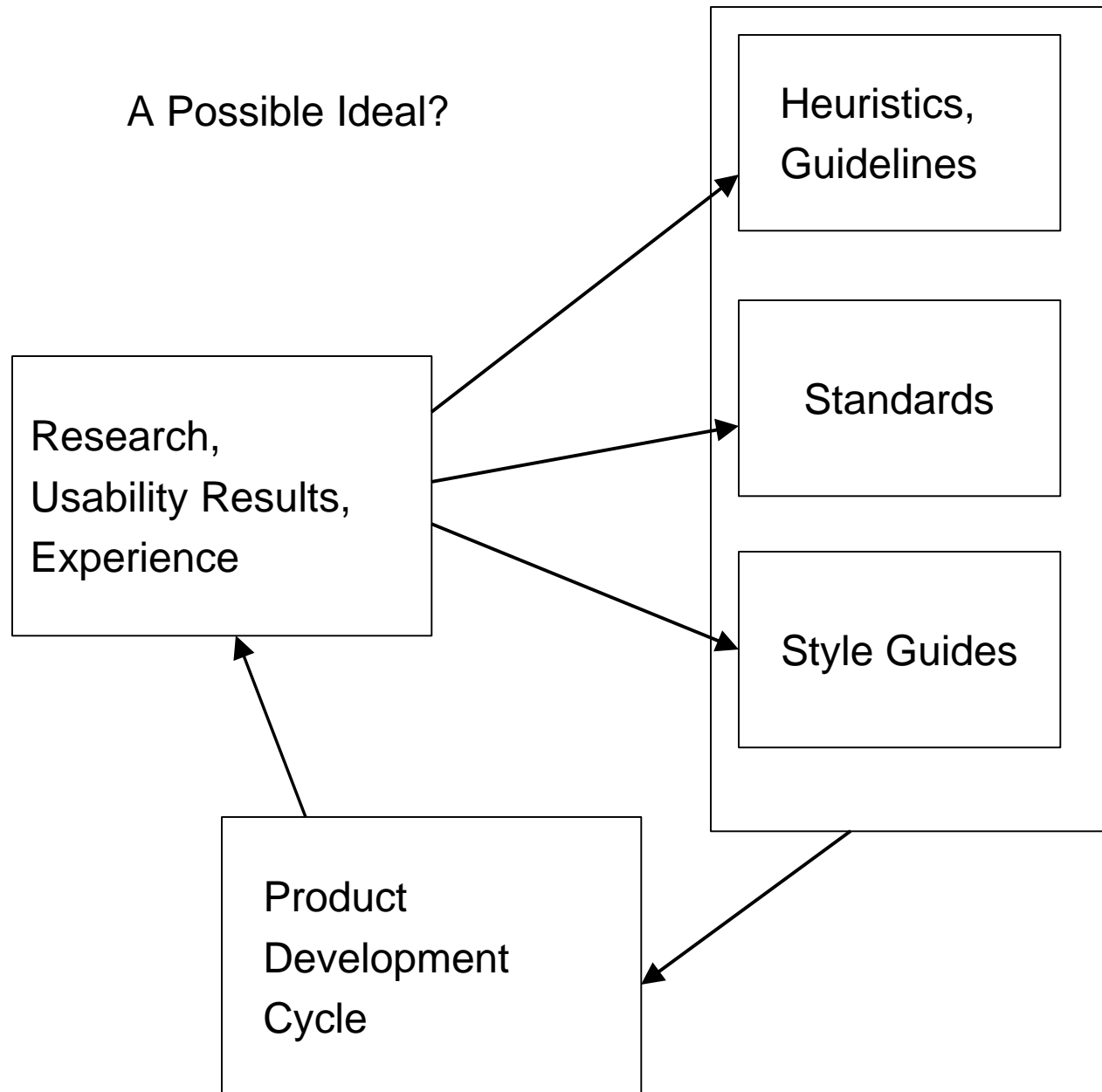


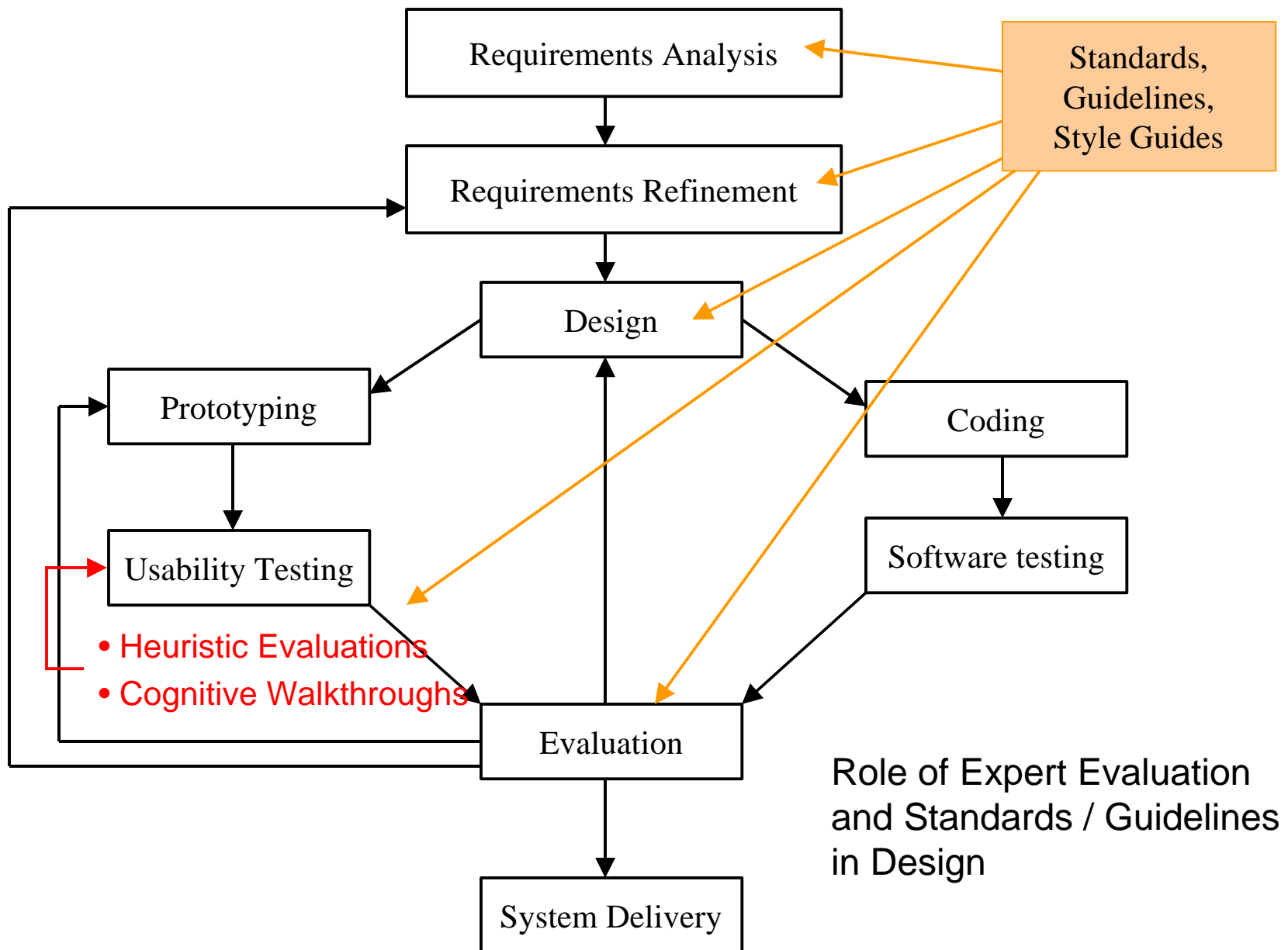
Lecture 6: Usability Methods III

- Standards, Guidelines, and Style Guides
- Rapid Prototyping
- Usability Testing
 - Verbal Reports
 - Performance Measures (“Usability Metrics”)
 - Questionnaires and Surveys
- Other Methods
 - Experimental Design
 - Storyboards, Scenarios, and Sketching



Standards, Guidelines, Style Guides

- Used in Design of System
 - Use set of guidelines in designing interaction
 - Refer to style guide in colors and shapes of windows
 - etc.
- Used in Heuristic Evaluation
 - Refer to general Nielsen principles in inspecting a prototype
 - Check program's appearance against style guide
 - etc.



Standards, Guidelines, Style Guides

Scope

- Standards
 - National
 - International
 - Company
 - Industry/Professional
- Guidelines
 - Published Literature (Books, articles)
 - Professional
 - Internal Company document
- Style Guides
 - Company
 - Industry
 - Project

Standards, Guidelines, Style Guides

Examples

- Guidelines
 - Smith and Mosier (1986) *Guidelines for designing user interface software*, Mitre
 - W. O. Galitz (1989) *Handbook of screen format design 3rd edition*, QED Information Sciences
 - Deborah J. Mayhew (1992) *Principles and guidelines in software user interface design*, Prentice Hall
 - Ballentine (1999) *How to build a speech recognition application*, EID
- Style Guides
 - Apple Macintosh Human Interface Guidelines
 - Microsoft Windows Interface Application Design Guide
 - OSF/Motif style Guide, Open Software Foundation
 - Open Look GUI Application Style Guidelines (Sun)
 - Java Look and Feel Design Guidelines (Sun, 1999)

Standards and Guidelines in Projects

Mulligan, Altom, & Simkin (1991)

- Adopt a User Interface Standard
 - Assuming:
 - You have corporate standards or adopted national/international standards
 - The standards are comprehensive and flexible
 - Allow for specific user needs or context (“It depends!”)
 - Organizations have “bought in” to standards
 - User Interface Standards convinces team of benefits
 - Consistent “look and feel”
 - Must have process of evolution from standard to innovations
- Project Specific Guidelines and Style Guides
 - Issues come up repeatedly, but aren’t covered by corporate or industry-wide standards
 - Examples
 - rules for menu format
 - window titles
 - form field labels
 - table layout
 - “Publish” style guide
 - Prevent making same decisions over and over again

Rapid Prototyping

Rapid Prototyping versus Prototyping (1)

- Prototype
 - Actually works, not a document, description, or simple drawing
 - Limited lifetime
 - Is discarded, or
 - Evolves into the actual system
 - Serves many purposes
 - Demo to management and marketing
 - Usability testing
 - Demo in marketing or user interface focus groups (show to customers)
 - Requirements for use by developers
 - etc.
 - Built quickly and cheaply
 - Required in Iterative Design Process

Rapid Prototyping versus Prototyping (2)

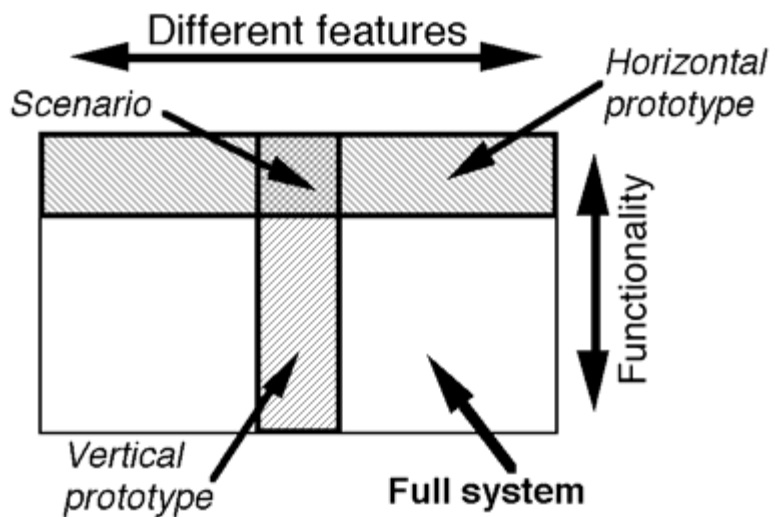
- Rapid Prototyping
 - Emphasis on *rapid*
 - Typically built on different platform or medium than actual system
 - Prototyping tool
 - Can be paper or slide show
 - Does not have “back-end” functionality, e.g. database access, etc.
 - Often, does not realize complete functionality of eventual system
 - By definition, a rapid prototype is *discarded* at the end of the design process
 - Purpose is for iterative design and testing
- Full Prototype
 - Other end of the spectrum
 - Complete functionality

Incremental and Evolutionary Prototyping

- Incremental Prototyping
 - Large system developed in phases implementing core functions first
 - Core feature system or “skeleton” delivered first
 - Requirements and system can be evaluated in the field before full implementation of all features
 - Less important features added later in phases
- Evolutionary Prototyping
 - Prototype is changed continuously until it becomes final system
 - Compromise between production and prototyping
- See Ch. 27

Horizontal and Vertical Prototypes

- Horizontal Prototype
 - Implements user interface but with *no* functionality
- Vertical Prototype
 - Contains all functionality (incl. “back end”) but only for part of the system
- Scenario (Nielsen, 1994)
 - Takes prototyping to extreme
 - Reduces level of functionality and number of features
 - User must follow a strictly limited test path
 - Paper or simple computer mock-ups
 - Cost-effective and simple to do



Nielsen (1994) Guerrilla HCI; using discount usability engineering to penetrate the intimidation factor. In Bias & Mayhew *Cost Justifying Human Factors*. (diagram also from this article)

High versus Low Fidelity Prototypes

- High fidelity
 - As close to the real system as possible
 - Polished look that matches what users will actually see
 - Most useful for demos, marketing, sales, etc.
- Low fidelity
 - Uses medium that is further away from actual system
 - Paper instead of computer screen
 - Fast and cheap, suited for rapid iterative design
- Research shows that low fidelity prototypes compares favorably to high fidelity prototypes in catching usability problems

Catani & Biers (1998)

- MS Windows library search software
- Compared three levels of fidelity in the prototype
 - Low (paper, static screens)
 - Medium (non-interactive slide show of screen shots)
 - High (Visual Basic prototype)
 - i.e., includes dialog flow from screen to screen
- Prototypes constructed with prior usability problems in mind
- Users given search tasks, think-aloud method and rating scales
- With the most frequently occurring problems: great commonality between three prototype groups in users identifying those problems
- No significant difference in the frequency or severity of problems encountered by users as a function of prototype

Catani, M. B., & Biers, D. W. (1998). Usability evaluation and prototype fidelity: Users and usability professionals. *HFES Proceedings*, p. 1331.

Other Types of Prototyping

- Chauffeured Prototyping
 - Users watch while “expert” or developer “drives” the system
- Demos and presentations for focus groups
- Wizard of Oz Prototyping
 - A “hidden” human being performs the “back end” or machine part of the functionality
 - User interface is working, actions eventually performed by machine are simulated by human
 - “Pay no attention to the man behind the curtain”
 - Early stages with no functionality at all
 - Very useful for automated speech recognition applications for testing prompts and dialog prior to completion of speech engine

Other Types of Prototyping

- Chauffeured Prototyping
 - Users watch while “expert” or developer “drives” the system
- Demos and presentations for focus groups
- Wizard of Oz Prototyping
 - A “hidden” human being performs the “back end” or machine part of the functionality
 - User interface is working, actions eventually performed by machine are simulated by human
 - “Pay no attention to the man behind the curtain”
 - Early stages with no functionality at all
 - Very useful for automated speech recognition applications for testing prompts and dialog prior to completion of speech engine

Prototyping Tools

- Historical (more or less)
 - Late 80s early 90s
 - Dan Bricklin's Demo II
 - Prototype character-based screens
 - MS-DOS based, but often used for prototyping mainframe terminal user interfaces
 - Mock up screens
 - Menu-based system to define logic of linking screens together
 - Test user free to explore system, but system does not provide real data
 - Hypercard (MacIntosh) -- see Ch. 27
 - Asymmetrix Toolbook (PC) -- hypercard analog for PCs
 - Later versions allowed drawing capability which provided for prototyping of hardware device / button user interfaces (e.g. telephones)
 - System evolved into its major orientation being Web-based / HTML training applications

Contemporary Prototyping Tools (1)

- Graphical User Interfaces
 - Visual Basic
 - Visio Drawing Package
 - Can be used for one step beyond paper-and-pencil prototypes
 - Other drawing packages may also fill this role
- Devices with buttons and small screens
 - Altia Faceplate 4.0
 - HDML cellular telephone simulator
 - <http://developer.phone.com/>
- Multimedia
 - Macromedia Director -- See Ch. 27
 - GUIs, text, graphics, animation, sound and music, video
 - Web multimedia via Shockwave
 - Drag and drop, menu, and scripting language, “no programming”

Contemporary Prototyping Tools (2)

- Telephone User Interface
 - TFLX
 - Tool for making MacIntosh into sophisticated answering machine
 - “Graphical” programming environment
 - Connect boxes and lines as in a flowchart (instead of programming)
 - PML / VXML
 - HTML / XML like languages to specify telephone user interfaces
 - Special server displays VXML “page” over telephone
 - Aspires to be full telephone user interface programming environment
 - Develop complete voice mail or call routing touchtone applications
- Web Page Design (HTML Authoring Tools)
 - Macromedia Dreamweaver
 - Microsoft FrontPage
 - Adobe

Web Page Authoring Tools

- Completely blurs the distinction between prototyping and system development
 - Complete graphical authoring solutions to replace HTML coding
 - FrontPage: little or no access to HTML
 - Dreamweaver: flexible switching between graphical authoring and HTML editing
- Uses in practice:
 - Complete authoring of simple pages, personal or company
 - Prototyping of pages
 - Screen dump of page mock-ups put in requirements or shown to users
 - HTML output of authoring tool discarded
 - Actual system pages in complex applications may be CGI-generated
 - Output of authoring application modified
 - Commands call CGI, Java, or other applications, are inserted by programmers within HTML generated by graphical authoring tool
- Distinction between user interface designer and developer are being broken down

Demonstrations

- Visio 5.0
 - Windows Objects Template
- Macromedia Dreamweaver
 - Prototypes and real web pages
- Microsoft Powerpoint
 - GUIs (Web-based)
 - Telephone user interfaces (IVRs)
- Touchtone user interface simulation
 - PML / VXML

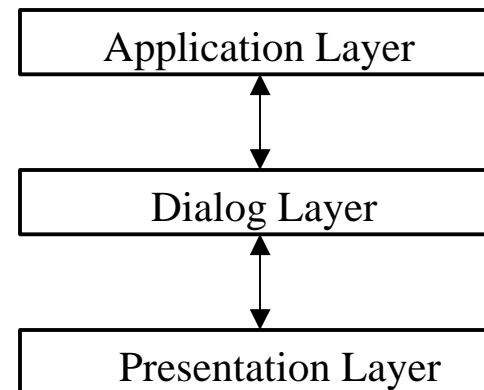
From Prototyping to Toolkits

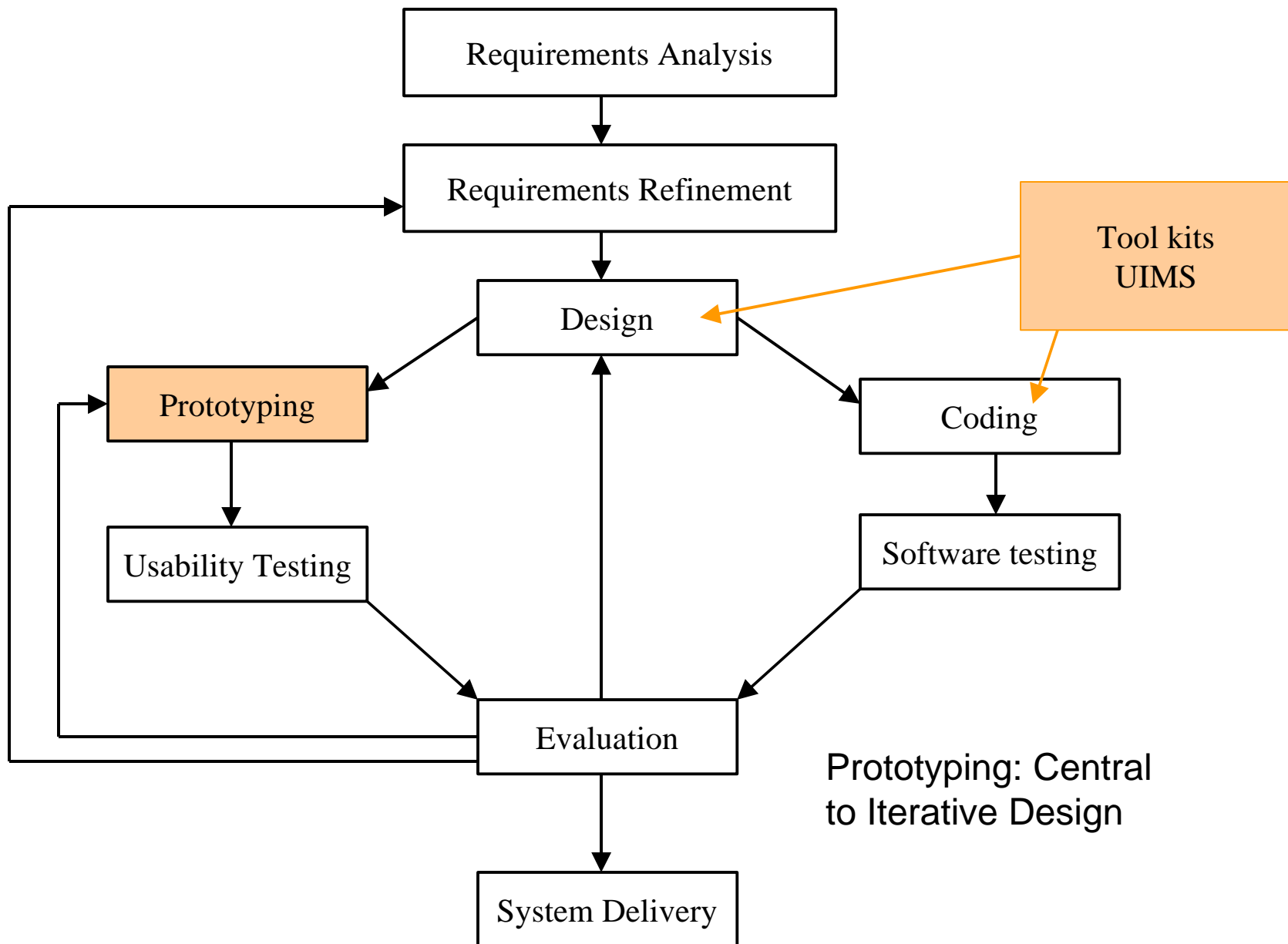
- Development environments serve as prototypers or development software platform
 - Microsoft Visual Basic
 - Microsoft Visual C
- User Interface Toolkits
 - Libraries of user interface objects
 - Icons, buttons, menus, scrollbars, dialog boxes (e.g. error messages), etc.
 - Drag and drop screen objects
 - Link together with scripting and/or programming language
 - Examples (See Ch. 28 for further elaboration)
 - NeXT User interface builder
 - UIM/X for OSF/Motif
 - Visual Basic / Visual

UIMS

- User Interface Management Systems
 - Not a prototyping tool, not a software toolkit, evolved from these
 - A UIMS is an authoring tool which is the system's final user interface
 - The UIMS is a software application which “runs” the user interface of the finished system
 - UIMS mediates the interaction between the user and the “application”
 - UIMS does all communication with user, “application” does the “work”
 - Examples:
 - Carnegie Mellon's Garnet (See Ch. 28 for those interested in more detail)
 - DRUID

Architecture separates core application functionality and user interface dialog





Usability Testing

