

L'interazione

Fabio Vitali

HCI



Per ragioni di sicurezza, in una banca inglese la password per sbloccare le casseforti di filiale era divisa in due semichiavi assegnate a due manager diversi.

Poiché era ritenuto disdicevole che i manager digitassero di persona su una tastiera, era pratica comune che la semichiave venisse affidata ad una segretaria.

Contemporaneamente, una politica di austerità applicata dalla banca dimezzò il numero di segretarie, obbligando più manager a condividere la stessa segretaria. Ci furono dunque svariati casi di segretarie in possesso di entrambe le semichiavi!

La discrepanza non venne rilevata che vari anni dopo.



Introduzione

Oggi esaminiamo in breve il terzo sostantivo del nome del corso, e in particolare

- ◆ I modelli di interazione
- ◆ Il design del dialogo
- ◆ Il design dello schermo
- ◆ Alcune regole pratiche per massimizzare l'interazione



Modelli di interazione

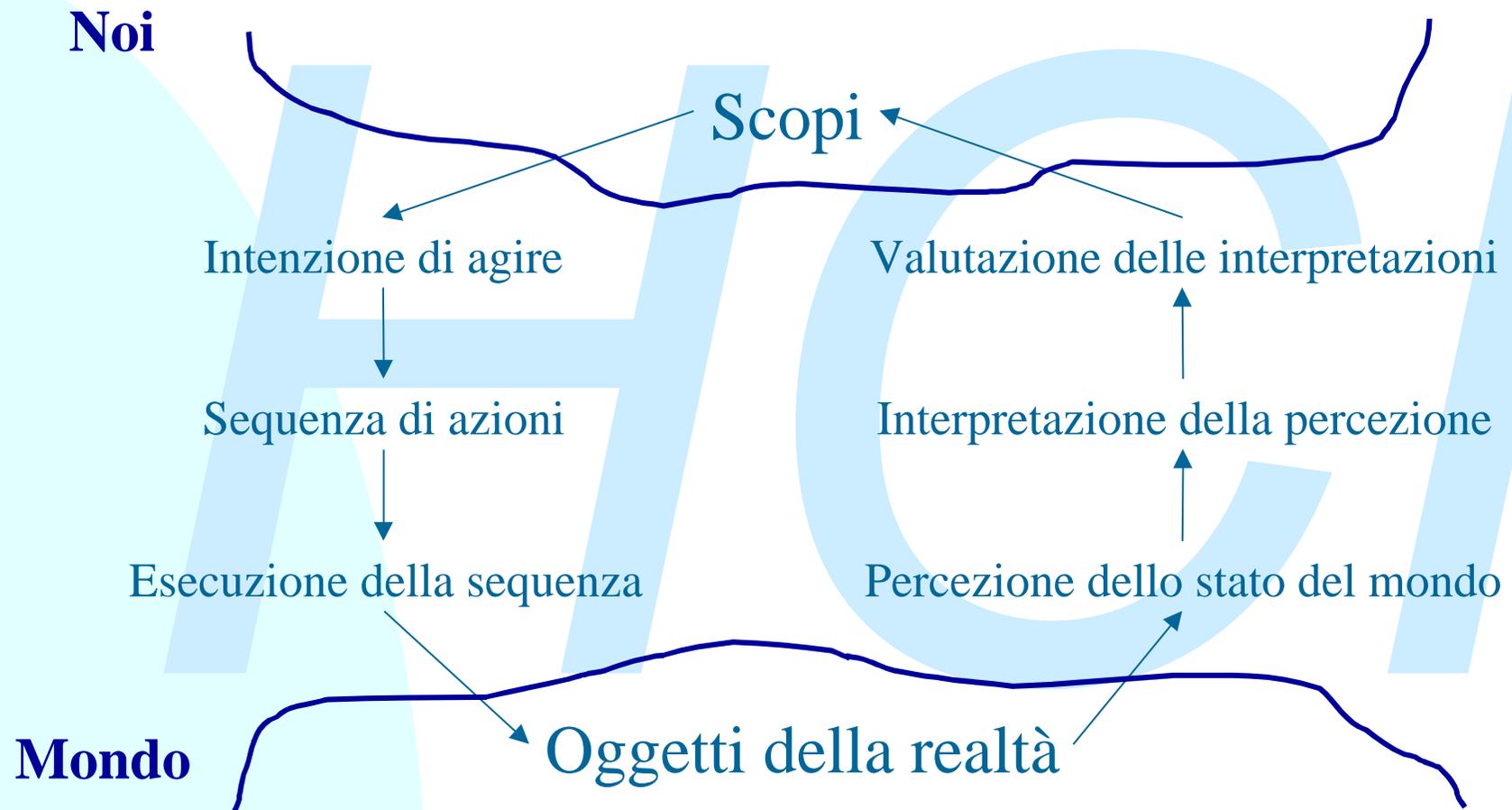
L'interfaccia è il luogo dove avviene l'interazione tra due sistemi complessi e disomogenei, e l'interfaccia realizza la traduzione del dialogo tra un sistema e l'altro.

Nel nostro caso, esseri umani e computer sono sistemi *particolarmente* complessi, e quindi maggiore sarà la possibilità di errori nella realizzazione di questa traduzione.

L'uso di modelli di interazione ci permette di evidenziare eventuali problemi di traduzione molto presto, e di confrontare tra loro le soluzioni.



Il modello di Norman

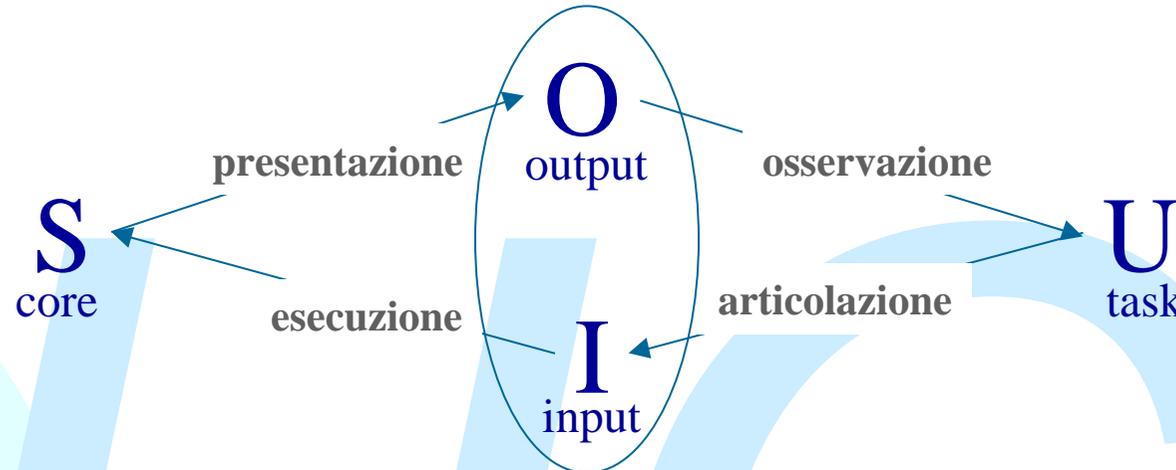


Il modello di interazione (1)

- Un **sistema interattivo** permette ad un **utente** di raggiungere un **goal**, ovvero uno scopo all'interno di un **dominio applicativo**, ovvero un'area di competenza e conoscenza in qualche attività.
- I **task** sono operazioni per manipolare i concetti del dominio, e il goal è il risultato desiderato di queste manipolazioni. Attraverso Input ed Output si ottiene il **dialogo** che realizza l'interazione.
- Ogni membro dell'interazione utilizza un proprio linguaggio, e compito del progettista è trovare una traduzione adeguata tra i linguaggi.
- Il sistema ha un proprio linguaggio detto core, di computazioni che è in grado di eseguire. L'utente ha un linguaggio di task in cui è in grado di esprimere i suoi goal.
- A loro volta, input ed output hanno un proprio linguaggio con cui sono in grado di mappare gli uni negli altri.



Il modello di interazione (2)



- Il goal dell'utente, espresso in un linguaggio di *task*, viene *articolato* nel linguaggio di *input*. Questa articolazione va valutata in termini di **facilità** di articolazione e **copertura** delle caratteristiche del task.
- Le attività dell'input sono poi trasformate in stimoli per il sistema, che *esegue* le computazioni relative. La trasformazione va quindi valutata in termini di **copertura** delle funzionalità del sistema.
- L'esecuzione di un'operazione mette il sistema in un nuovo stato, che viene *presentato* attraverso l'output. Va valutata la capacità dell'output di **catturare** le caratteristiche più rilevanti del nuovo stato del sistema.
- Infine l'utente *osserva* l'output e cerca di stabilire una correlazione con i propri goal. Va valutata la **facilità** di interpretazione e la **copertura** del goal.

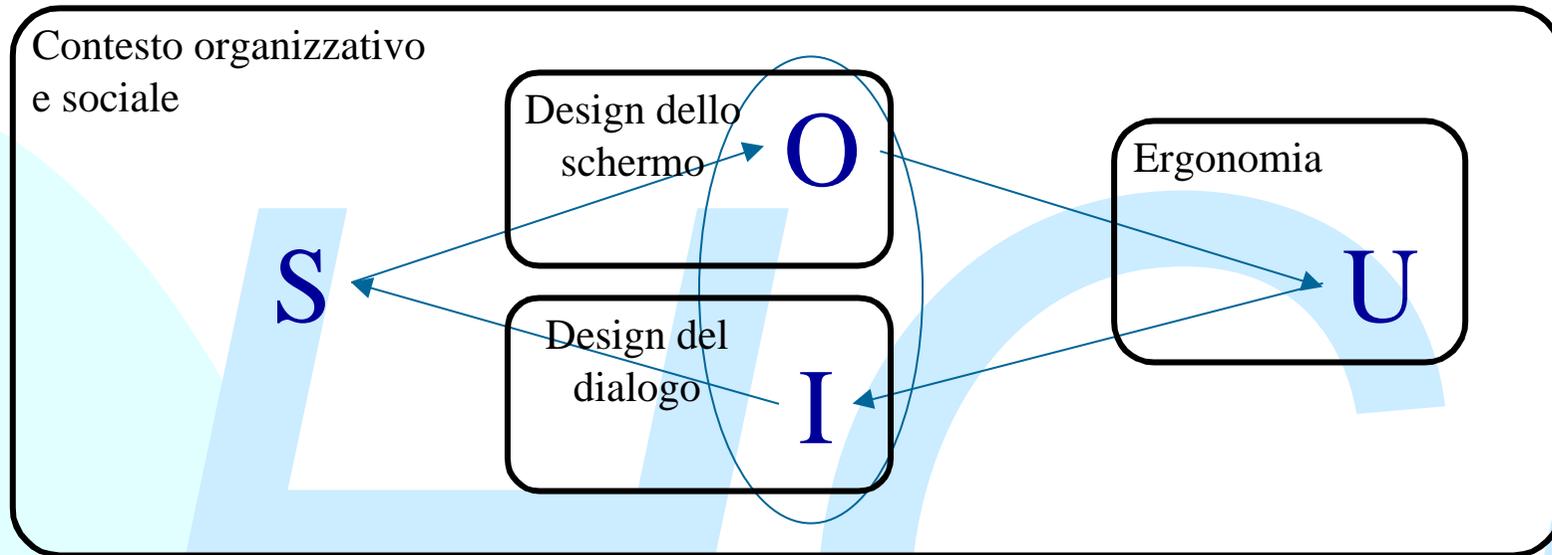


Il modello di interazione (3)

- Il modello di interazione qui presentato divide in quattro fasi la valutazione dell'interazione, e sempre come valutazione di correttezza, completezza e facilità di traduzione da un linguaggio ad un altro.
- L'enfasi nella valutazione è data in particolare alla capacità di tradurre i task umani in compiti al sistema, piuttosto che a metriche interne al sistema stesso. Questo è un modo complicato per dire che “il programma migliore dipende da quello che ci si vuole fare.”



Il modello dell'interazione e l'HCI



La *Association for Computing Machinery* ha identificato lo schema qui proposto per la classificazione delle varie aree di interesse dell'HCI.

- ◆ L'ergonomia è lo studio delle caratteristiche fisiche dell'interazione
- ◆ Il design del dialogo ci permette di scegliere tra vari stili di interazione
- ◆ Il design dello schermo ci fa organizzare il layout dell'interazione.
- ◆ Lo studio del contesto sociale ed organizzativo ci permette di situare l'interazione in un contesto più ampio.



Il contesto organizzativo e sociale

- L'interazione non avviene nel vuoto. C'è un contesto sociale ed organizzativo che va valutato.
 - ◆ Vedi al proposito l'aneddoto della cassaforte.
- Esistono fattori che influenzano notevolmente l'interazione con i sistemi complessi
 - ◆ Competitività tra pari,
 - ◆ Desiderio di impressionare il superiore,
 - ◆ Paura di sbagliare in pubblico, ecc.,
- Alan Cooper basa proprio sull'effetto sociale dell'interazione con il sistema tutto il suo modello di progettazione dell'interazione. Lo vedremo in seguito.



L'ergonomia

Lo studio delle caratteristiche fisiche dell'interazione e dei controlli che la permettono.

Lo scopo primario è l'incremento di efficienza degli esseri umani. Ci occupiamo brevemente di:

- ◆ Organizzazione di controlli e display
- ◆ L'ambiente fisico dell'interazione
- ◆ Aspetti connessi con la salute
- ◆ L'uso dei colori



Organizzazione di controlli e display

La disposizione fisica dei controlli e dei display è importante. In applicazioni critiche è fondamentale, ma anche nelle applicazioni su PC di tutti i giorni: tasti vicini possono avere significati molto diversi e potenzialmente in contrasto.

Il raggruppamento dei comandi è importante. Possiamo avere:

- ◆ **Raggruppamenti funzionali:** i comandi funzionalmente collegati sono messi vicini
- ◆ **Raggruppamenti sequenziali:** i comandi sono organizzati per riflettere l'ordine con cui vengono attivati (soprattutto in situazioni dove le sequenze sono obbligate: aviazione)
- ◆ **Raggruppamenti per frequenza:** i comandi usati più frequentemente sono raggruppati insieme.



L'ambiente fisico dell'interazione

Anche le condizioni ambientali sono importanti:

- ◆ i controlli sono messi ad altezza comoda?
- ◆ I display sono posti in modo da non riflettere la luce delle finestre o dell'illuminazione?
- ◆ Un utente in sedia a rotelle riuscirà a raggiungere tutti i controlli?
- ◆ Un utente molto alto o molto grasso non sarà reso impacciato dai comandi troppo vicini?
- ◆ Riusciranno tutti gli utenti a vedere comodamente tutti i display?



Altri aspetti ergonomici

Aspetti connessi con la salute

- ◆ Lavorare con i computer non è intrinsecamente pericoloso, ma a lungo andare possono sorgere dei problemi:
- ◆ Sforzi fisici, temperatura, luce, rumore possono avere alla lunga effetti nocivi sul nostro corpo. In particolare esistono malattie alle mani e agli occhi caratteristiche dell'uso dei computer.

Uso dei colori

- ◆ Non solo le nostre percezioni sono limitate (per esempio nel numero di colori singoli identificabili), ma esistono molte variazioni individuali.
- ◆ Moltissime persone hanno difficoltà a distinguere colori agli estremi della gamma (ad es., blu e nero), e molte hanno altri tipi di deficienze (ad es., daltonismo).
- ◆ E' opportuno dunque non usare MAI i colori come unica differenziazione e MAI in contrasto con le aspettative culturali locali.
- ◆ Un trucco per verificare la leggibilità dei propri design per persone con problemi di colori è visualizzarli su uno schermo a toni di grigio.



Il design del dialogo

L'interazione può essere vista come un dialogo tra utente e computer.

La scelta dello stile di interazione ha profondi effetti sulla natura del dialogo e, di conseguenza, sull'efficacia dell'interazione.

Sono stati identificati 6 stili di interazione primari:

- ◆ Command entry
- ◆ Menu e navigazione
- ◆ Domanda/risposta
- ◆ Form-fill e spreadsheet
- ◆ Linguaggio naturale
- ◆ Manipolazione diretta



Command entry

- Consiste nel comandare direttamente il computer tramite comandi basati su parole intere, abbreviazioni, caratteri o tasti funzione.
- E' stata la prima forma di interazione con il computer, ed è ancora molto diffusa.
- Spesso è l'unico modo di comandare un sistema (es. Unix shell), a volte è complementare ad un sistema basato su menu.

PREGI

- Flessibile e potente
- Favorisce l'iniziativa dell'utente
- Favorisce la creazione di script e macro

DIFETTI

- Apprendimento lungo
- Difficile memorizzazione
- Guidato dalla sintassi
- Scarsa tolleranza agli errori

Attira ed è adatto ai *power users*.



Menu e navigazione

- Le opzioni a disposizione dell'utente sono di volta in volta mostrate a schermo, occupandone gran parte.
- Poiché le opzioni spesso non stanno tutte sullo schermo, richiede l'adozione di meccanismi di organizzazione che nascondono set di opzioni (menu gerarchici).
- Un corretto matching con le attività dell'utente può aiutare. Ogni altra organizzazione porta a confusions e difficoltà di apprendimento.

PREGI

- Apprendimento breve
- Poche azioni richieste (es. tasti)
- Struttura i task dell'utente
- Facile gestione degli errori

DIFETTI

- Poco adatto a sistemi complessi
- Occupa schermo
- Struttura i task dell'utente
- Rallenta i power users

Adatto per compiti semplici e strutturati.



Domanda/risposta

- All'utente viene posta una serie di domande (perlopiù con risposte sì/no, codici, selezione da liste, ecc.) ed è condotto passo passo attraverso il task.
- Il sistema è in controllo dell'interazione, e a volte non permette all'utente di variare rispetto alla sequenza dei comandi.
- Adatto per compiti dalla struttura ben nota e lineare (es. bancomat)

PREGI

- Apprendimento inesistente
- Facile gestione degli errori
- Poche azioni richieste

DIFETTI

- Adatto a sistemi molto semplici
- Controlla l'iniziativa dell'utente
- Biforcazioni di task, anche molto semplici, sono irreversibili.

Adatto per compiti semplicissimi.



Form-fill e spreadsheet

- Per l'inserimento e la ricerca di dati, è utile organizzare lo schermo come se fosse un modulo cartaceo (form).
- Ogni dato da inserire ha una sua posizione sullo schermo (campo), e il passaggio da un campo all'altro avviene attraverso sistemi noti.
- L'uso e la correzione di errori è facile, tuttavia l'applicabilità è limitata
- Gli spreadsheet generalizzano questo tipo di interfaccia.

PREGI

- Apprendimento modesto e di tecniche generali
- Semplifica l'inserimento dati
- Buona gestione degli errori
- Facile da realizzare

DIFETTI

- Poco adatta per ogni compito oltre all'inserimento dati
- Occupa schermo
- Limita i task dell'utente

Adatto per inserimento dati.



Linguaggio naturale

- La comprensione del linguaggio naturale è molto desiderabile, ma è molto difficile per via dell'ambiguità intrinseca nel linguaggio.
- Essa può avvenire via voce o tramite una tastiera, ma non va confusa con il riconoscimento del parlato.
- Un sistema generale è attualmente al di fuori della nostra portata, ma esistono sistemi su domini limitati. Diventa tuttavia difficile tirare una linea tra questi sistemi e sistemi a command entry

PREGI

- Non c'è apprendimento
- E' naturale e immediato

DIFETTI

- Non esistono di tipo generale
- Possono richiedere molte azioni
- Richiedono spesso dialogo di chiarificazione
 - Non sono predicibili.

Adatto per compiti specifici.



Manipolazione diretta (1)

- I sistemi a manipolazione diretta permettono l'interazione immediata, fisica con gli oggetti dell'interfaccia.
- Essa richiede un'intelligente rappresentazione visuale dei concetti del dominio di interazione, e la possibilità di identificare oggetti ed azioni da compiersi.
- L'uso della tastiera e scelta di comandi sono sostituiti (o integrati) da attività motorie con l'ausilio di meccanismi di puntamento.

PREGI

- Presenta visivamente i task
- Facile da apprendere e ricordare
- Permette l'esplorazione
- Facile gestione degli errori
- Dà soddisfazione soggettiva

DIFETTI

- Difficile da programmare
- Richiede display grafici e sistemi di puntamento.
- Richiede una rappresentazione visuale adatta (metafore???)

Adatto per compiti semplici e strutturati.



Manipolazione diretta (2)

I sistemi a manipolazione diretta hanno le seguenti caratteristiche:

- ◆ Visibilità degli oggetti di interesse
- ◆ Azioni rapide, reversibili, incrementali
- ◆ Manipolazione motoria degli oggetti di interesse.

Tra i sistemi a manipolazione diretta ricordiamo:

- ◆ Interfacce WIMP (Window, Icon, Menu, Pointers): MacOS, Windows, X-Windows
- ◆ Interfacce point-and-click: WWW browsers
- ◆ Interfacce tridimensionali: VRML, ecc.



Manipolazione diretta (3)

Al cuore della manipolazione diretta c'è il problema di collegare in maniera diretta azioni e comandi dell'utente agli oggetti dell'interfaccia. Questa maniera si dice *directness*.

Distinguiamo due tipi di *directness*:

- ◆ *Directness semantica*: c'è un rapporto diretto tra quello che l'utente vuole fare (task) e quello che il sistema permette (function)? O sono necessari dei workaround? C'è un ovvio discorso di *affordance*.
- ◆ *Directness articolatoria*: C'è un rapporto diretto tra la funzione ed il comando che la attiva? I comandi sono pensati in modo da permettere un associazione intuitiva con il loro effetto? C'è un ovvio discorso di mapping.



Il design dello schermo

- Lo schermo (grafico o a caratteri, b/n o a colori, ecc.) è il meccanismo principe di output dell'informatica corrente.
- A seconda di quali caratteristiche abbiamo a disposizione, abbiamo diverse funzionalità di interfaccia che possono essere attivate.
- Esistono tuttavia delle regole generali da rispettare:
 - ◆ Come presentare e inserire informazione
 - ◆ Come fornire indizi sulle attività possibili
 - ◆ Estetica ed utilità
 - ◆ Localizzazione ed internazionalizzazione



Presentare ed inserire informazioni

Cosa bisogna mostrare? Testo, numeri, immagini, schemi, mappe, tavole, record, ecc.

Con che device? Per quale scopo?

A come presentare le informazioni dedicheremo una lezione intera, ma qui vale la pena ricordare:

- ◆ Fornire molte rappresentazioni diverse delle stesse informazioni permette di non sbagliare mai!
- ◆ Allineamenti e raggruppamenti sono importanti per dare indizi d'uso, appartenenza e rilevanza ai vari elementi dell'interfaccia
- ◆ L'uso dei colori va limitato al massimo, per sobrietà e generalità.
- ◆ Seguire le **8 regole d'oro del dialogo**, del **data display** e del **data entry** proposte da Ben Shneiderman



Le 8 regole d'oro del dialogo (1)

1 Cercare la coerenza

- ◆ interna: sintattica e semantica
- ◆ esterna: con le altre applicazioni e con il mondo reale

2 Fornire feedback informativo

- ◆ proporzionale all'importanza ed al ruolo dell'azione

3 Disegnare dialoghi che portino ad una chiusura

- ◆ I gruppi di azioni debbono avere un inizio, un centro ed una fine
- ◆ Fornire un grado di soddisfazione da raggiungimento dello scopo
- ◆ Permettere l'abbandono di strategie contingenti

4 Fornire una strategia di gestione dell'errore semplice



Le 8 regole d'oro del dialogo (2)

5 Fornire reversibilità alle azioni

6 Permettere agli utenti esperti delle scorciatoie

7 Garantire all'utente il senso del controllo

- ◆ evitare mancanze di causalità (es. arbitrarietà nella sequenza dei comandi)
- ◆ Rendere l'utente "iniziatore" piuttosto che "risponditore"

8 Ridurre il carico della short-term memory

- ◆ Mantenere un display semplice, informativo, comprensibile



Regole del data display (1)

1 Coerenza del display

- ◆ terminologia, abbreviazioni, formati, azioni simili siano standardizzate

2 Efficiente assimilizzazione dell'informazione

- ◆ Formati familiari all'utente, e connessi con il task da svolgere

3 Carico di memoria minimo

- ◆ Gli utenti debbono memorizzare il meno possibile da uno schermo al successivo. Il completamento di un task deve richiedere poche azioni e pochi cambi di contesto. Aiuti ed etichette debbono aiutare a mantenere il contesto.



Regole del data display (2)

Compatibilità della visualizzazione con l'inserimento

- ◆ Il formato dei dati usati per l'inserimento deve essere simile o facilmente riconducibile al formato utilizzato per la visualizzazione

Flessibilità nel controllo del display

- ◆ Gli utenti debbono ottenere dal display le informazioni nel modo più corretto per il task che si sta svolgendo



Regole del data entry

Coerenza delle transazioni per il data entry

- ◆ Le stesse azioni debbono essere usate per operazioni simili in luoghi diversi

Azioni di inserimento minimali per l'utente

- ◆ Meno azioni = maggiore velocità e minori possibilità di errore.

Carico di memoria minimo

- ◆ Non deve essere necessario per l'utente ricordare codici o sintassi complesse per l'esecuzione di task

Compatibilità del data entry con il data display

Flessibilità e controllo utente nel data entry

- ◆ La sequenza degli inserimenti, e i formati dati accettabili, debbono essere flessibili ma non ambigui.



Regole pratiche

- Consistenza nelle etichette e nelle convenzioni grafiche, standardizzazione delle abbreviazioni, coerenza nei formati
- Visualizzare solo i dati utili all'utente, visualizzazione del numero della pagina nei display a schermi multipli
- Presentare le informazioni graficamente laddove possibile (larghezza delle linee, tacche su termometri, e altre tecniche grafiche per ridurre il bisogno di leggere ed interpretare dati numerici)
- Mostrare valori in forma numerica solo quando l'esattezza del numero è importante
- Utilizzare monitor ad alta risoluzione e sfruttarli per la qualità del display
- Progettare per un uso monocromatico ed aggiungere i colori con giudizio dove aiutino l'utente
- Coinvolgere l'utente nella progettazione del display



Fornire indizi

- Un'aspetto importante nell'interazione è la continua ricerca da parte degli esseri umani di esplorare, provare nuove strade, fare esperimenti.
- Inoltre la caratteristica naturale degli umani di razionalizzare e fornire spiegazioni (anche antropomorfizzando le reazioni automatiche di un sistema) vanno considerate.
- Alcuni elementi dell'interfaccia sono passivi, altri permettono inserimenti, altri lo richiedono.
- Affordance e mapping sono la base per le nostre esplorazioni, e standard e linee guida (di piattaforma o aziendali) aiutano a fornire affordance e mapping noti.
- Ad esempio, cliccare su un'icona è naturale per utenti in misura della loro esperienza di computer, non delle loro esperienze di vita reale.



Altri aspetti

Estetica ed utilità

- ◆ Un bella interfaccia non è necessariamente una **buona** interfaccia.
- ◆ A volte bellezza ed utilità possono essere in contrasto.
- ◆ Tuttavia le regole di bel layout possono fornire preziosi indizi di usabilità.

Localizzazione ed internazionalizzazione

- ◆ Localizzare o internazionalizzare del software non si limita a tradurre le voci dei menu o dei manuali.
- ◆ Ad esempio, allineamento e layout si basano su sistemi di scrittura da sinistra a destra, dall'alto al basso.
- ◆ Ad esempio, molte icone o l'uso culturale di colori può essere molto diverso da cultura a cultura.



Conclusioni

Oggi abbiamo parlato di

- ◆ Modelli standard di interazione
- ◆ Il modello ACM
- ◆ Stili primari di interazione
- ◆ Le 8 regole d'oro di Shneiderman
- ◆ Altre regole per data display e data entry



Riferimenti

Preece et alii, HCI, cap. 13

Altri testi

A. Dix, J. Finlay, G. Abowd, R. Beale, *Human Computer Interaction*, Prentice Hall, Cap. 3

B. Shneidermann, *Designing the User Interface*, 3rd Edition, Addison Wesley, Cap. 2

T. Hewett, R. Baecker ed altri. *ACM SIGCHI Curricula for Human-Computer Interaction*, Report of the ACM SIGCHI Curriculum Development Group, ACM, 1992

