Addison-Wesley, 1995, ISBN 0-201-11972-2
8. Shaw--SOFTWARE ARCHITECTURE: PERSPECTIVES ON AN EMERGING DISCIPLINE, Prentice Hall, 1996, ISBN 0-13-182957-2
9. Suh, Nam Pyo – AXIOMATIC DESIGN: ADVANCES AND APPLICATIONS, Oxford University Press, 2001, ISBN: 0195134664
10. Adams, J.L. (1986), "Conceptual Blockbusting: A Guide to Better Ideas (3rd ed.)", Perseus Books, ISBN 0-201-55086-5.
11. Buzan, T. (1996), "The Mind Map Book", Plume, ISBN 0-452-27322-6.
12. De Bono, E. (1993), "Serious Creativity: Using the Power of Lateral Thinking to Create New Ideas", Harper Business, ISBN 0-88730-566-0.

Grading:

Opportunity sets (OS)	40%
Case studies (CS)	20%
Active Class Participation	20%
Principles Journal	20%
	100 %

Note: Each Opportunity Set will contain an extra credit question. This can be solved out of pure interest or to offset a missing case study, but it cannot make up for lacking class participation or a missing principles journal.

No exam.
The true test of what you have learned will come in real life, when you are called upon to lead and structure the conceptual and preliminary design phases of a new product or system.

9/6/2001