Benjamin Fenster
CS-296 (Damon)
15 March 2005

Interaction Models

I.   Introduction
    a. Execution-Evaluation
        i.   Execution: Decide what you want done (type a sentence) and what actions go into that.
        ii. Evaluation: Is the result what you wanted?
        iii. Gulf of Execution: Don't know what to do
        iv. Gulf of Evaluation: Don't know why something happened.
    b. Add the Computer
        i.   User articulates a request.
        ii. The computer performs an action.
        iii. The computer presents results.
        iv. The user observes results.

II.   Command Line
    a. User has to learn vocabulary
    b. Huge gulf of execution
    c. May be insufficient output too (just "?") so that's a gulf of evaluation too.

III. Menus
    a. This was the next phase historically.
    b. Very limited (like automated customer service menus, or an ATM)
    c. Because the hierarchy goes deep, may not know which top-level option you want.

IV.   Natural Language
    a. Computers can't interpret well.
    b. Gulf of evaluation

V.   Query Dialog
    a. Computer prompts with questions; give an easy answer (falls between menu and natural language). Limited vocabulary.
    b. Easy to "learn" / use. Very, very easy.
    c. It's not always clear which answer is correct. Gulf of execution.
    d. It feels like the computer is in control; that's not good.
    e. "Wizards" tone that down (the sense of control). How? Good question.

VI.   Forms
    a. Like query dialog, but you get to see what all the questions are at once.
    b. Same amount of work, but user has more control now – s/he can answer questions in any order, for example.
    c. Doesn't work for all applications.

VII. WIMP
    a. Windows, Icons, Menus, Pointers.
    b. Back to Xerox Star
    c. Select an object and then an action.
    d. Gives absolute control to the user.
    e. It's hard to see how to use it. Gulf of execution.

VIII. Point-Click
    a. Fundamental difference from WIMP: There's no "selection" and "action" there's just a single Click somewhere that invokes an action.
    b. Basic websites are point-click.
    c. Simpler. There are fewer things you can do at once. Can't just select an object and apply any number of actions, which means a lot of illegal actions are avoided.

IX.   Interface Metaphors
    a. These are the *styles* in use.
    b. We also want a *model* (metaphor) for how the application works.
    c. These are largely orthogonal to style, though some metaphors don't match with some styles.
    d. Metaphors should be consistent with the *external* view of the application.

e. Usually built around real-world objects: Desktop, Folder, Trash Can
f. Violating Metaphors
    i. The icon for the filing cabinet and the icon for the document are the same size. It seems like it won't fit.
    ii. The disk ("Filing Cabinet") holds essentially infinite documents.
    iii. Few people are bothered by these, but they do violate the metaphor.
    iv. Documents don't really have scroll bars – a button to turn the page would perhaps be better fitting. That still doesn't bother people after a while.
    v. Some things really destroy the metaphor: many people don't ever even *see* the desktop anymore, so what good does that do?
g. Uber Metaphor: Spreadsheet (ledger sheet). A computer spreadsheet is the only meaning now associated with that term. The metaphor took over.
h. Physical movement used as a metaphor for changing applications / documents
i. Physical objects: not broadly used
j. Characters / Wizards: Trying to give a sense that there's someone in the computer helping out.