# Architecture Milestone

CSE 403, Spring 2004

Software Engineering

http://www.cs.washington.edu/education/courses/403/04sp/

# Readings and References

- References
  - » *Anchoring the Software Process*, Barry Boehm, USC, 1995
    - http://citeseer.nj.nec.com/boehm95anchoring.html
  - » *Software Architecture*, David Garlan, CMU, 2001
    - http://www-2.cs.cmu.edu/~able/publications/encycSE2001/
  - » *A Practical Method for Documenting Software Architectures*, Clements, et al, CMU, 2002
    - http://www-2.cs.cmu.edu/~able/publications/icse03-dsa/
  - » *I Have Abandoned My Search for Truth, and Am Now Looking for a Good Fantasy*, Ashleigh Brilliant

# Elements of Lifecycle Architecture (LCA)

- Operational Concepts — What is it?
- System Requirements — What does it do for us?
- System and software architecture — How?
- Lifecycle plan — Who wants it? Who'll support it?
- Feasibility Rationale — Is this really true?

# Sound Familiar?

- These are the same elements as for the Life Cycle Objectives milestone
- Now we are making the system real
  - » No longer just a **`public interface`**
  - » At least a **`public abstract class`**
- Definition of system and software architecture

# Elaboration of Operational Concept

- Detailed system objectives and scope
  - » User community?
    - business, personal, demographic
  - » Environment this program works in?
    - device availability, networking fabric, ...
  - » Major benefits?
    - Given the above, is the user still interested?
  - » Establish what the system does and does not do
    - Now is the time for all the stakeholders to recognize what they are and are not getting - highlight changes

# Elaboration of System Requirements

- *All* features of the system
  - » well defined now or can be defined later with low risk
  - » capabilities, interfaces, appearance
  - » include all out-of-band functions - support, admin, update
- Features include
  - » performance and reliability of particular functions
  - » specifics of security requirements
- Prototypes are an appropriate tool for providing an interpretation of the requirements
  - » be careful that customer/marketing don't get confused about which is prototype and which is the real product

# Elaboration of System Architecture

- Specific choices
  - » make some decisions - you are headed for action
  - » document why you dropped previous options
- Identify specific existing packages that will be used in your product
  - » Commercial-off-the-shelf, in-house, open source, ...
- Identify evolutionary paths
  - » Which packages can be replaced or upgraded?
  - » Where do you anticipate change? Can you support it?

# Hand-Wave Reduction Act

- LCA review
  - » incorporates detailed requirements specification
    - shows that you really know what is being built
  - » incorporates detailed design
    - shows that you know how to build it
- Details
  - » “are the mark of a great con” - Jonas Nightingale
  - » but also important to help you work through how this thing is actually going to work

# Details

- System and Software Components
  - » hardware, programs, data blocks
- Connectors
  - » mediate interactions among components
- Configurations
  - » combinations of components and connectors
- Constraints
  - » resource limitations, operating environment

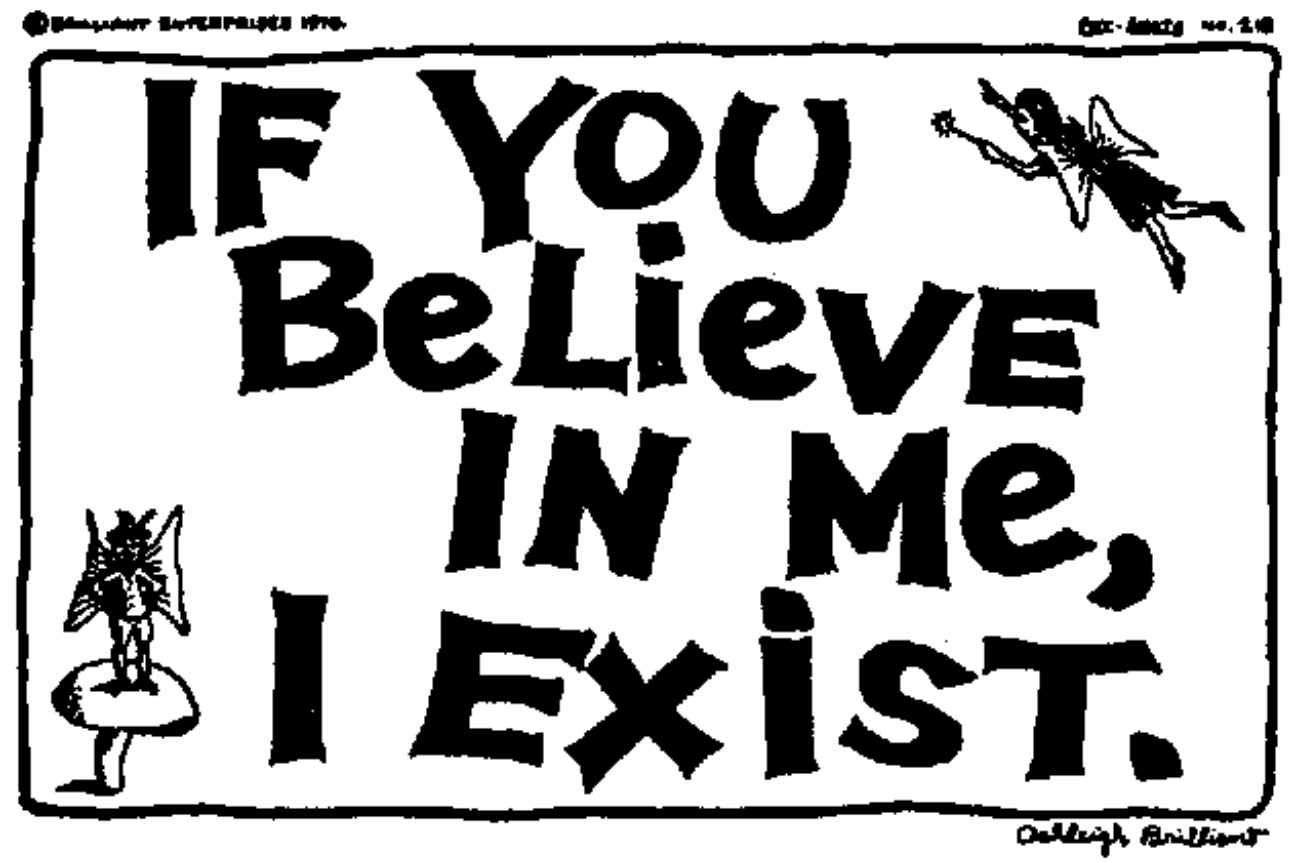
# Elaboration of the Life-Cycle plan

- "The WWWWWHH principle"
  - » Why is the system being developed? Objectives
  - » What will be done When? Schedules
  - » Who will do it? Where are they? Responsibilities
  - » How will the job be done? Approach
  - » How much of each resource? Resources

- This is now the detailed project development plan

# Feasibility Rationale

- Establish the consistency and conceptual integrity of the other elements
  - » ie, Will it work?
- Get the stakeholders' concurrence that the LCA elements are compatible with their objectives for the system
  - » ie, Do the customers and deployers want it?




Ashleigh Brilliant
Pot Shots