CPE/CSC 486: Human-Computer Interaction

Franz J. Kurfess

Computer Science Department
California Polytechnic State University
San Luis Obispo, CA, U.S.A.

Course Overview

- Introduction
- Interacting with Devices
- Interaction Styles
- UI Elements
- UI Design Guidelines
- UI Development Tools
- User Assistance
- Interaction through Speech
- Interaction with Mobile Devices
- Project Presentations

Chapter Overview Introduction

- Logistics
- Motivation
- Objectives
- Relevance of HCI
- Technologies and Ideas
- Evaluation Methods
- Rapid Prototyping
- User-Centered Design
- Important Concepts and Terms
- Chapter Summary

[Dix, Preece, Mustillo]

© Franz J. Kurfess

Motivation

- **utilization of computers to deal with knowledge**
  - quantity of knowledge available increases rapidly
  - relieve humans from tedious tasks
- **computers have special requirements for dealing with knowledge**
  - acquisition, representation, reasoning
- **some knowledge-related tasks can be solved better by computers than by humans**
  - cheaper, faster, easily accessible, reliable

Objectives

- to know and comprehend the main principles, components, and application areas for Knowledge-Based Systems
- to understand the structure of Knowledge-Based Systems
  - knowledge base, inference engine
- to be familiar with frequently used methods for knowledge representation in computers
- to evaluate the suitability of computers for specific tasks
  - application of methods to scenarios or tasks

© Franz J. Kurfess

Logistics

- Introductions
- Course Materials
- Term Project
- Homework
- Exams
- Grading

© Franz J. Kurfess

Instructor

- **Dr. Franz Kurfess**
- **Professor, CSc Dept.**
- **Areas of Interest**
  - Artificial Intelligence
    - Knowlege Mangement
    - Neural Networks & Structured Knowledge
  - User-Centered Design
  - Computer Support for Learning and Teaching
- **Contact**
  - preferably via email: fkurfess@calpoly.edu
  - phone (805) 756-7179
  - office 14-218
  - office hours Tue/Thu 2:10-3:00, Wed 2:10-5:00

© Franz J. Kurfess

Course Material

- **on the web http://www.csc.calpoly.edu/~fkurfess**
  - syllabus
  - schedule
  - project information
  - lecture at http://users.csc.calpoly.edu/~fkurfess/Courses/486/S12/Slides/
    - Keynote (original), PowerPoint, PDF
- **on PolyLearn (soon)**
  - grades
- **Semantic MediaWiki at http://kurfess.wikia.com**
  - project topics and discussion
- **TRAC Wiki**
  - project documents
  - individual student and project materials

© Franz J. Kurfess

Assignments

- two homework assignments
  - heuristic usability evaluation
  - data collection
- similar to two assignments in CSC 484
- preferably aligned with the project topic

Research Activity

- **teams will investigate a topic of their choice**
  - related to Human-Computer Interaction
  - ideally aligned with the project topic
- **flexible format**
  - conventional paper, Wikipedia article (or similar), video, blog entries
- **expectations**
  - Deliverable: There must be a concrete outcome to the research activity.
  - Educational Value: The outcome should be beneficial for others, both within this class as well as outside of the class.
  - High-Quality Presentation: The deliverable must be comparable in the quality of the presentation to a publication in conference proceedings or a journal..
  - Public Availability
- **length**
  - paper, Wikipedia article, blog posts about 3,000 words per person (5-10 pages)
    - follow ACM Crossroads formatting guidelines at http://www.acm.org/crossroads/submit/
  - see also the CfP (long past) for a special issue on HCI at http://www.acm.org/crossroads/doc/cfas/hci.html and the accepted papers at http://www.acm.org/crossroads/xrds12-2/
  - video: to be decided
- **peer reviews**
  - outcomes of the activity will be reviewed by somebody else
  - details to be discussed in class

Term Project

- **two options**
  - development of a computer-based system
    - prototype, emphasis on user interaction
  - exploration of a novel development in HCI
    - design/development aspect
    - experiments
- **peer evaluation**
  - teams evaluate the system of another team
- **information exchange via TRAC Wikis**

Project Themes:

- **interaction aspects for computer-based systems that do not rely on the traditional arrangement of screen, keyboard, and mouse or trackpad**
  - user interaction through touch, gestures, voice, or other methods.
- **constraints imposed by**
  - device size (as in mobile devices),
  - purpose (entertainment and gaming devices),
  - environment (hands-free operation, background noise),
  - other limiting factors.

Exams

- no exams, unless the majority of students in class wants exams

Class Participation

- **will contribute 10% to the overall grade**
- **factors**
  - attendance
  - speaking up in class
  - contributions to discussions
    - in class
    - on-line (e.g. Blackboard)
- **evaluation criteria**
  - similar to the ones used for presentations
- **self-assessment at the end**
  - similar to 484

Grading Policy

- Assignments: 20%
- Research Activity: 30%
- Project: 40%
- Class Participation: 10%

Human-Computer Interaction

- **addresses any interaction by humans with computer systems:**
  - as users
  - as developers
  - as individuals
  - as groups
- **also referred to as User Interface Design, or Human-Computer Interface Design**

[Mustillo]

Human-Computer Interaction (cont)

- **concerned with the process of design**
  - not only the what, but also the how & the why of interface design
- **part of the larger discipline of Human Factors or Human Factors Engineering**
  - known as Ergonomics in Europe
  - looks at how users:
    - perform activities, tasks, jobs
    - interact with systems
    - use tools, machines, computers, software

[Mustillo]

Relevance of HCI

- **the goal is to develop and improve systems so that users can carry out their tasks:**
  - effectively
  - efficiently
  - enjoyably
  - safely
    - especially in safety-critical systems like air traffic control
- **these aspects are also known collectively as Usability**

[Mustillo]

Examples

- Three-Mile Island nuclear accident
- Helios Airways Flight 522

http://upload.wikimedia.org/wikipedia/commons/thumb/2/2e/Three_Mile_Island_nuclear_power_plant.jpg/800px-Three_Mile_Island_nuclear_power_plant.jpg

http://upload.wikimedia.org/wikipedia/commons/thumb/4/49/Helios_737_olympia.jpg/220px-Helios_737_olympia.jpg

[Mustillo]

Example 1: Three-Mile Island Nuclear Disaster

- **Trigger: Stuck valve**
- **Consequence: Emergency shutdown**
- **Human Factors**
  - Conflicting feedback messages
    - control light seemed to indicate a closed valve
      - the actual valve was open, but the solenoid sensor malfunctioned
    - prevented technicians from understanding what was happening and reacting quickly in the appropriate manner.
  - Improperly located displays/controls
    - obscured key components from one another

http://en.wikipedia.org/wiki/Three_Mile_Island_accident

© Franz J. Kurfess

Example 2: Airplane Accident Helios in Athens, Greece

  - see http://en.wikipedia.org/wiki/Helios_Airways_Flight_522
- **the same alarm signal was used for different purposes**
  - cabin altitude warning horn
  - take-off configuration warning
    - can only sound on the ground
- **technicians and pilots did not realize that a lever crucial for oxygen delivery was in the wrong position**
  - manual instead of auto

Approaches to HCID

- **systems engineering approach**
  - engineering model, bottom-up, reductionistic
- **user-centric approach**
  - (user-task model, top-down, holistic)
  - Example: water faucet design
    - conventional faucet design, separate hot/cold taps
    - single-handle faucet design, integrated flow rate and temperature control mechanism
  - Example: programmable remote control device
    - hexadecimal representation
      - 16 pages of storage, 16 command sequences, 0-F
    - task oriented, common task terms
      - VOLUME, CHANNEL, POWER, STOP

[Mustillo]

Scope of HCID

- **primary goal is to design usable systems**
- **requires knowledge about:**
  - Who will use the system - the user
    - motivation, satisfaction, experience level, etc.
  - What will it be used for - the tasks
    - office, information retrieval, transaction-based, etc.
  - Work context and environment in which it will be used
    - job content, power and influence, personnel policies, etc.
  - What is technically and logistically feasible
    - technological capabilities, memory size, costs, time scales, budgets, etc.

[Mustillo]

HCID Principles

- understand the user and the application
- ensure self-evident feature operation
- use users’ knowledge across systems
- don’t slow down the user
  - work with the user, not against him/her
- provide simple ways to deal with user errors

[Mustillo]

What is a “User Interface”?

- refers to the methods and devices that are used to make the interaction between machines and the humans who use them (users) possible
- UIs can take many forms, but always accomplish two fundamental tasks:
  - communicating information from the machine to the user
  - communicating information from the user to the machine

[Mustillo]

What is a “Good” User Interface?

- **the UI should represent the capabilities of the entire system**
- **the more complex the system, the more important is the UI**
- **UI should help the user build a “mental model”**
  - intuitive understanding of how the system works
- **when a system feels natural to use, the UI is doing a good job**
- **helps tailor the system to the user (adaptive)**
- **helps users absorb information**
- **meets the principal design goals**
  - e.g., learnability vs. usability, first-time use, infrequent use, or expert use

[Mustillo]

© Franz J. Kurfess

Seven Deadly Sins of User Interface Design

- **1. Design for technology rather than the user**
  - technology is not the panacea
- **2. “Coolness”**
  - flashy graphics do not improve a bad UI
- **3. Logical vs. visual thinking**
  - users don’t think like software designers
- **4. User input as right or wrong**
  - design for error
- **5. Overextend basics**
  - make simple things simple, complex things possible
- **6. Fix it with documentation**
  - users don’t read documentation; don’t try to fix a UI defect through documentation
- **7. Fix it in the next release**
  - old habits are hard to break

(Adapted from Trower, 1994)

[Mustillo]

Confession Time ...

- Which of the seven deadly sins have you committed?
- What were the
  - causes
  - consequences
  - repair (attempts)
- Write down two examples
- Sharing in class is optional

© Franz J. Kurfess

History of HCID

- arranged roughly into decades
- user interface design and related issues
- experimental and commercial systems

© Franz J. Kurfess

Your First Computer Memories

- What was the user interface of your first computer?
- In retrospect, what did you
  - like about it
  - dislike about it
- Jot down the year you started using a computer, and what type it was.
- Compare the year with your neighbor; the one with the “older” year wins!

© Franz J. Kurfess

1940s

- **increasing complexity of aircraft fighter cockpits and increasing no. of "pilot error" accidents during W.W.II**
- **coining of the term "man-machine interface"**
- **introduction of the first modern electronic computers**
- **ENIAC (1943)**
  - the world's first all electronic numerical integrator and computer
- **Mark 1 (1944)**
  - the world's first paper tape reader
- **publication of Vannevar Bush's "As we may think" (1945) article in Atlantic Monthly**

[Mustillo]

1950s

- introduction of assembly language
- use of transistors for computers

[Mustillo]

1960s

- **invention of the mouse (1963)**
  - Douglas Engelbart at Stanford University
- **Ivan Sutherland’s SketchPad (1963 Ph.D. Thesis)**
  - introduced many new ideas/concepts now found in today’s interfaces (e.g., hierarchical menus, icons, copying, input techniques (light pen), etc.)
- **data tablet (1964) as an input device**
- **multiple tiled windows**
  - Engelbart 1968
- **idea of overlapping windows**
  - proposed by Alan Kay in 1969 Ph.D. dissertation
- **Dynabook (1969) by Alan Kay**
  - the first prototype of a notebook computer

[Mustillo]

1970s

- emergence of the first personal computers
  - Altair, Apple
- start of migration to the desktop

[Mustillo]

1980s

- **first Graphical User Interface (GUI) developed at Xerox PARC**
  - familiar user conceptual model (simulated desktop)
    - Introduction of the “desktop” metaphor
  - promotes recognition/pointing rather than remembering
    - What You See Is What You Get” (WYSIWYG)
- **Xerox Star (1981)**
  - first commercial PC designed for “business professionals”
  - design of the Star UI incorporated human factors as a major method of design

[Mustillo]

1980s (cont.)

- CHI conference (1982) draws 2000 - 3000 people
- Apple Lisa (1983)
  - successor to the Xerox Star, predecessor of the Macintosh
  - overlapping windows
  - a commercial failure
- X Window System developed by MIT in 1984
- Apple Macintosh (1985)

1980s (cont.)

- emergence of new interface technologies
- emergence of User Interface Management Systems (UIMS), toolkits, & interface builders
  - separation of the Interface from the application functions
- emphasis on user-centered design
  - mostly preaching
- battle between the Mac & Windows

[Mustillo]

1990s

- **MS Windows becomes desktop king**
  - but there’s room for improvement
- **growing importance and acceptance of user-centered design philosophy in industry**
- **growing importance of object-oriented technologies**
- **emergence of other interface modalities**
  - e.g., speech, pen
  - technological innovations
  - new metaphors
- **emergence of intelligent agents**
  - starting to become commodity technology

[Mustillo]

© Franz J. Kurfess

1990s (cont.)

- **tremendous shift in the perception of UI design**
  - from a mere afterthought to a critical aspect of an application
  - treatment of users
    - from treating users as a monolithic, homogeneous group, differentiated primarily by discipline or task
    - to recognizing that users are unique

[Mustillo]

2000s

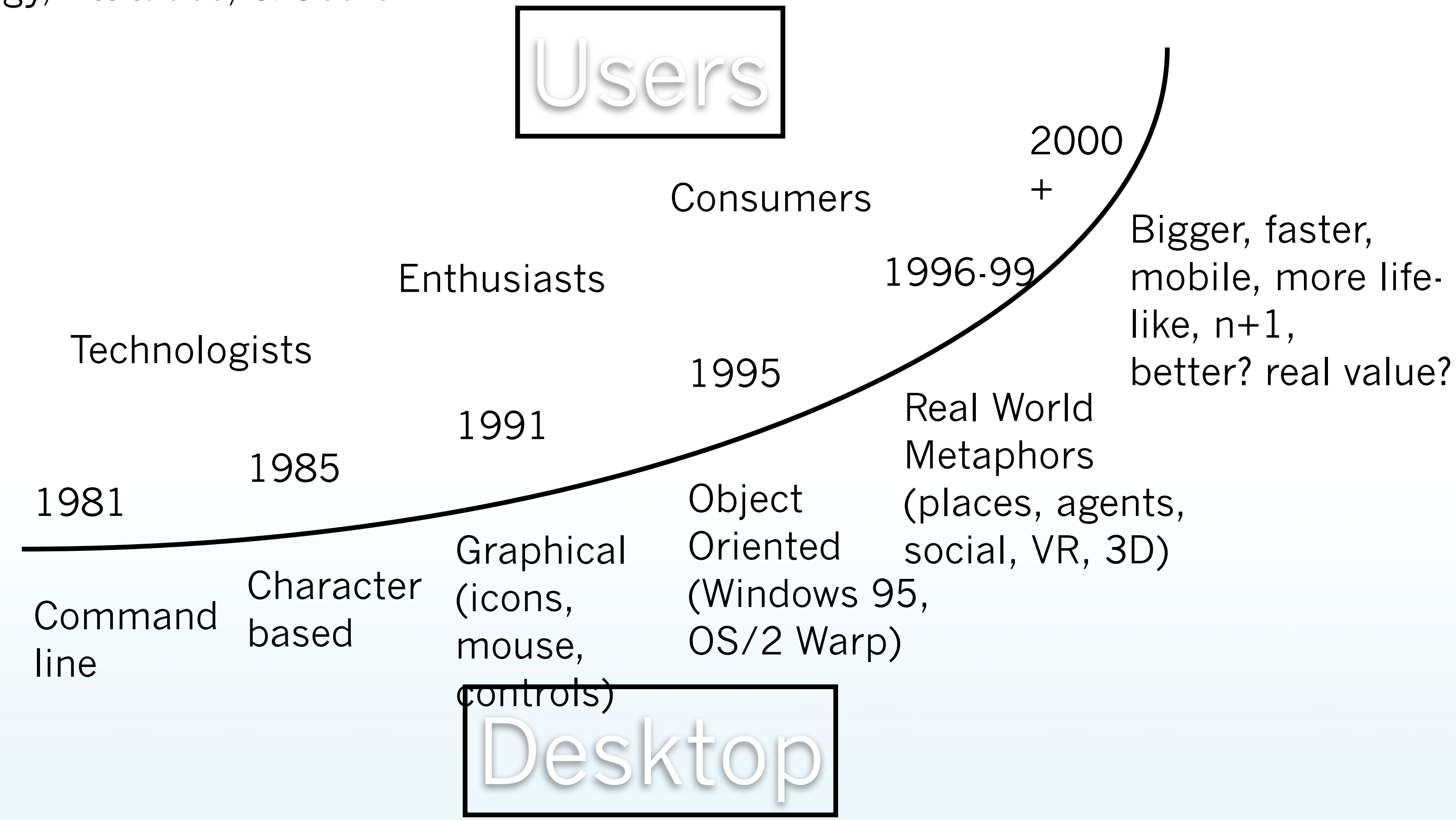
- **computers become more ubiquitous**
  - used in many aspects of our professional and personal lives
  - incorporated in many products
  - less exposed
- **convergence computers – personal devices**
  - smartphones, GPS systems, entertainment systems
- **Web-based interaction**
  - cloud computing
- **alternative interaction methods and devices**
  - touch-based, speech-based

2010s

- **touch-based interactions widely used**
  - smart phones, tablets
- **speech-based interaction**
  - beyond commands and dictation
- **“natural” user interfaces**
  - gesture-based
  - full body
    - not just hands
- **emotional interfaces**
  - emotion recognition in humans
  - expression of emotion in computer-based systems (agents)

HCID Evolution

Technology, Attitudes, & Users



Users

Technologists

Enthusiasts

Consumers

1981

Command line

1985

Character based

1991

Graphical (icons, mouse, controls)

1995

Object Oriented (Windows 95, OS/2 Warp)

1996-99

Real World Metaphors (places, agents, social, VR, 3D)

2000 +

Bigger, faster, mobile, more life-like, n+1, better? real value?

Desktop

[Mustillo]

HCI Evaluation

- **literature reviews**
- **user needs assessment**
  - determine what users need, are able to do, ...
- **use case scenarios**
  - form of requirements analysis
  - used to analyze, specify, define the system to be built
  - specifies functionality from a user's perspective

[Mustillo]

HCI Evaluation (cont.)

- **contextual task analysis**
  - observation/monitoring
    - non-invasive approach is better
  - interviews
    - exploratory, few users, subjective, structured or unstructured
  - surveys & questionnaires
    - feedback, many users, broad sampling, highly structured
- **simulations/prototyping/demos**
  - check feasibility, explore new ideas, evaluate alternatives
  - actual working systems
    - or systems with simulated functionality
      - e.g., Wizard of Oz

[Mustillo]

HCI Evaluation (cont.)

- **retrospective analysis**
  - user reviews own performance on a task, and provides comments
  - provides additional insight into user's mental models
- **comparative analysis**
  - users do the same task on multiple similar UIs or products
    - Find out which one is best
- **competitive analysis**
  - users test competitors' products, applications, & services
- **participatory design**
  - users participate in the design of the user interface

[Mustillo]

HCI Evaluation (cont.)

- **usability studies**
  - determine where users make errors, how often they make errors, can they use the system, number of requests for help, task completion times, etc.
  - viewing what they do (visual), listening to what they say (auditory)
- **heuristic evaluation**
  - use of experts and non-experts to find high-level usability problems early in the design phase
  - often based on guidelines
    - Nielsen’s heuristics

[Mustillo]

HCI Evaluation (cont.)

- **lab studies**
- **field studies**
  - trials, site visits, on-premise structured observation, testing, and use of other data gathering techniques
- **focus groups**
  - moderated session with few users, focused exploration and feedback

[Mustillo]

Evaluation Tools & Methods

- **Wizard of Oz**
  - Evaluate functions or features before developing anything, by having a human playing the role of the computer
- **rapid iterative developmental testing**
  - feedback from tests of small numbers of representative users is used to suggest modifications and improvements that can be made to early design prototypes
- **failure analysis**
  - find out where users go wrong, make mistakes, or are unsatisfied with some aspects of the design, system, etc

[Mustillo]

Evaluation Tools & Methods (cont.)

- **individual differences analysis**
  - determine characteristics of users who find various systems or features easy or hard to learn to use (via questionnaires, observations, or testing)
- **time profile analysis**
  - Formalization of places and things in a user interface design where users devote the most time may reveal areas of improvement

[Mustillo]

User-Centered Design (UCD)

- **approach that focuses on users and on activities that meet users' needs**
- **embodies four key concepts:**
  - early focus on the user
  - integrated design
  - early and continual user testing
  - iterative design

[Mustillo]

Principles of UCD

- **objective of UCD is to match whatever is being designed/developed to the characteristics**
- **not much of a problem for one or two users**
  - problem complexity increases when there are many users
- **user differences will always exist**
  - but design for the greatest commonalties
- **focuses not on technology, but on the user**
  - cognitive abilities
  - limitations
  - cultural, professional, or personal preferences

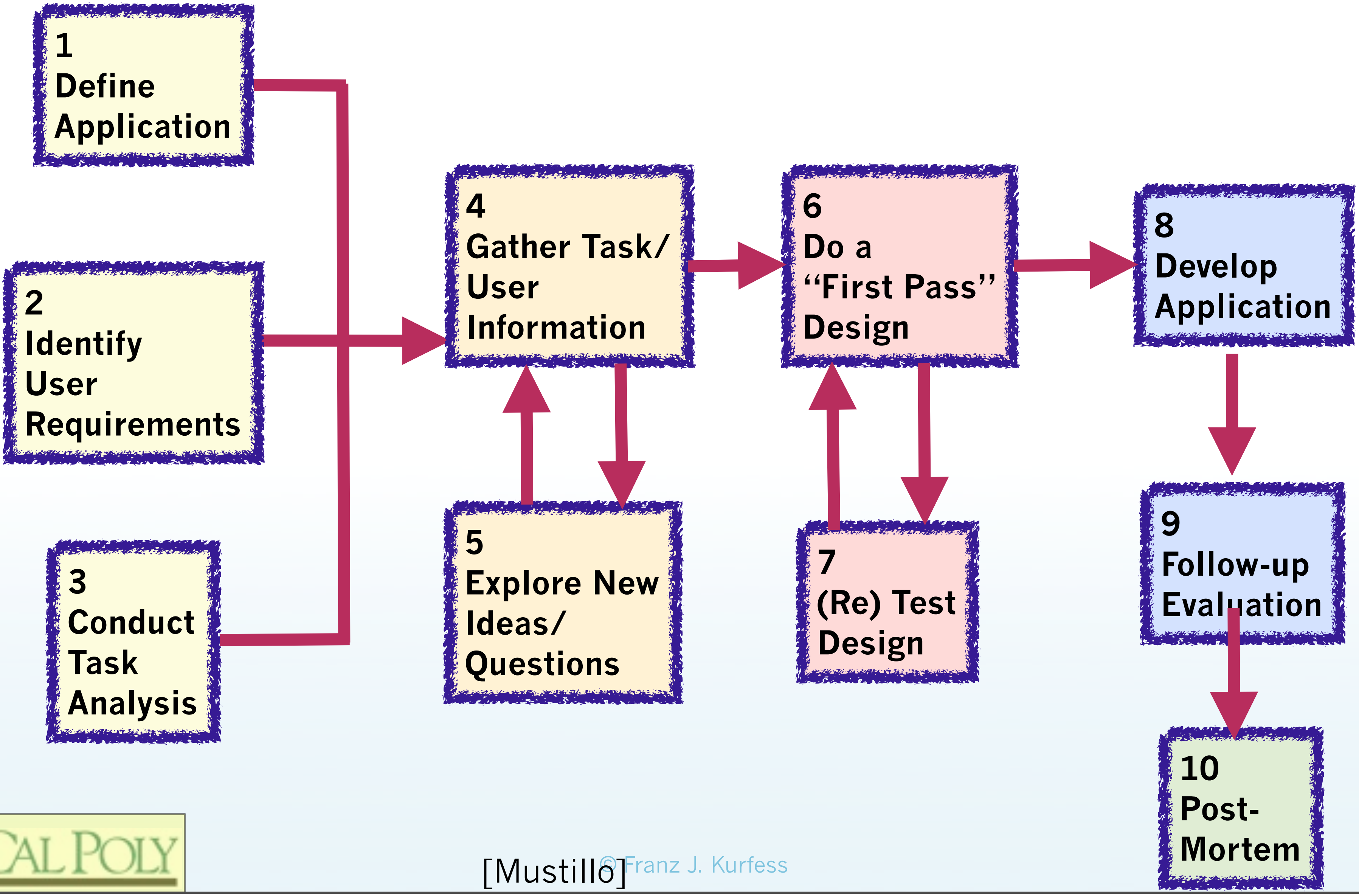
[Mustillo]

Principles of UCD

- identifies information needed from users
- provides explicit phases for collecting and interpreting data from users
- provides criteria for triggering moves back and forth between phases

[Mustillo]

Typical UCD Cycle



1 Define Application

2 Identify User Requirements

3 Conduct Task Analysis

4 Gather Task/ User Information

5 Explore New Ideas/ Questions

6 Do a "First Pass" Design

7 (Re) Test Design

8 Develop Application

9 Follow-up Evaluation

10 Post-Mortem

[Mustillo]

UCD Phases

- **Define the Application**
  - Scope out the problem, and clearly lay down ground rules.
    - What is the application?
    - Who are the intended users?
    - How and where will the application be used?
- **Identify User Requirements**
  - Know your users, and know them well.
  - Designers and developers are not users.
  - Managers & vice-presidents do not represent real users.

[Mustillo]

UCD Phases (cont.)

- **Conduct a Task Analysis**
  - Context is important.
    - What types of tasks do users typically use in order to do their jobs?
    - What cognitive, perceptual, or motor-task demands are normally imposed on users?

[Mustillo]

UCD Phases (cont.)

- **Gather Existing Information on Users and Tasks**
  - Gather information; investigate where information is lacking.
    - What are the users' preferences for different interfaces?
    - What are the users' preferences for different features?
    - What factors affect usability measures (e.g., performance, satisfaction) in different interfaces?

[Mustillo]

UCD Phases (cont.)

- **Explore New Ideas and Questions**
  - Don’t be afraid to ask hard questions.
- **Do a “First Pass” Design**
  - Prototype early.
  - Design to clear and objectively defined usability goals.
- **(Re-) Test the Design**
  - Test repeatedly and iterate the design until usability goals are met.

[Mustillo]

UCD Phases (cont.)

- **Develop the Application**
  - By now, you should be fairly confident that you are developing the right application.
- **Follow-up Evaluation**
  - Observe and evaluate the effectiveness of the user interface in the real world of real users.
  - Conduct field studies.
- **Post-Mortem**
  - Presume that there is a better way, and set out to find it.

[Mustillo]

Important Concepts and Terms

- contextual task analysis
- desktop
- ergonomics
- Evaluation Methods
- focus groups
- graphical user interface (GUI)
- heuristic evaluation
- human factors engineering
- human-machine interface
- input/output devices
- knowledge management
- mouse
- participatory design
- pervasive computing
- Rapid Prototyping
- simulation
- systems engineering
- task analysis
- ubiquitous computing
- usability
- use case scenarios
- User-Centered Design
- user interface design
- user requirements
- What You See Is What You Get" (WYSIWYG)
- window

Chapter Summary

- introduction to important concepts and terms
- relevance of HCID
- historical development of HCID
- emphasis on the user
  - user-centered design

© Franz J. Kurfess