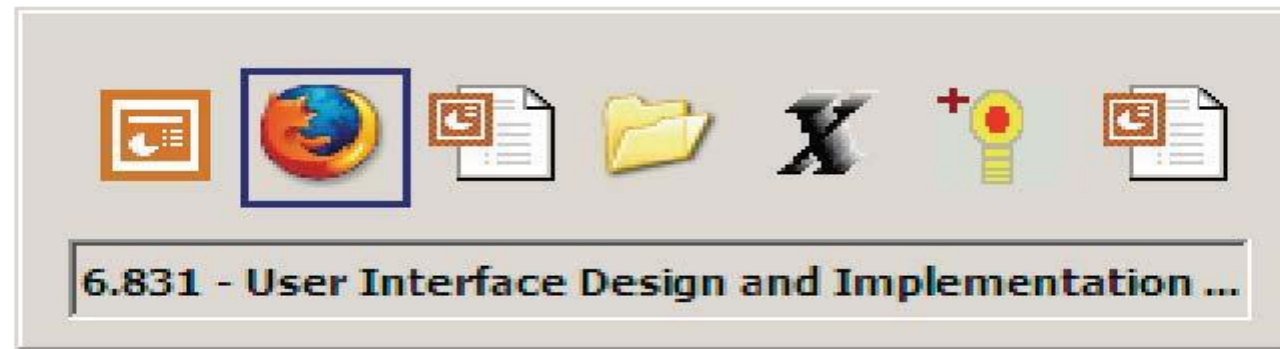
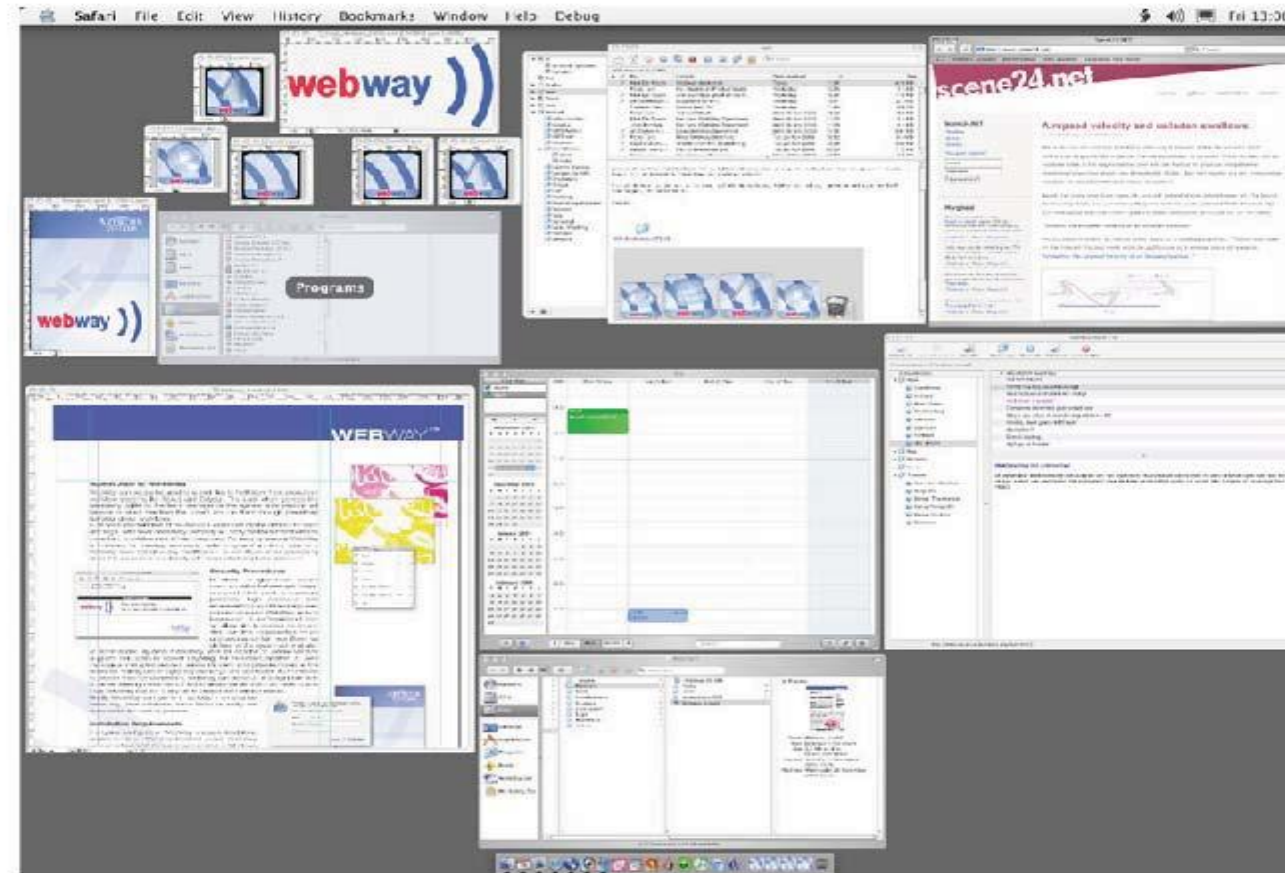


Task Analysis

UI Hall of Fame or Shame?



UI Hall of Fame or Shame?



Spring 2011

6.813/6.831 User Interface Design and Implementation

3

Today's Topics

- User analysis
- Task analysis
- Domain analysis
- Requirements analysis

Know Your User

- Identify characteristics of target user population
 - Age, gender, culture, language
 - Education (literacy? numeracy?)
 - Physical limitations
 - Computer experience (typing?)
 - Motivation, attitude
 - Domain experience
 - Application experience
 - Work environment and other social context
 - Relationships and communication patterns

Multiple Classes of Users

- Many applications have several kinds of users
 - By role (student, teacher)
 - By characteristics (age, motivation)
- Example: Olympic Message System
 - Athletes
 - Friends & family
 - Telephone operators
 - Sysadmins

Personas

- A persona is a fictitious character used as a specific representative of a user class
 - Yoshi is a 20-year-old pole vaulter from Tokyo who speaks some English
 - Bob is an IBM sysadmin in New York
 - Fritz is the 50-year-old father of a German swimmer
- Advantages
 - Convenient handle for talking about user classes
 - Focuses on a typical user, rather than an extreme
 - Encourages empathy
- Disadvantages
 - May be misleading
 - Stereotype trap

Example

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 users are expensive to talk to
 - Doctors, executives, union members

Task Analysis

- Identify the individual tasks the program might solve
- Each task is a goal (*what*, not *how*)
- Often helps to start with overall goal of the system and then decompose it hierarchically into tasks

Essential Parts of Task Analysis

- What needs to be done?
 - Goal
- What must be done first to make it possible?
 - Preconditions
 - Tasks on which this task depends
 - Information that must be known to the user
- What steps are involved in doing the task?
 - Subtasks
 - Subtasks may be decomposed recursively

Example from OMS

- Goal
 - Send message to another athlete
- Preconditions
 - Must know: my country code, my username, my password, the other athlete's name
- Subtasks
 - Log in (identify yourself)
 - Identify recipient
 - Record message
 - Hang up

Other Questions to Ask About a Task

- Where is the task performed?
 - At a kiosk, standing up
- What is the environment like? Noisy, dirty, dangerous?
 - Outside
- How often is the task performed?
 - Perhaps a couple times a day
- What are its time or resource constraints?
 - A minute or two (might be pressed for time!)
- How is the task learned?
 - By trying it
 - By watching others
 - Classroom training? (probably not)
- What can go wrong? (Exceptions, errors, emergencies)
 - Enter wrong country code
 - Enter wrong user name
 - Get distracted while recording message
- Who else is involved in the task?

How to Do a Task Analysis

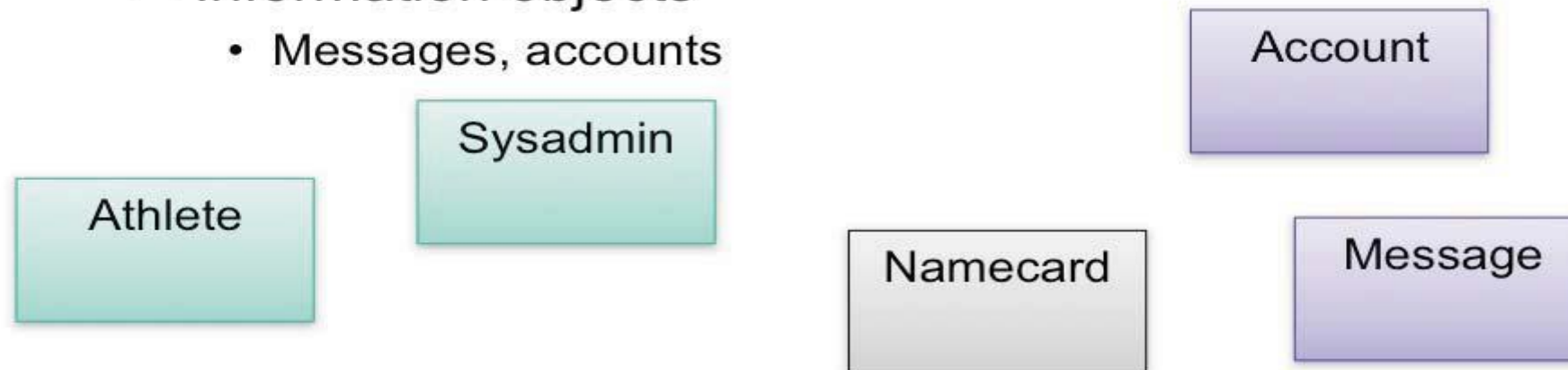
- Interviews with users
- Direct observation of users performing tasks

Example: Elevator Task Analysis

- Suppose we're designing the Student Center elevator interface
- What are the tasks?

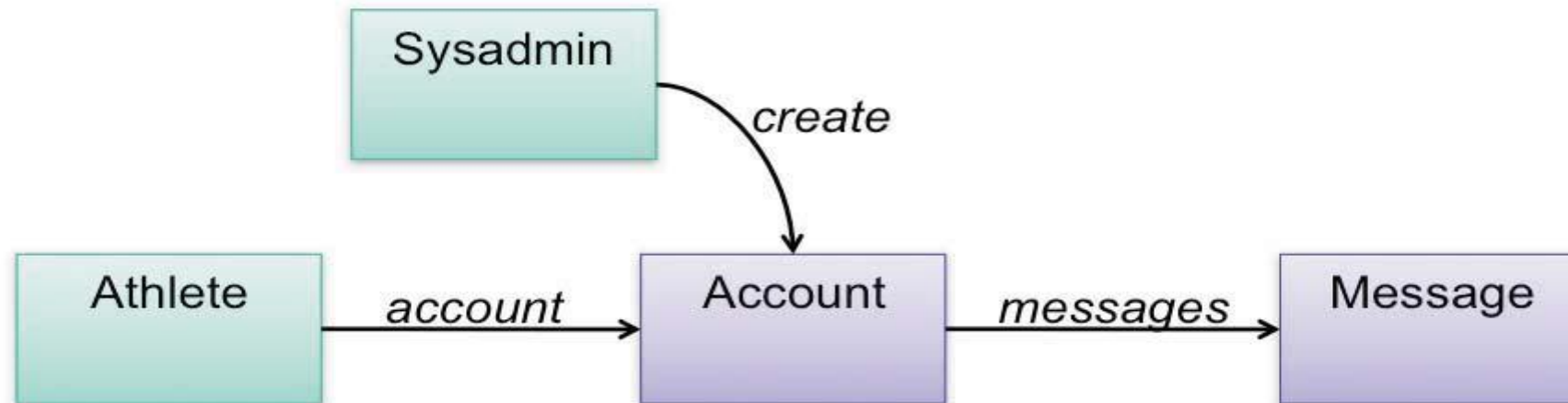
Domain Analysis

- Identify important things in the domain
 - People (user classes)
 - Athletes, friends & family, sysadmins
 - Physical objects
 - Namecard, telephone
 - Information objects
 - Messages, accounts



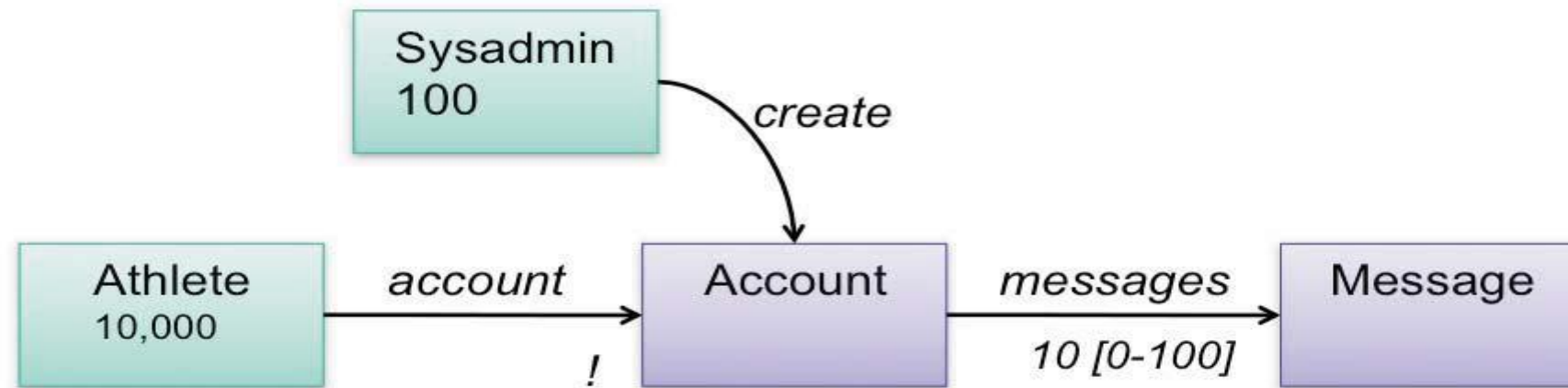
Domain Analysis

- Determine important relations between the things
 - Athletes *have* accounts
 - Accounts *have* messages
 - Family & friends *know* athletes
 - Sysadmins *register* athletes or *create* accounts



Domain Analysis

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



Feedback to User & Task Analysis

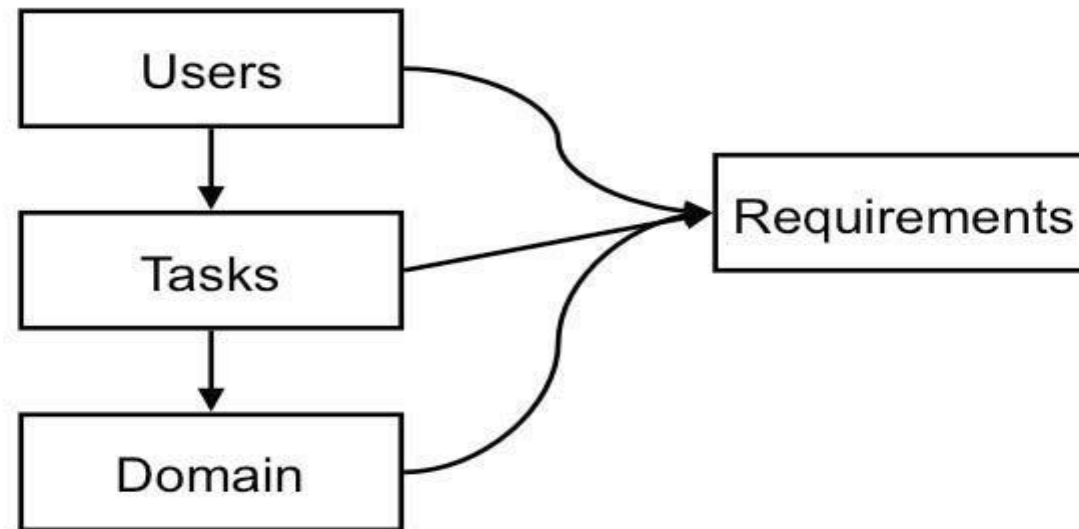
- People entities who really should be user classes
- Missing tasks
 - CRUD: Create, Read, Update, Delete

Example: Twitter Domain Analysis

- Suppose we're reimplementing Twitter.
- What are its entities, relationships, and multiplicities?

Requirements Analysis

- Requirements: what should the system do?



Common Errors in User Analysis

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

Common Errors in Task Analysis

- Thinking from the system's point of view, rather than the user's
 - "Notify user about appointment"
 - vs. "Get a notification about appointment"
- Fixating too early on a UI design vision
 - "The system bell will ring to notify the user about an appointment..."
- Bogging down in *what* users do now (**concrete** tasks), rather than *why* they do it (**essential** tasks)
 - "Save file to disk"
 - vs. "Make sure my work is kept"
- Duplicating a bad existing procedure in software
- Failing to capture good aspects of existing procedure

Hints for Better User & Task Analysis

- Questions to ask
 - Why do you do this? (goal)
 - How do you do it? (subtasks)
- Look for weaknesses in current situation
 - Goal failures, wasted time, user irritation
- Contextual inquiry
- Participatory design

Contextual Inquiry

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

Participatory Design

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

Summary

- User analysis identifies the user classes
- Task analysis discovers their tasks
- Domain analysis finds the entities and relationships in the domain