Object-Oriented Analysis & Design (OOAD) I Supplement CS577a, Fall 2006

Developed by
David Klappholz, Stevens Institute of Technology
Supannika Koolmanojwong , USC Center for Software Engineering,&
Ed Colbert, USC Center for Software Engineering

Who Am I?

- Research Associate, USC Center for Software Engineering
  - MBASE developer
  - Security Cost Modeler
  - Security Metrics
- 25 years industrial teaching & consulting on object–oriented methods, software engineering, & programming languages
- Consultant on definition of Architecture Design Language (ADL) for real–time, safety–critical systems
  - Based on Unified Modeling Language ("UML") & Honeywell's MetaH
  - To be proposed as standard of Society of Automotive Engineers (SAE)
- Created Colbert Object–Oriented Software Development method ("OOSD")
  - Noted for strength in real–time software development
    - NASA Langley Research Center used for software engineering process manual
- Founded Absolute Software Co., Inc. in 1986

Course Etiquette

- Observe normal rules of classroom etiquette
  - Be on time
    - If you are late, don't slam the door
      - (tell your friends)
  - One conversation at a time
  - One topic at a time
  - Direct all comments to instructor
  - Encourage (rather than criticize) other students
  - E-mail & browsing at breaks or after class
  - Turn off/silent cell phones & pagers

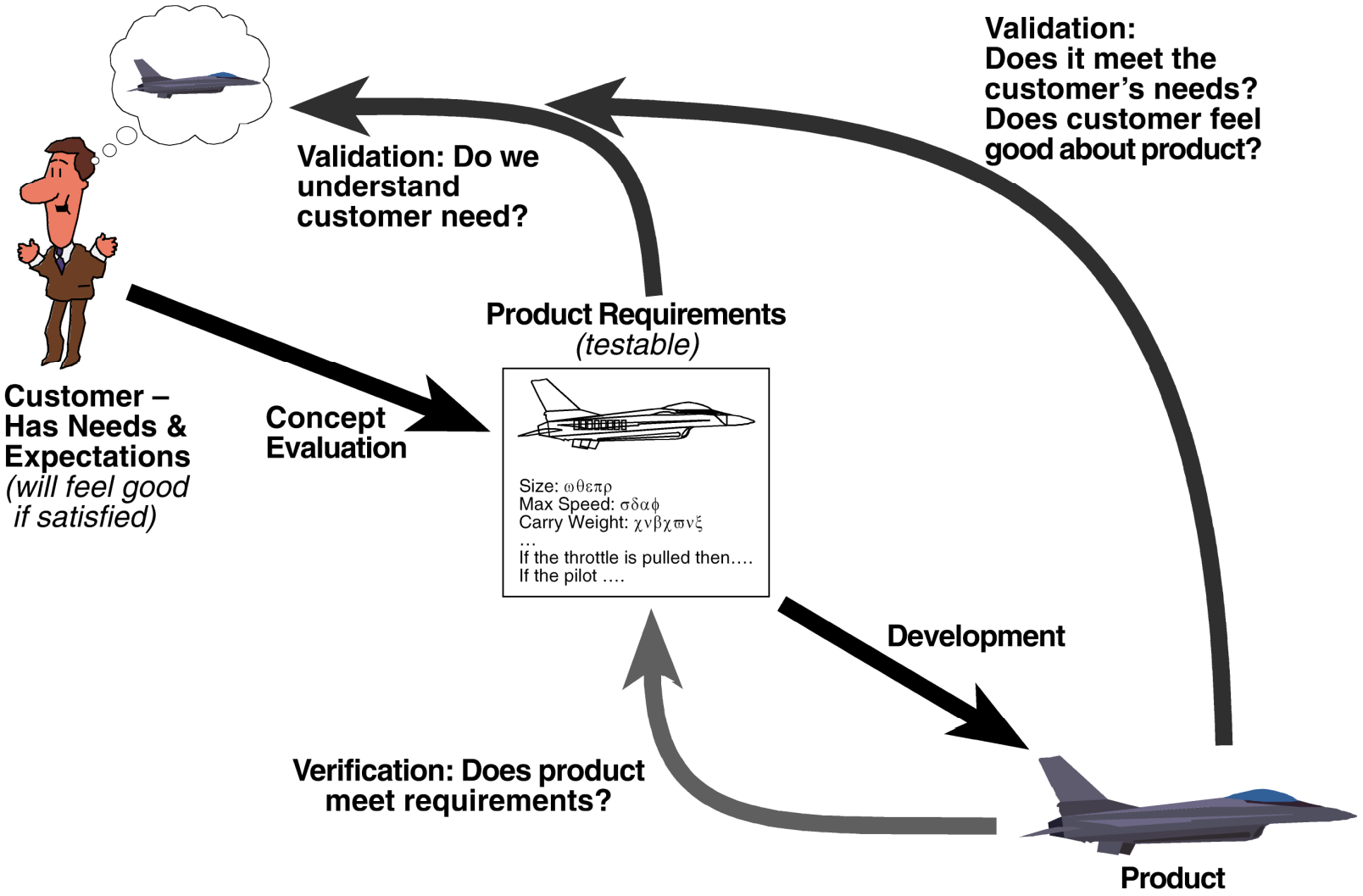
Course Etiquette (cont.)

- Respect others & yourself
- Cheat
  - –Don't!
- Be ready to learn….

Learning

- We each have a role
- Lou Holtz said
  - "Your talent determines what you can do.
  - Your motivation determines how much you are willing to do.
  - Your attitude determines how well you do it."
- Plutarch said
  - "A mind is not a vessel to fill; but a fire to light."

Ultimate Goal — Satisfied Customer



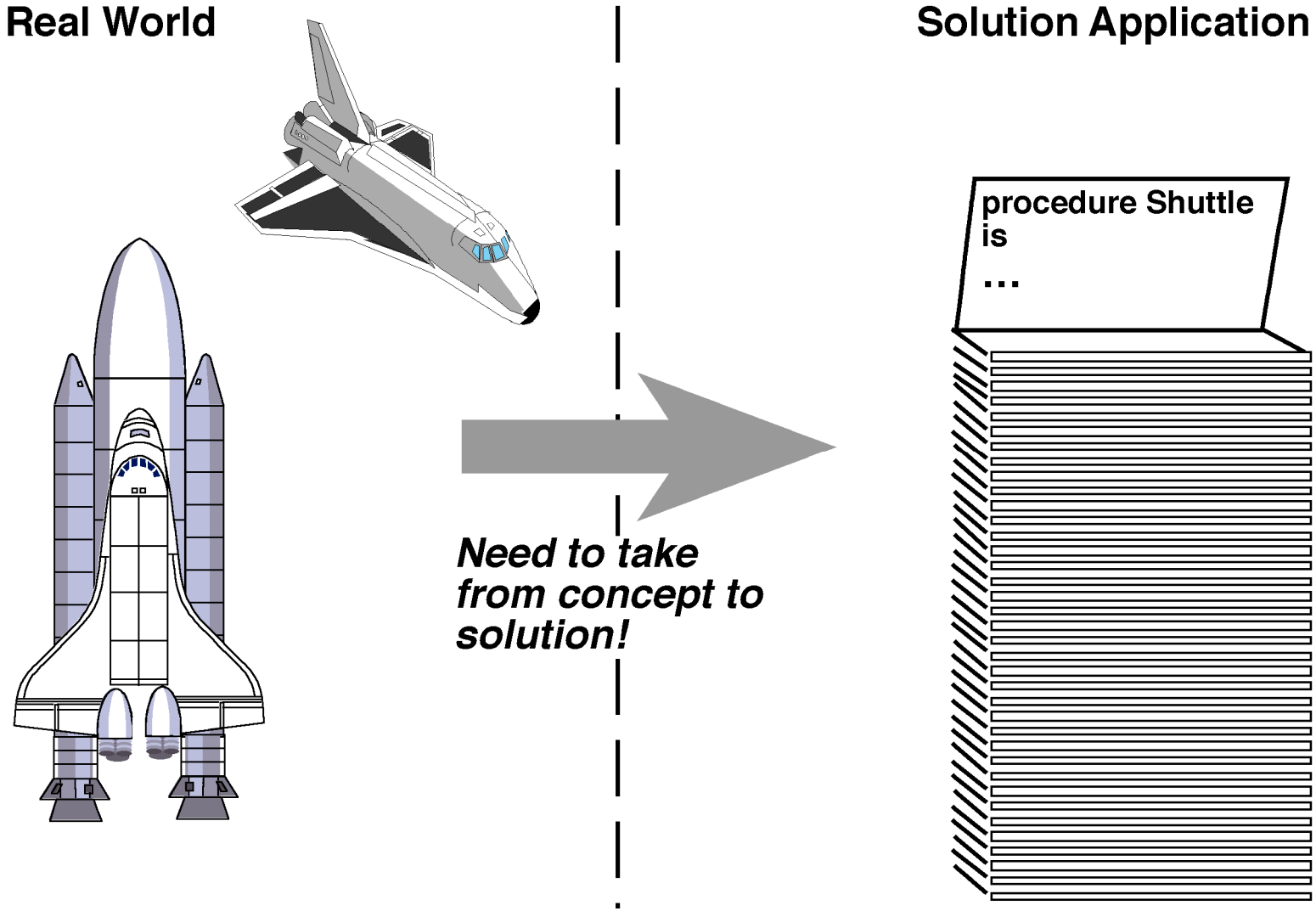
To Satisfy A Customer, We Must…

- Build a software product that satisfies a human need & meets:

Operational System Objectives*	Life–cycle Objectives
•**Reliability** •**Efficiency** •**Suitability**[1] • **Must be verifiable & testable** [1] Also called "appropriateness"	•**Understandability** •**Adaptability** •**Portability** •**Re–usability** •**Tunability** •**Plasticity** •**Maintainability**
Development Process's (Economic) Objective	
•**Cost–Effectiveness** •**Productivity** •**Return on investment**	

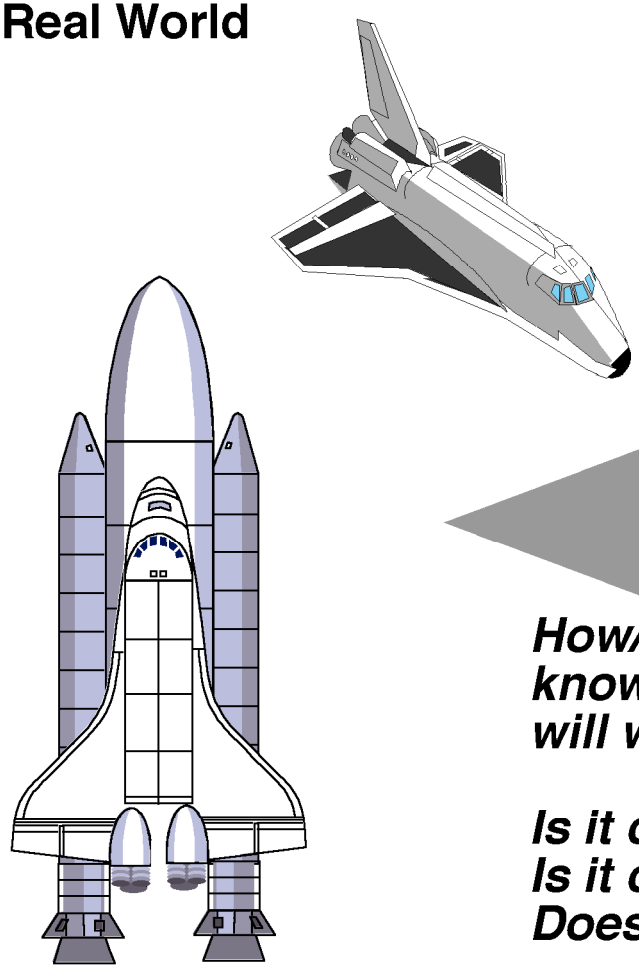
i.e. achieve "Software Engineering & System Building Goals"

Developer's Problem

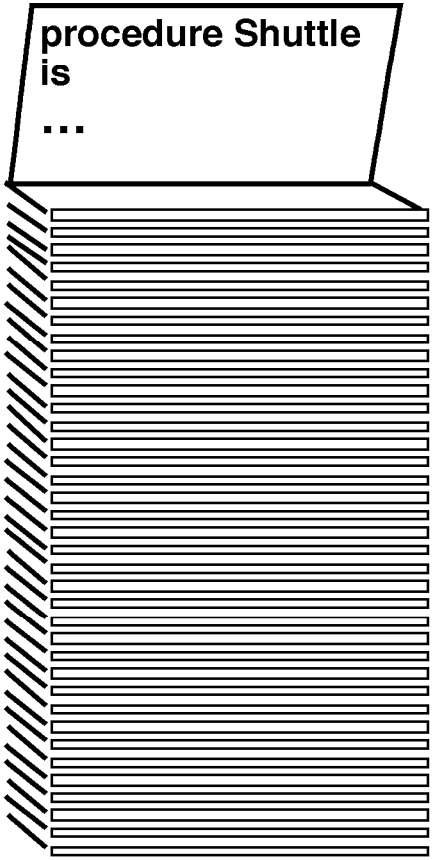


Developer's Problem (cont.)

Real World



Solution Application



How/When do we know that program will work?

Is it complete?
Is it correct?
Does it meet goals?

What is an Object?

- **An Object is**[1]**:**
  - Anything that is visible or tangible and is stable in form.

    e.g., A particular:

Airplane

City

Person

Planet

Song Group

Computer

What is an Object? (cont.)

- **An Object is[1]:**
  - Anything that may be apprehended intellectually: *objects of thought*.

    e.g., A particular:

Flight Software	Operating System	Database
Navigation System	Calendar	Schedule

What is an Object? (cont.)

- **An Object is[1]:**
  - A person or thing with reference to the impression made on the mind or the feeling or emotion elicited in an observer: *an object of pity*.
  - A thing, person, or matter to which thought or action is directed.

    e.g., A particular target

[1] Webster's Encyclopedic Unabridged Dictionary of the English Language, Portland House, 1989.

I have this box here that will solve your problem. What do you want to know about it?

What is an Object? (cont.)

- Object–Oriented Definition:
  - A self-contained thing which is characterized by:
    - Unique Identity
    - Structure
  - Responsibilities[1],
    - its state
    - its dynamic behavior
    - the operations it can perform
    - the requests it makes of other objects[2]
  - Qualities
    - Quantitative (e.g., size, speed)
    - Qualitative (e.g., “user friendly”, re-usable)

[1] Equivalent to the pre-OO concepts of functionality & dynamic behavior, or the concept capabilities.

[2] Each object collaborates (works) with other objects in performance of its responsibilities.

What is a Class?

❑ A Class is:

– A number of persons or things [objects] regarded as forming a group by reason of common attributes, characteristics, qualities, or traits.

Definition:

– A class is a description of a set of objects that share same [similar]: [1]
  - Attributes
  - Operations
  - Methods
  - Relations
  - Semantics

– A *Class* is a kind of *Classifier*

[1] The use of the term “set” in this definition does not imply that there will automatically be a “set” object in the executing program

- A class is like a blue print:
  - It describes what the instances will "look like" & how they'll behave
  - May defer some details of implementation
    - **e.g.** Describe the structure, responsibilities, & qualities of a car without specifying the class of upholstery used for the seats (or the class of seats) in the car
  - Instances can be created that specify details previously deferred