# Mensch-Maschine-Interaktion 1

Chapter 8 (July 15, 2010, 9am-12pm):

User-Centered Development Process

# User-Centered Development Process

- **Software Development Process Models**

- User-Centered Development

- Integrating Usability into the Development Process
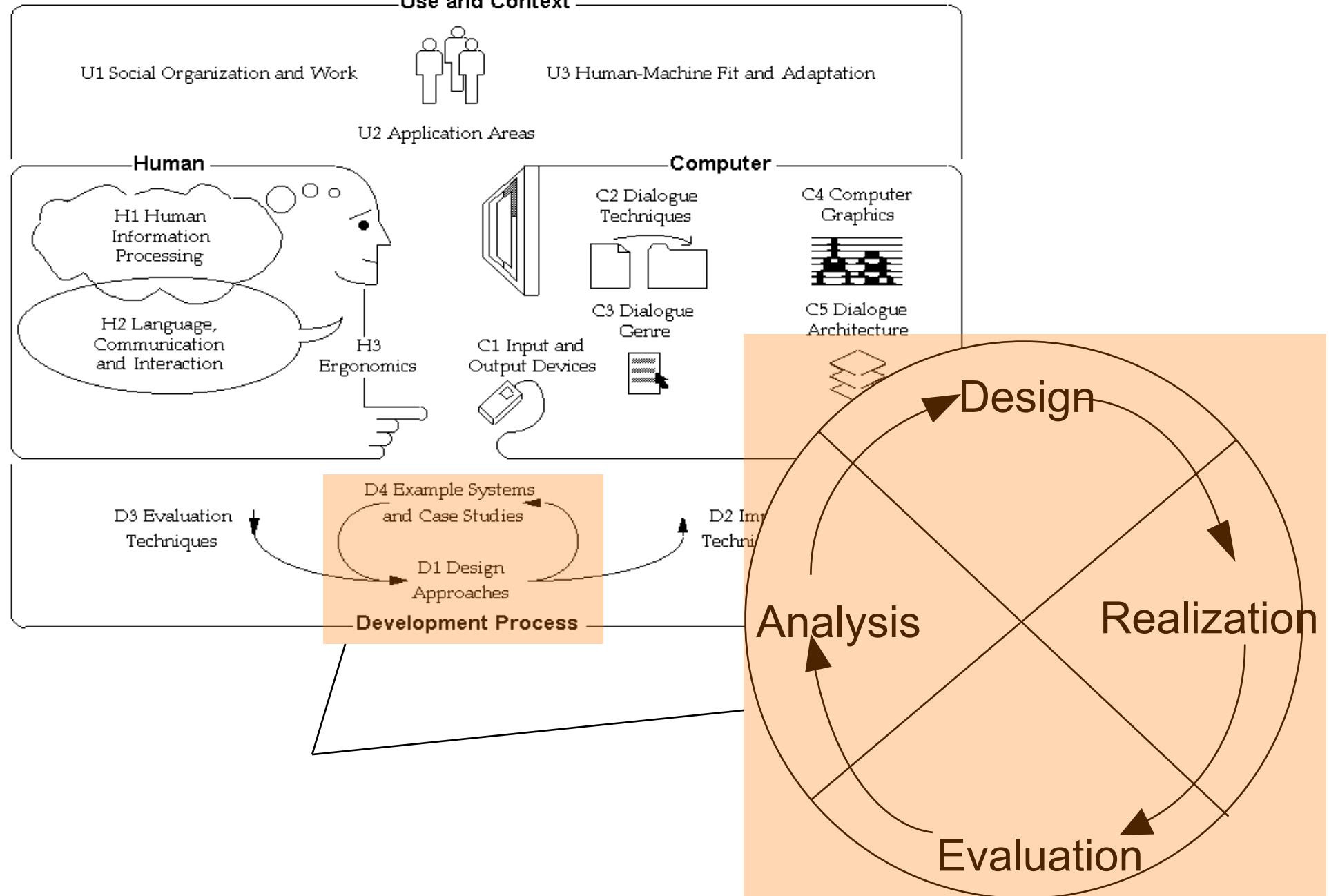
# Software Development Process Model

- ***Process model***
  - Segmentation of the overall (team) activity of software development into smaller portions of work
    - high-level structure: phases
    - low-level structure: steps, activities
  - Definition of an order for carrying out work units
  - Guideline for the production of intermediate results

- Basic activities covered in all models:
  - Analysis
  - Design
  - Implementation
  - Validation (in particular Test, Integration)
  - Deployment (in particular Maintenance)

# "Pure" Waterfall Model
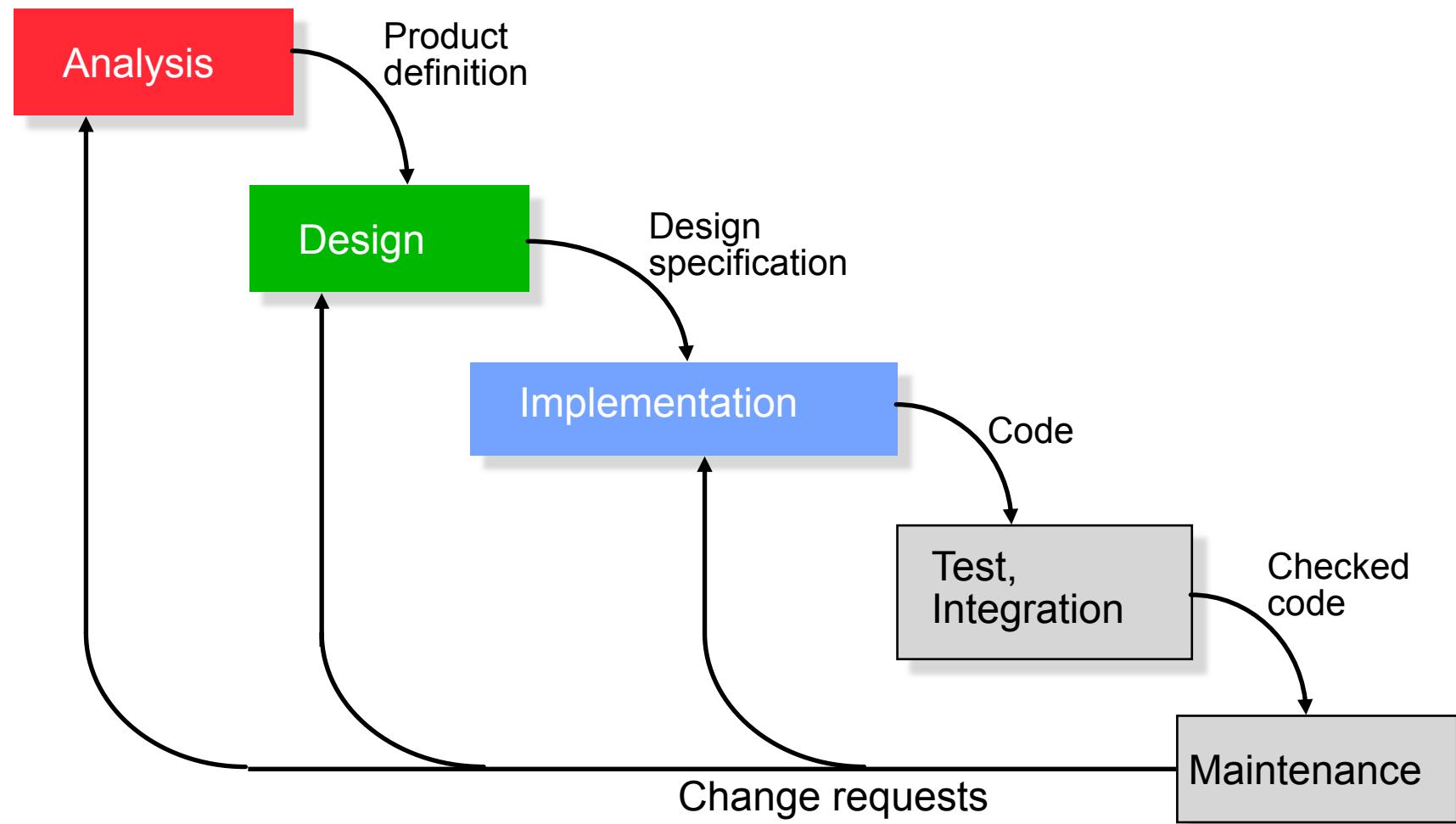


W. Royce (1970)

# "Waterfall" with Feedback Loops and Prototyping

This is how the original "waterfall model" by Royce looked like!
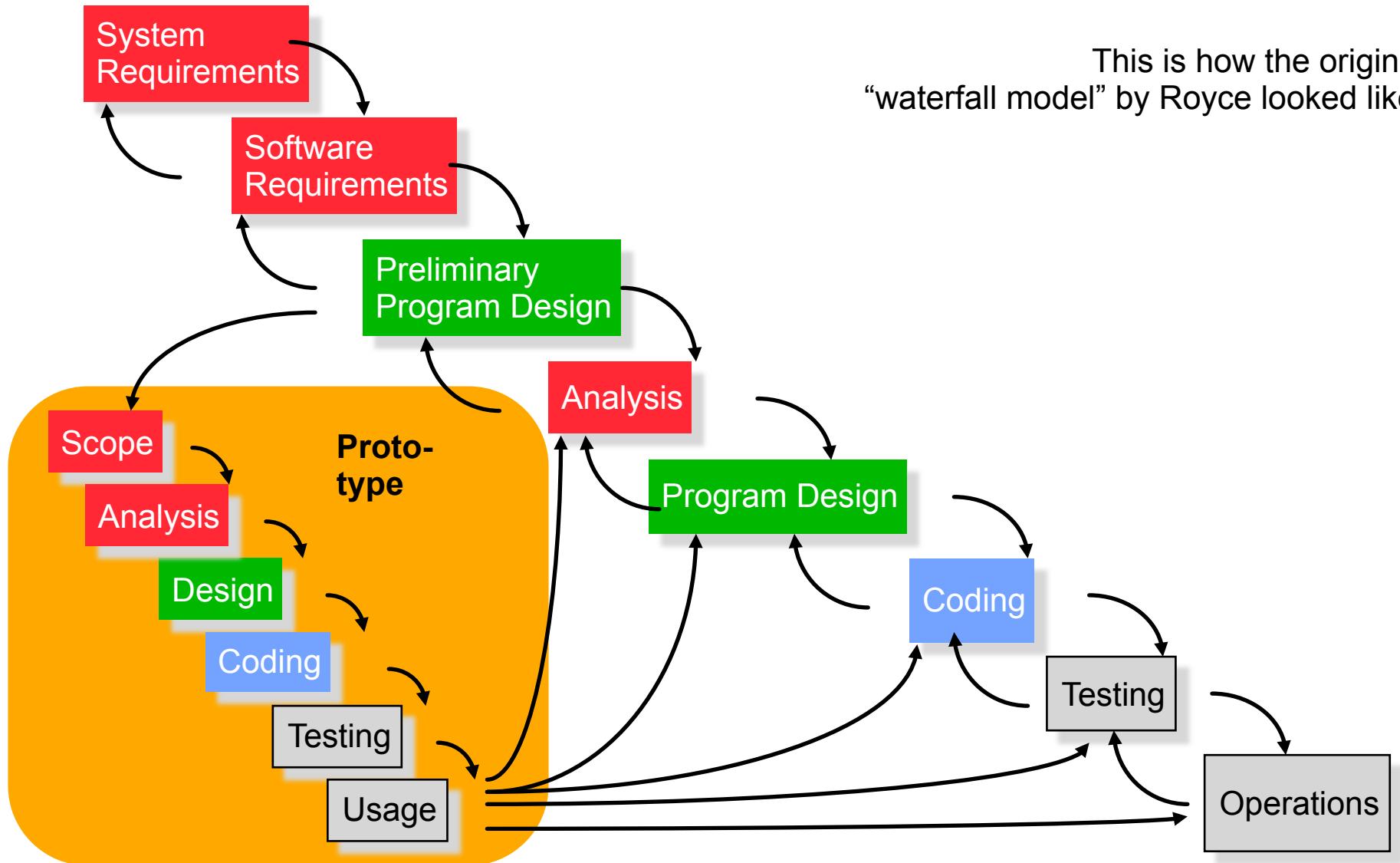
# Evolutionary Development
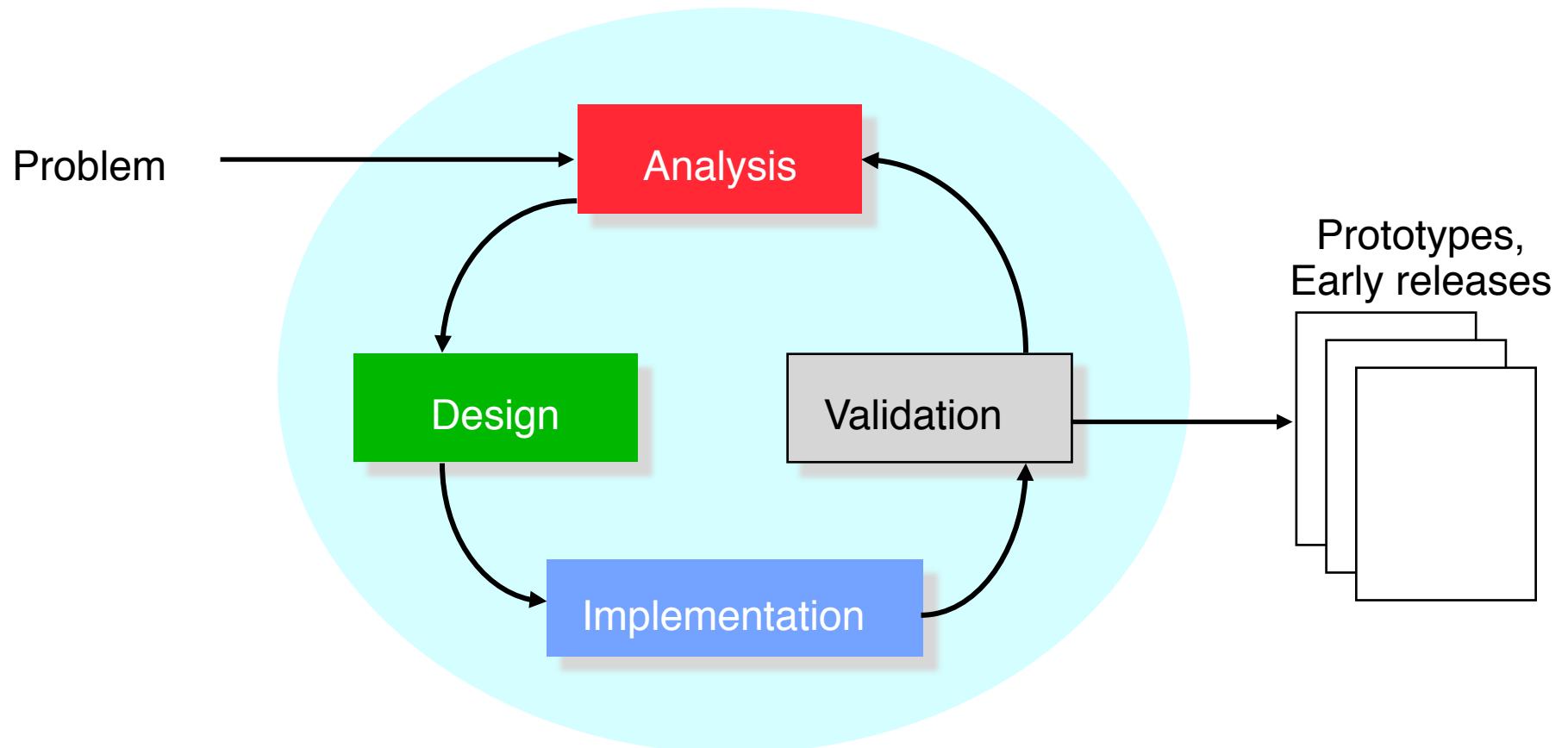


- Typical for small projects and experimental systems
- Technological progress (e.g. object orientation) may have improved scalability to large systems

# Modern Adaptation – "Spiral Model"



Design

Implementation

Analysis

Test

Products (Releases)
including *Prototypes*
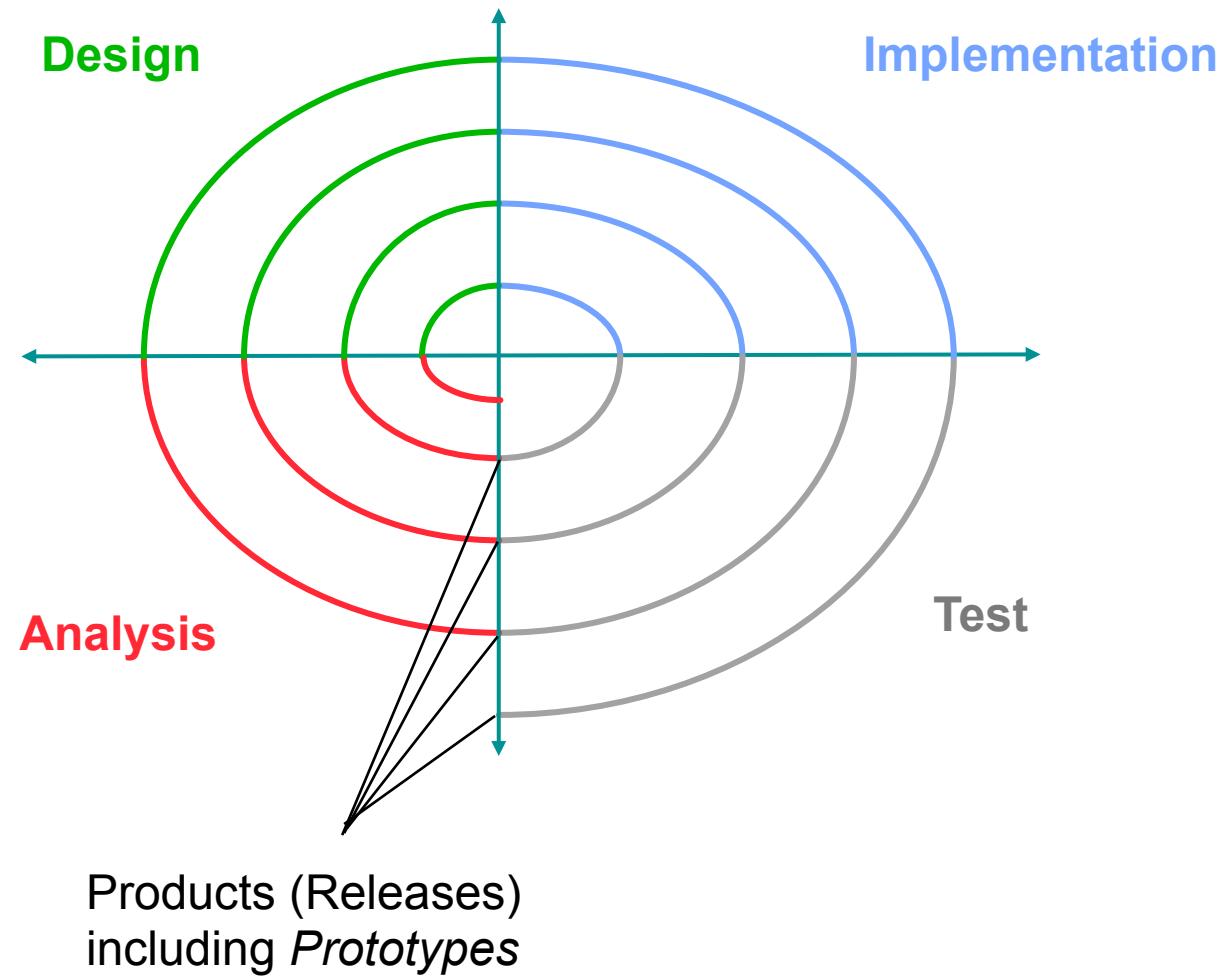
# Rational Unified Process (RUP)
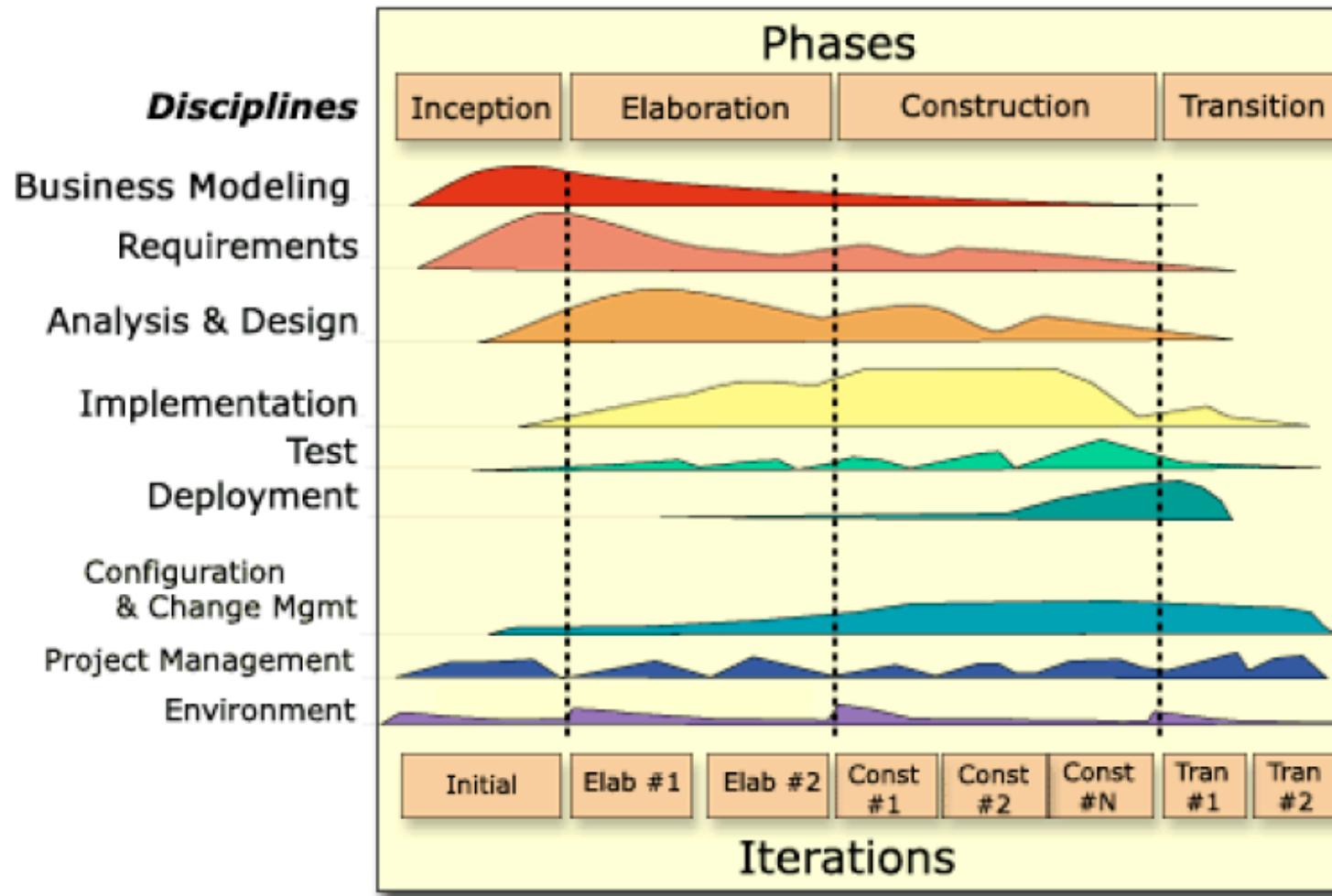


Software Process framework is a commercial product of Rational, now IBM.

# Detailed Prescriptions in RUP
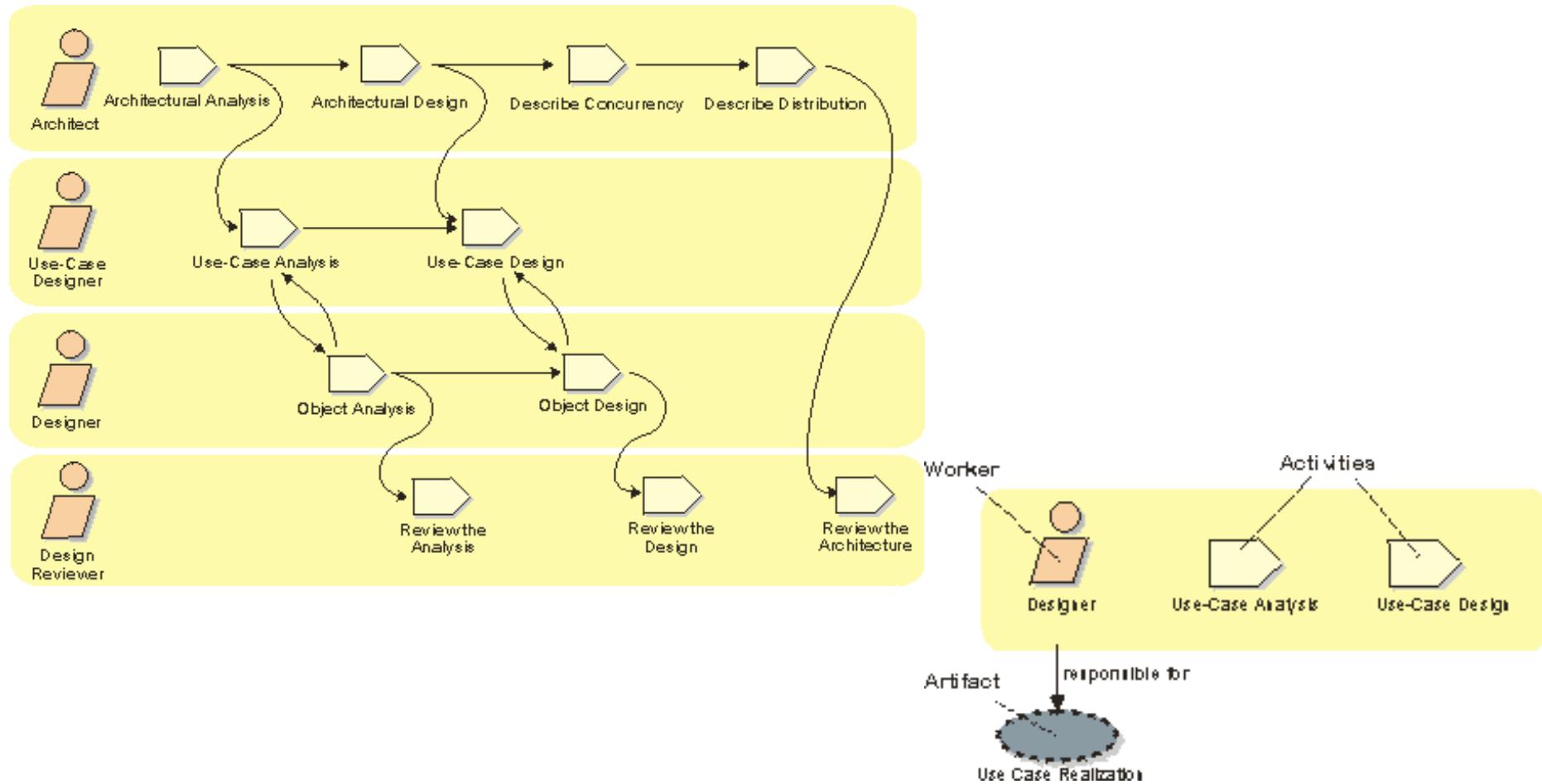


- Developers often consider this as not flexible enough for creative work.

# Think different ... be creative ...

# Agile Development

- "Agile" Software development (www.agilemanifesto.org):
  - E.g. Extreme Programming (XP), Crystal, Scrum

- Recent trend in software development processes
  - Radical evolutionary development

- Key characteristics of agile development:
  - Individuals and interactions (rather than processes and tools)
  - Working software (code rather than extensive documentation)
  - Customer collaboration (instead of contract negotiations)
  - Responding to change (instead of following a plan)

- Agile development is not just "hacking along"!
  - Clear and strict rules

- Mixed information about success in practice
  - Good experiences in small and innovative projects
  - Large-scale projects tend to stay "conservative", mainly due to transparency for project management

# User-Centered Development Process

- Software Development Process Models
- User-Centered Development
- Integrating Usability into the Development Process

# Usability Aspects are Mostly Ignored by Software Engineers

- Example:
  - IEEE "SWEBOK" body of knowledge definition for SE mentions HCI as "related discipline" under the name "software ergonomics"

- System perspectives
  - SW Engineers take the "System 1" perspective
  - Usability Engineers take the "System 2" perspective (following examples)

System 2

User ↔ System 1 Interface | Application
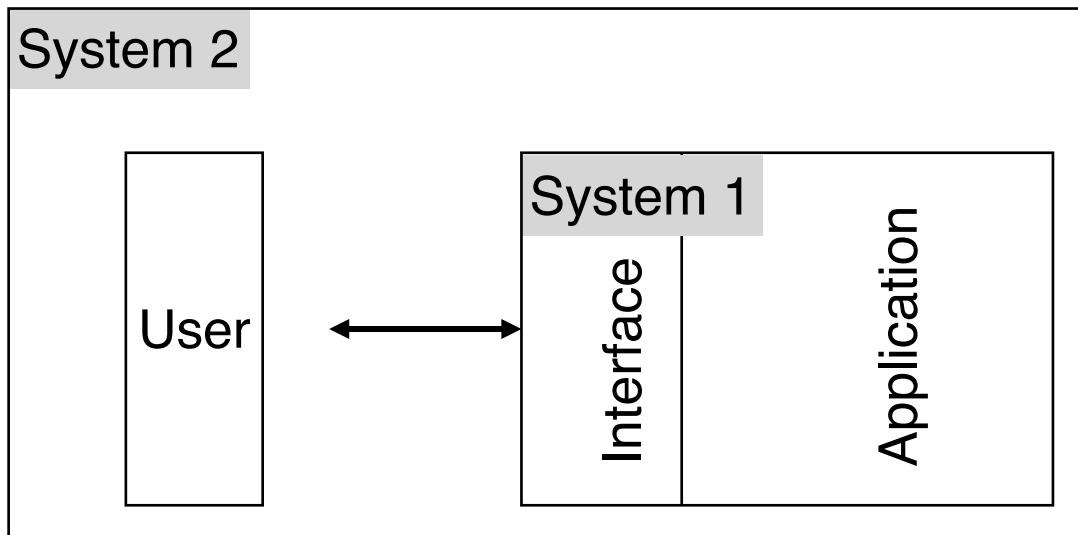
Seffah/Desmarais/Metzker

# Separation between Interaction Design and Technical Design

- For interactive applications a separation into a two stage process is often advisable

- 1st – Interaction design (iterative)
  - concept
  - Interaction analysis
  - Prototypes
  - Evaluation
  - Stable and tested design

- 2nd – Technical realization
  - Technical analysis
  - Technical specification (e.g. architecture, platform)
  - Implementation
  - Evaluation and Quality management

# Star Lifecycle
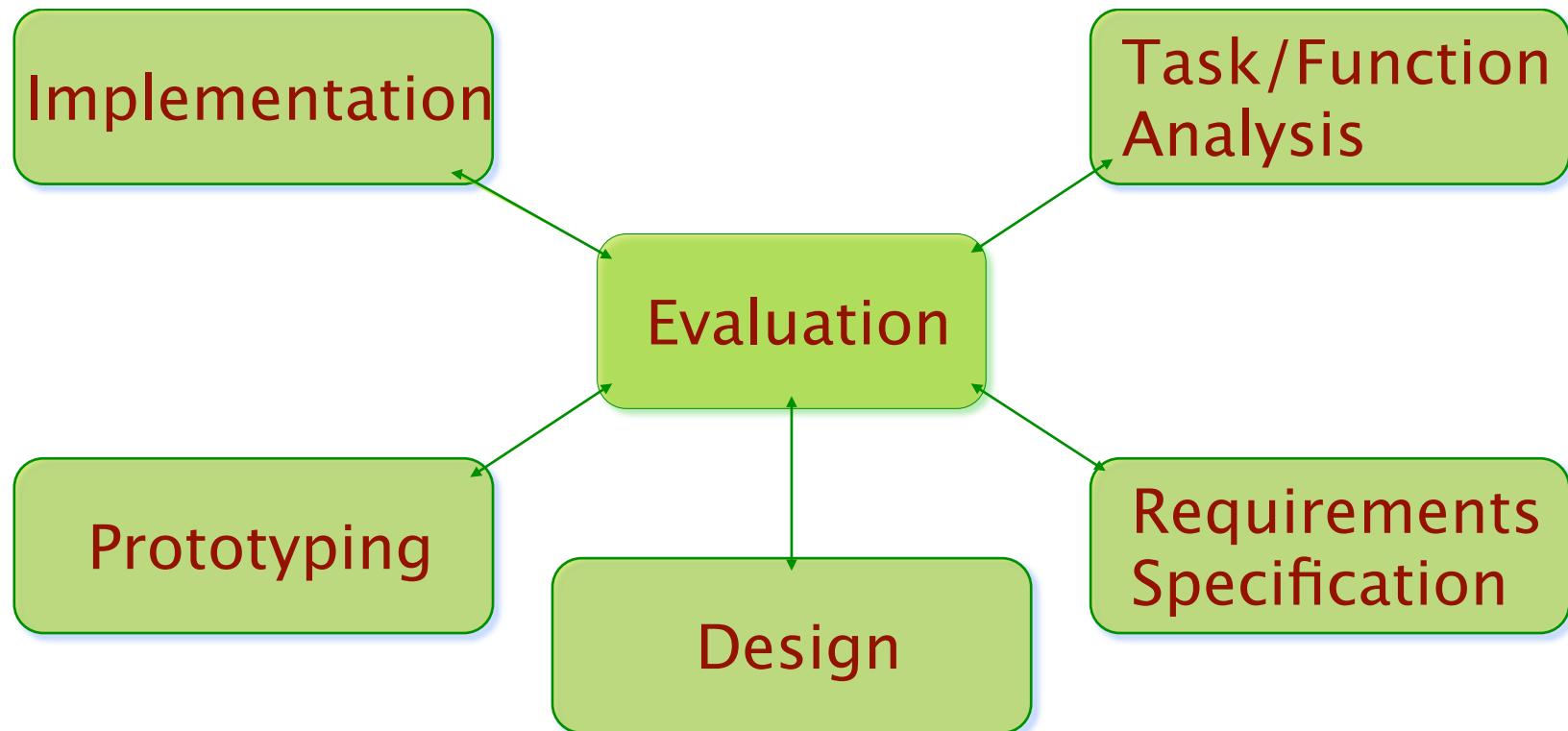


- Hix, Hartson 1993
  - Non-sequential: any order of activities
  - Evaluation-centric: every activity is evaluated
  - Interconnected: evaluation connects everything

# Scenario-Based Design

- Mary Beth Rosson, John M. Carroll: Usability Engineering - Scenario-Based Development of Human-Computer Interaction, Academic Press 2002

**ANALYSIS**

Analysis of stakeholders, field studies → Problem scenarios ← Claims about current practice

↓

**DESIGN**

Metaphors, information technology, HCI theory, guidelines ↔

Activity scenarios

Information Design Scenarios

Interaction Design Scenarios

↔ Iterative Analysis of usability claims & redesign

↓

**PROTOTYPE & EVALUATION**

Summative evaluation ← Usability specifications → Formative evaluation
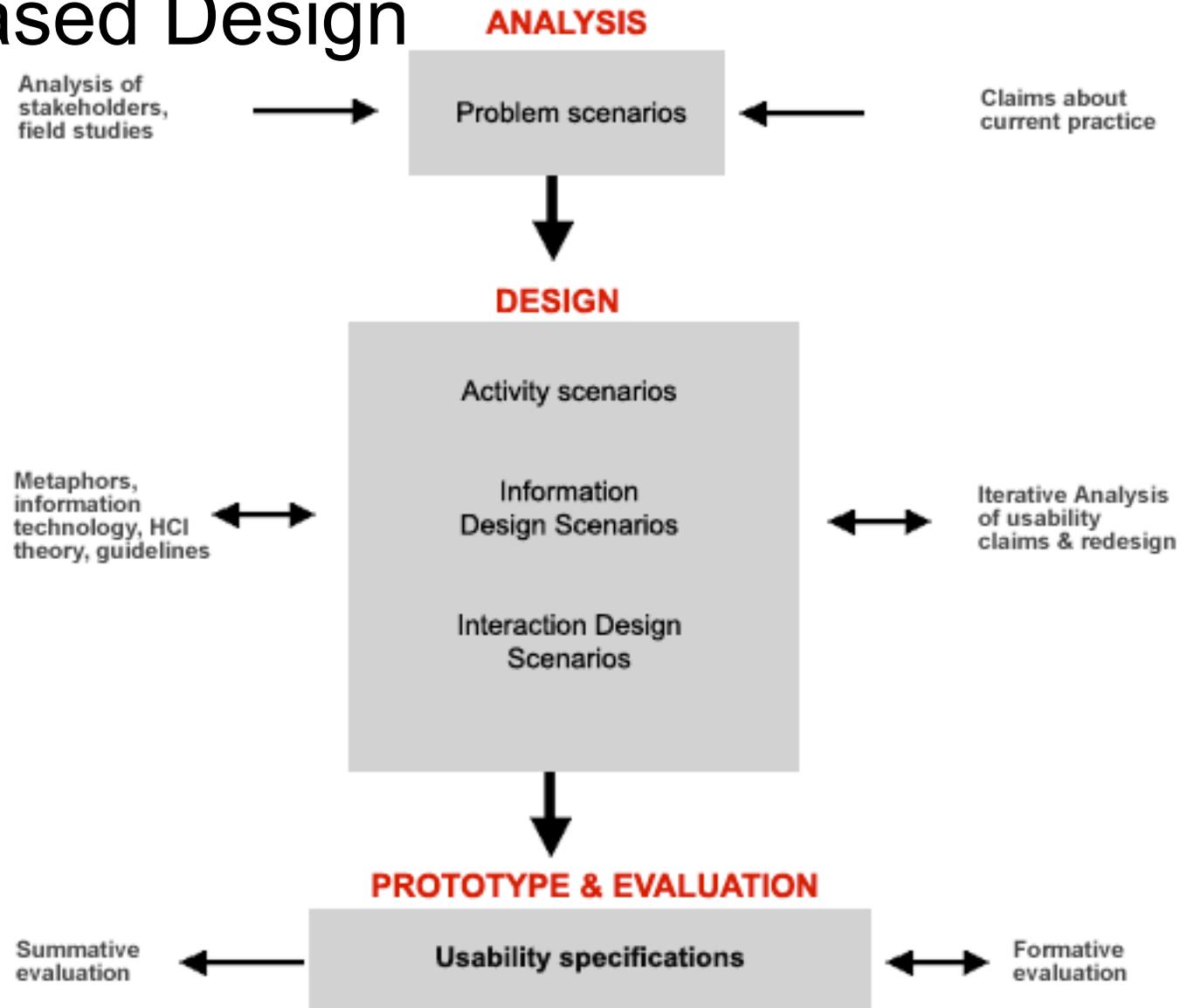
# ISO 13407



ISO 13 407 Model Overview

- 1. plan the human centered process
- 2. specify the context of use
- 3. specify user and organisational requirements
- 4. produce design solutions
- 5. evaluate designs against user requirements
- complete
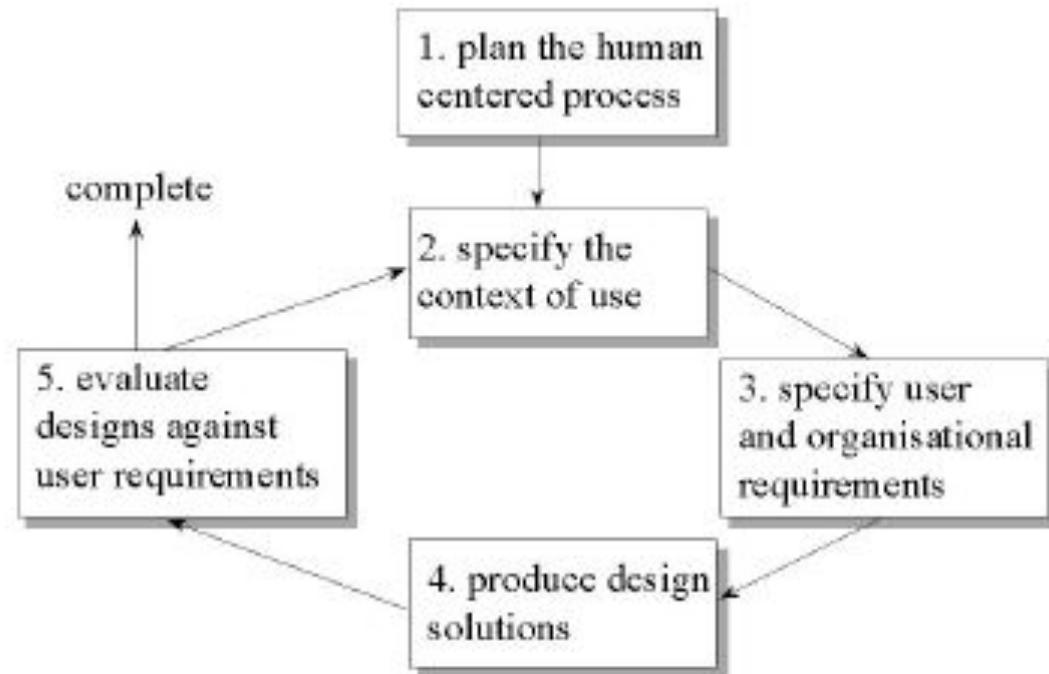
- Guidelines for integrating usability aspects into the development process
  - Proposes iterative process
  - Stresses evaluation
  - Design solutions cover also lightweight prototypes, mock-ups etc.

- See e.g. http://www.ucc.ie/hfrg/emmus/methods/iso.html

# Problems of User Centered Design

- Users may be wrong

- Users may be resistant to change

- Users may expect disadvantages (e.g. being replaced by software)


- Be aware – you are expected to create an optimal system with regard to *the goals specified*
  - this is unfortunately NOT necessarily the system users would like to have (e.g. trade-off between employers and employees)

# User-Centered Development Process

- Software Development Process Models

- User-Centered Development

- Integrating Usability into the Development Process

# Concurrent Workflows, Competing Cultures

- User Interaction Design and Software/System Design are concurrent activities
  - One depends on the other, one influences the other
- Separate cultures led to competing models of the development process
  - Software Engineering: Artefact-centric (e.g. design documents), disciplined order of steps, quantitative evaluation (metrics, tests), highly compatible to project management needs
  - User-Centred Development: Interdisciplinary, loose (e.g. rough guidelines), flexible in order of steps, open to late changes, continuous qualitative evaluation (e.g. user tests with prototypes), difficult to "sell" to project managers
- Ambiguous overlaps in terminology
  - The same terms are used in many methods with differently defined or weakly defined semantics
    - E.g. "scenario", "use case", "test"
- Integration of process models
  - "Interface development is transitioning from an artistic exercise into an engineering discipline." (Curtis/Hefley)

# Concurrency of UI and SW Engineering

| User Interface Engineering | Development Phase | Software Engineering |
|---|---|---|
| User & task analysis ⇐ | **Requirements analysis** | ⇒ Application design |
| Human vs. Machine ⇐ | **Requirements allocation** | ⇒ Hardware vs. Software |
| Dialogue design ⇐ | **Preliminary design** | ⇒ Architectural design |
| Screen design ⇐ | **Detailed design** | ⇒ Logical design |
| Coding ⇐ | **Implementation** | ⇒ Coding |
| Usability lab ⇐ | **Implementation testing** | ⇒ Unit & Integration testing |
| Contextual observation ⇐ | **System testing** | ⇒ System testing |
| Human performance ⇐ | **Optimization** | ⇒ Machine performance |

(Curtis/Hefley)

# User Experience "Plugin" for the RUP

- Extensions of roles, activities and (UML) artifacts
  - Use cases extended by "use case storyboards"
  - UI Prototyping as a specific activity
  - Screens as special cases of classes
    (derived from Conallen's UML-based Web Design Method)

- Steps to create User Experience Storyboards:
  1. Add actor characteristics to the use case.
  2. Add usability guidance and usability requirements to the use case.
  3. Identify UX elements.
  4. Model the use-case flows with the UX elements.
  5. Model screen navigation for the use case.

http://www-128.ibm.com/developerworks/rational/library/content/RationalEdge/nov03/f_usability_jh.pdf

# User Experience "Plugin" for the RUP

**Actor characteristics**

Usability guidance / requirements

UX elements

Use-case flows

Screen navigation

> **4.3 Actor Characteristics**
>
> **4.3.1 BUYER**
>
> 4.3.1.1 FREQUENCY OF USE
>
> 4.3.1.1.1 The typical Buyer will bid on an item three times per week.
>
> 4.3.1.1.2 Near the end of an auction, bidding activity may be very intense.
>
> 4.3.1.2 GENERAL LEVEL OF COMPUTER EXPERIENCE
>
> 4.3.1.2.1 The typical Buyer only uses his/her computer on a casual basis.
>
> 4.3.1.3 ENVIRONMENT
>
> 4.3.1.3.1 The typical Buyer uses the system from his/her home.
>
> 4.3.1.4 NUMBER OF USERS
>
> 4.3.1.4.1 The targeted number of users is 50,000.

**Figure 5: Actor characteristics for the Bid on Item use case**

# User Experience "Plugin" for the RUP

Actor characteristics

**Usability guidance / requirements**

UX elements

Use-

Scre



4. Special Requirements

4.1 User Experience Guidance

4.1.1 At AF PENDING PAYMENTS, pending payments normally occur in only 10% of the cases.

4.1.2 At AF INVALID BID ENTERED, Invalid bids are normally entered 15% of the time.

4.1.3 At BF BUYER CONFIRMS BID, the legal statement will be approximately 150 characters in length.

4.1.4 At BF ENTER AMOUNT the system should automatically provide choices at the next three bid increments.

Figure 6: Usability guidance for the Bid on Item use case



4.2 Usability Requirements

4.2.1 The Buyer must be able to confirm his/her bid with a single mouse click.

4.2.2 The system must update the current bid within 5 seconds of the Buyer confirming his/her bid.

4.2.3 The system must return confirmation of an accepted bid within 2 seconds.

gure 7: Usability requirements for the Bid on Item use case

# User Experience "Plugin" for the RUP

Actor characteristics

Usability guidance / requirements

**UX elements**

Use-case flows

Screen navigation
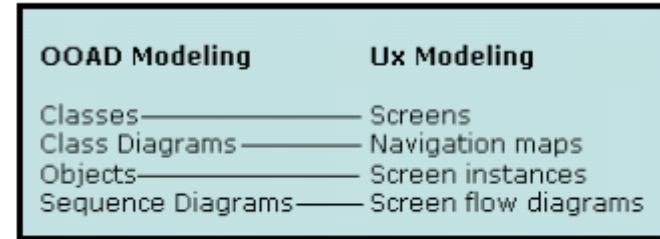


Figure 8: Mapping between OOAD and UX modeling elements
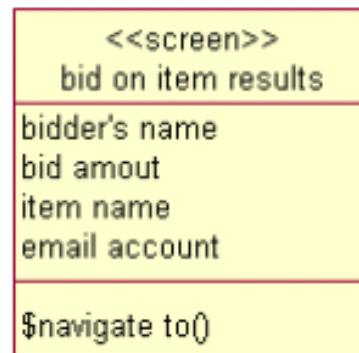


Figure 12: UML representation of a screen for
Bid on Item use case (basic flow)
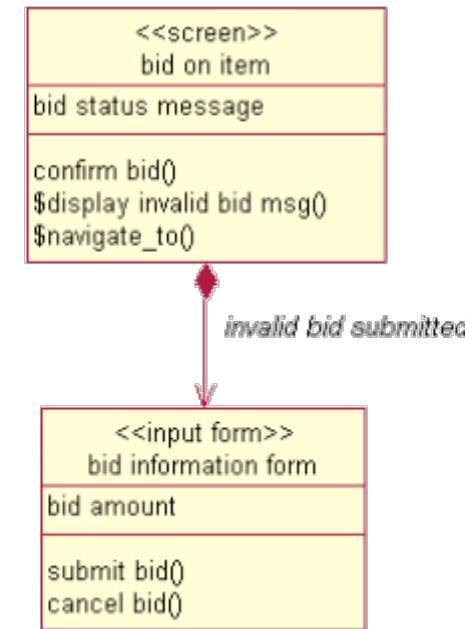
Figure 18: UML representation of a screen with an input form

# User Experience "Plugin" for the RUP

Actor characteristics

Usability guidance / requirements

UX elements

**Use-case flows**
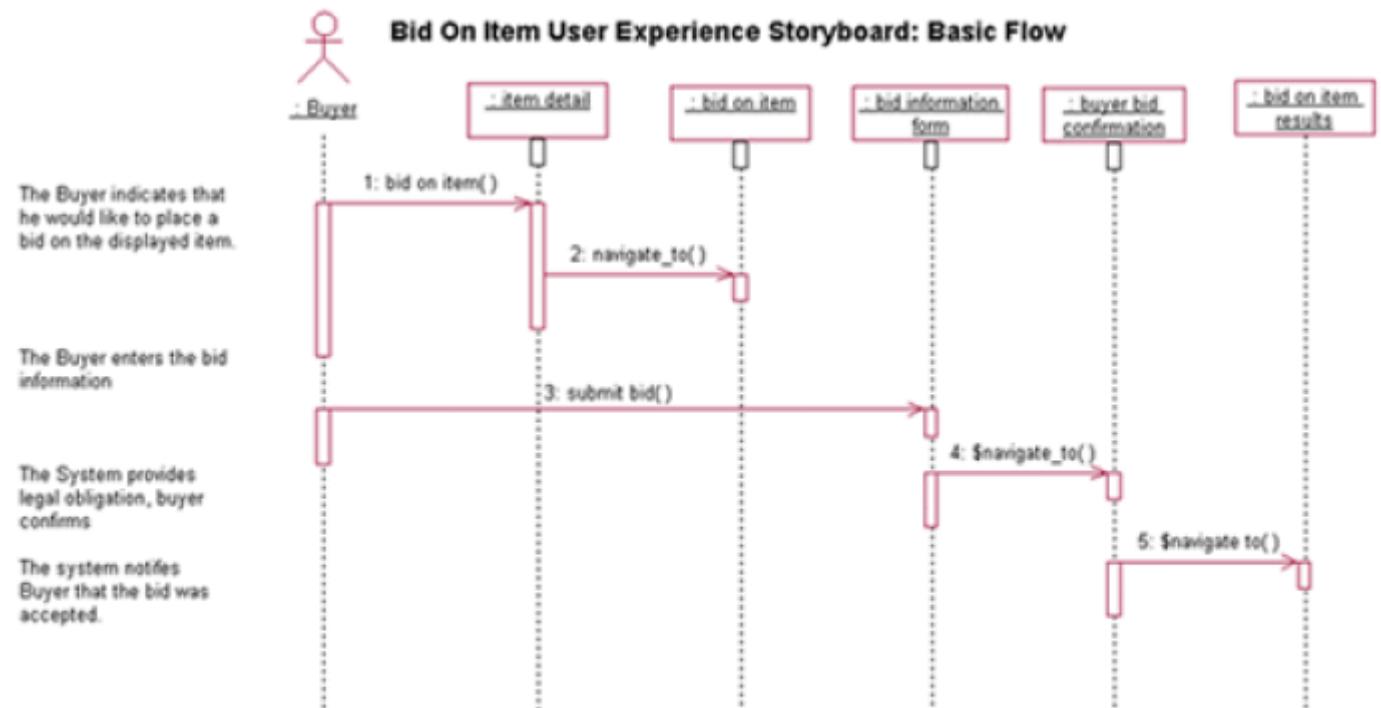
Screen navigation



Figure 19: Sequence diagram showing the basic flow of events for the Bid on Item use case

# User Experience "Plugin" for the RUP

Actor characteristics

Usability guidance / requirements

UX elements

Use-case flows

**Screen navigation**
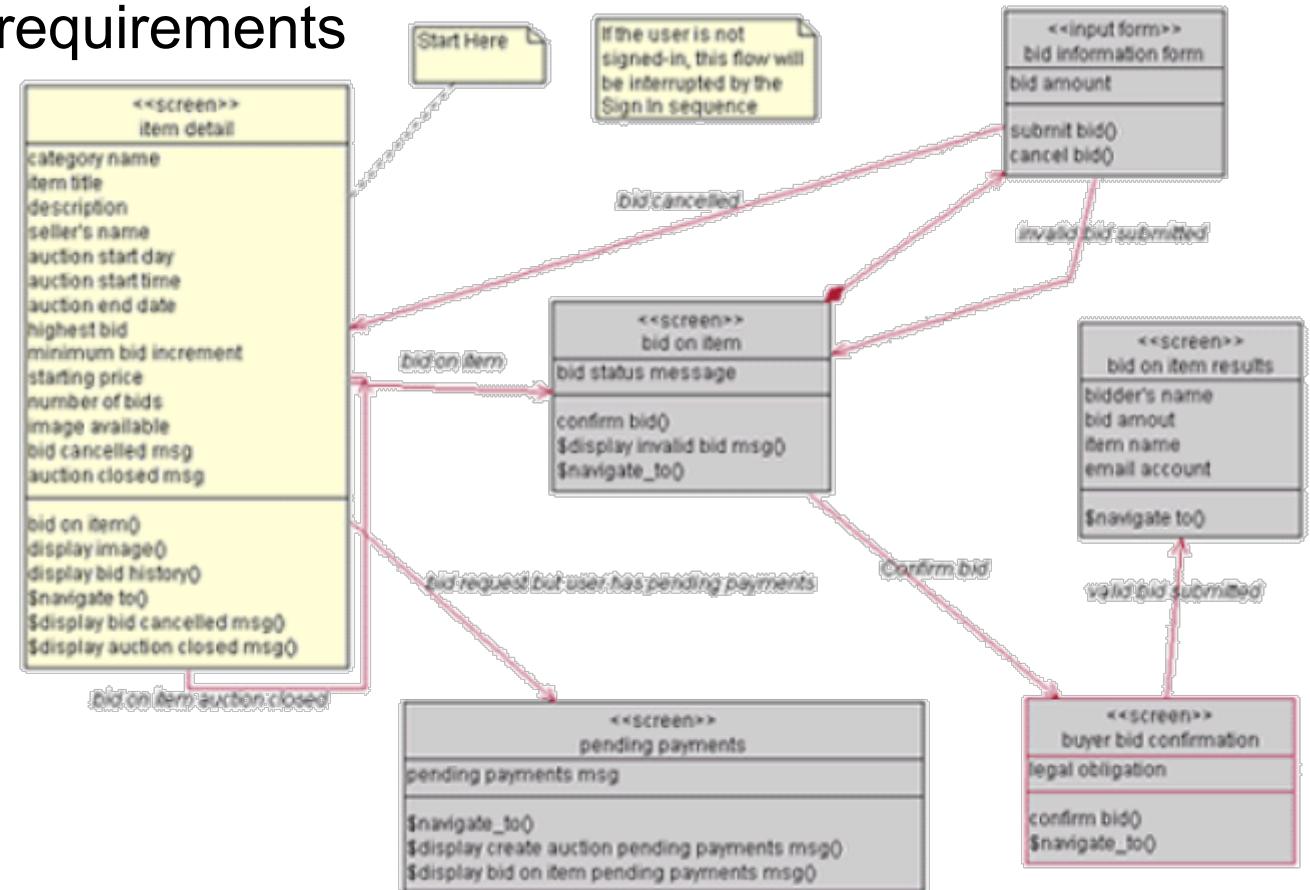


Figure 20: Navigation diagram for the Bid on Item use case

# Wrap-Up & Outlook

- Mensch-Maschine-Interaktion

- Informationsverarbeitung des Menschen

- Ein- und Ausgabeeinheiten für Computer, eingebettete Systeme und mobile Geräte

- Grundlagen und Beispiele für den Entwurf von Benutzungsschnittstellen

- Prinzipien, Richtlinien und Standards für den MMI-Bereich

- Designgrundlagen und Designmethoden

- Methoden zur Modellierung von Benutzungsschnittstellen

- Evaluierung von Systemen zur Mensch-Maschine-Interaktion

- Mensch-Maschine-Interaktion 2:

  - HCI and the Web

  - Mobile and Ubiquitous User Interfaces

  - Information Visualization

  - Interactive Surfaces

- Next lecture: results from and evaluation of (video) prototypes

# References

- Ahmed Seffah, Jan Gulliksen, Michel C. Desmarais (eds.): Human-Centered Software Engineering - Integrating Usability in the Development Process, Springer 2005

- Mary Beth Rosson, John M. Carroll: Usability Engineering - Scenario-Based Development of Human-Computer Interaction, Academic Press 2002

- Deborah Hix and H. Rex Hartson: Developing User Interfaces Ensuring Usability Through Product & Process, John Wiley 1993

- Bill Curtis, Bill Hefley: A WIMP no more: the Maturing of User Interface Engineering, ACM *interactions* 1(1), January 1994, 22-34

- John M. Carroll, Mary Beth Rosson: Getting Around the Task-Artifact How to Make Claims and Design Cycle: by Scenario, ACM Transactions on Information Systems, Vol. 10, No. 2, April 1992, 181-212