HUMAN COMPUTER INTERACTION

Lecture 5: Task Analysis

Hall of Fame or Shame?

Hall of Fame or Shame?

Today’s Topic

- User Analysis
- Task Analysis
- Domain Analysis
- Requirement Analysis

Know Your Users

- Identify characteristics of target user population
  - *Age, Gender, Culture, Language*
  - *Education (Literacy?, Numeracy?)*
  - *Physical limitation*
  - *Computer experience (typing?)*
  - *Motivation, attitude*
  - *Domain experience*
  - *Application Experience*
  - *Work Environment and social context*
  - *Relationships and communication pattern*

Multiple Classes of Users

- Many Applications have several kinds of users
  - *By role(Parent, Teacher)*
  - *By Characteristics(age, Motivation)*
- Example: Olympic Messaging System
  - *Athletes*
  - *Friends and Families*
  - *Telephone operators*
  - *System Admins*

Personas

- A Persona is a fiction character used as a specific representative of a class
  - *Yoshi is a 20 year old pole vaulter from Tokyo who speaks Some English.*
  - *Bob is an IBM Sysadmin in New York*
- Advantages
  - *Convenient handle for talking about user classes*
  - *Focuses of a typical user rather than an Extreme*
  - *Encourages Empathy*
- Disadvantages
  - *May be Misleading*
  - *Stereotype trap*

How to do User Analysis

- Techniques
  - *Questionnaires*
  - *Interviews*
  - *Observation*
- Obstacles
  - *Developers and users are sometimes systematically isolated from each other*
    - Tech support shields developers from users
    - Marketing shields users from developers
  - *Some user are expensive to talk to*
    - Doctors, executives, union member

Task Analysis

- Identify the individual tasks the program might solve
- Each task is Goal (What, not How)
- Often helps to start with overall goal of the system and then decompose it hierarchically into task.

Essential Parts of Task Analysis

- What needs to be done?
  - *Goal*
- What must be done to make it possible?
  - *Preconditions*
    - Tasks on which these tasks depend
    - Information that must be known to the user
- What steps are involved in the task?
  - *Subtasks*
  - *Subtasks may be decomposed recursively*

Example form OMS

- Goal
  - *Send message to other athlete*
- Preconditions
  - *Must know: my country code, my username, my password, other athlete's name*
- Subtasks
  - *Login*
  - *Identify receipt*
  - *Record Message*
  - *Hang up*

Other Questions to ask about a Task

- Where is the task Performed?
- What is the Environment Like?
- How often is the task performed?
- What are its time and resource constraints?
- How is the task learned?
- What can go wrong?
- Who else is involved in the task?

How to do a Task Analysis

- Interviews with User
- Direct Observation of the user performing task

Example: Elevator Task Analysis

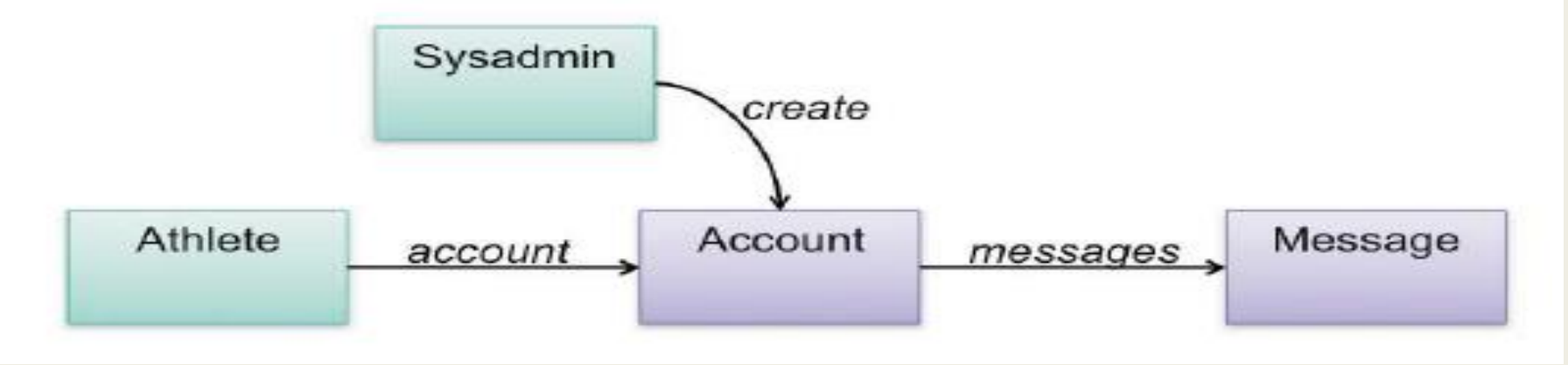
- Suppose we are designing a student center Elevator interface
- What are the Tasks?

Domain Analysis

- Identify important things in the domain
  - *People*
    - Athlete, Family and Friends, Sysadmins
  - *Physical Objects*
    - Namecard, telephone
  - *Information Objects*
    - Messages, Accounts

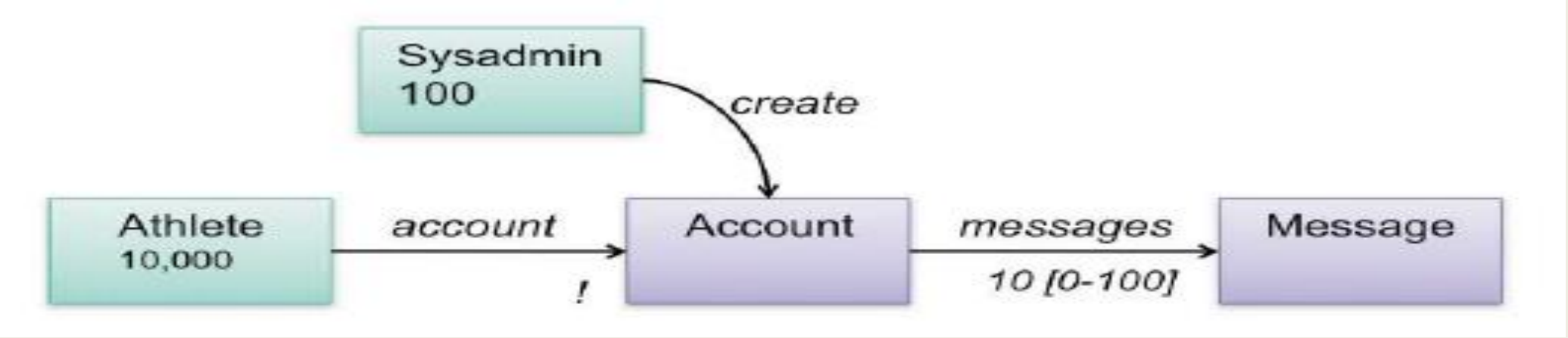
Domain Analysis

- Determine important relations in things
  - *Athletes have accounts*
  - *Accounts have messages*
  - *Family and Friends know athletes*
  - *Sysadmin register athlete or create accounts*



Domain Analysis

- Identify the multiplicities of things and relations
  - *Number are best, but simple multiplicity indicator(+, !, *, ?) help too*



Feedback to User and Task Analysis

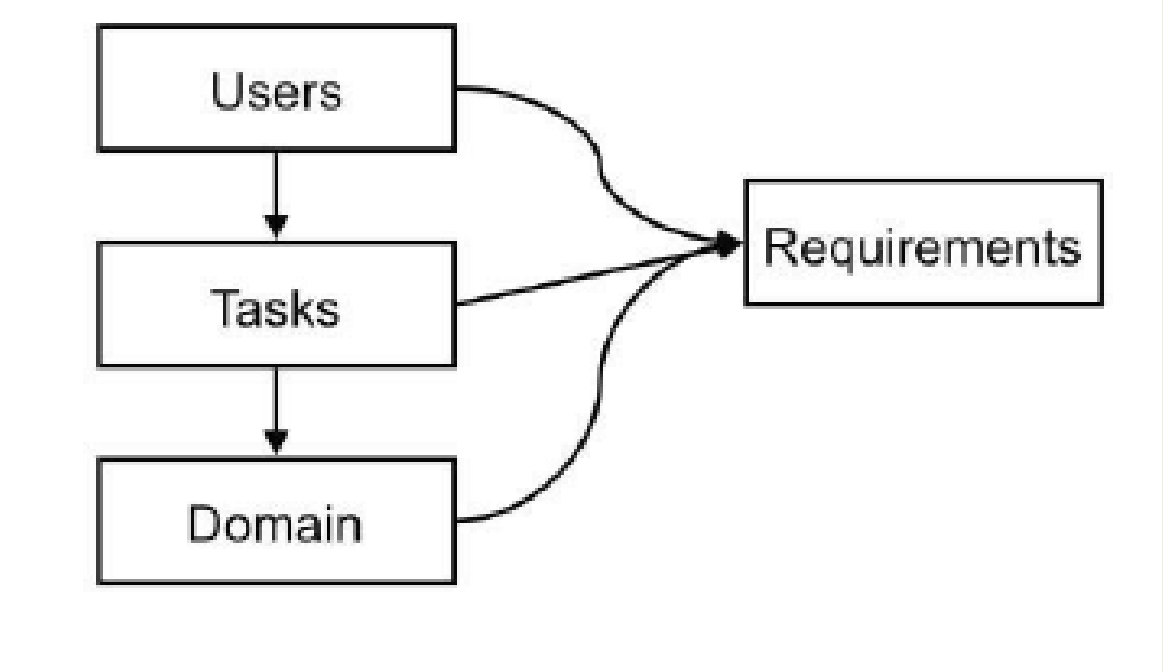
- People entities who really should be user classes
- Missing tasks
  - *CRUD*

Example: Twitter Domain Analysis

- Suppose we are reimplementing twitter
- What are its entities, relationships and multiplicities?

Requirement Analysis

- Requirement: what should the system do?



Common Error in user Analysis

- Describing what your ideal users should be, rather than what they actually are
  - *User should be literate in English, fluent in spoken Swahii, right-handed, and color blind*

Common Errors in Task Analysis

- Thinking for System point of view rather than user's
- Fixating too early on a UI design vision
- Bogging down in what user do know(**concrete** tasks) rather than why they do it (**essential** tasks)
- Duplicating a bad existing procedure in the software
- Failing to capture good aspects of existing procedure

Hints for the better user

- Questions to ask
  - *Why do you do this? (Goal)*
  - *How do you do it? (Subtasks)*
- Look for weakness in the current situation
  - *Goal Failure, User irritation, wasted time*
- Contextual inquiry
- Participatory design

Contextual Inquiry

- Observe user doing real work in the real world environment
- Be concrete
- Establish a master-apprentice relationship
  - *User shows how and talks about it*
  - *Interviewer watches and asks questions*
- Challenge assumptions and probe surprise

Participatory design

- Include representative user directly in the design team
- OMS design team included an Olympic athlete as a consultant