



UNIVERSITÀ DEGLI STUDI DI SALERNO  
**DIPARTIMENTO DI INFORMATICA**

# LEZIONE

**I pattern come prezioso patrimonio di conoscenze**

***L'adozione di design pattern nell'HCI***

Anno Accademico 2022/2023

# COME POSSIAMO CODIFICARE LA CONOSCENZA DEL DESIGN?

- Un design efficace e flessibile è difficile da ottenere al primo colpo.
- Eppure i progettisti esperti realizzano buoni progetti
- I progettisti principianti di solito sono sopraffatti da tutte le opzioni di design disponibili.
- I progettisti esperti evidentemente sanno qualcosa che quelli inesperti non sanno, che cos'è?

# COME I PROGETTISTI ESPERTI RISOLVONO UN PROBLEMA

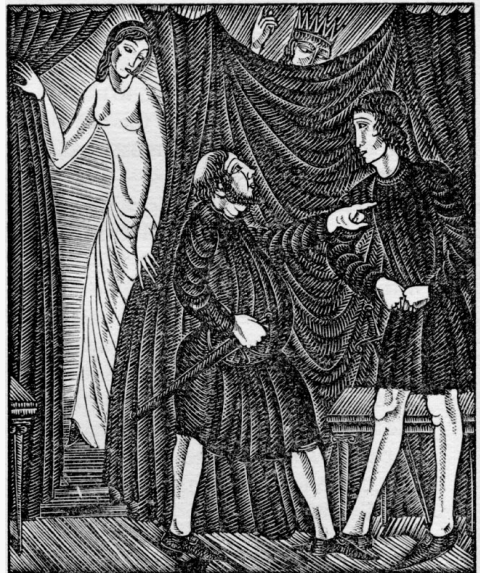
- I progettisti esperti di solito *non* risolvono tutti i problemi partendo da zero, ma *riusano* soluzioni che hanno funzionato per loro in passato.
- Quando trovano una buona soluzione, la usano ancora e ancora.
- Tale esperienza fa parte di ciò che li rende esperti.
- Questi tipi di esperienze possono essere registrati come *design pattern*

- "I buoni artisti prendono in prestito (da altri artisti), ma i grandi artisti rubano!" -Pablo Picasso
- Il design visivo avvincente richiede pratica ed esperienza, una parte naturale della quale è lo studio e la critica del lavoro altrui



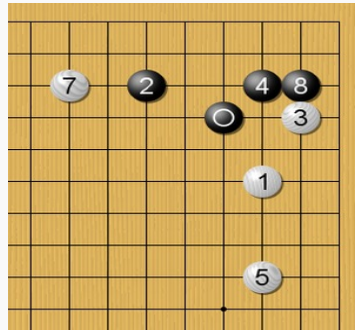
# I ROMANZIERI USANO PATTERN

- Romanzieri e drammaturghi raramente progettano le loro trame da zero.
- Seguono schemi come "Tragically Flawed Hero" (Macbeth, Hamlet, ecc.) O "The Romantic Novel" (innumerevoli romanzi rosa)

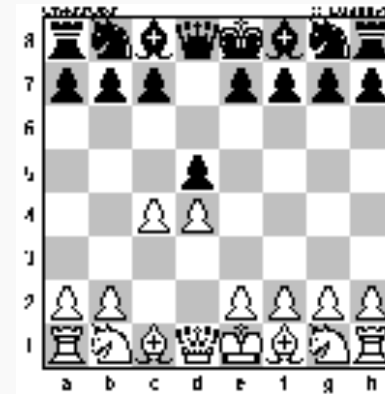


# I GIOCATORI USANO I PATTERN

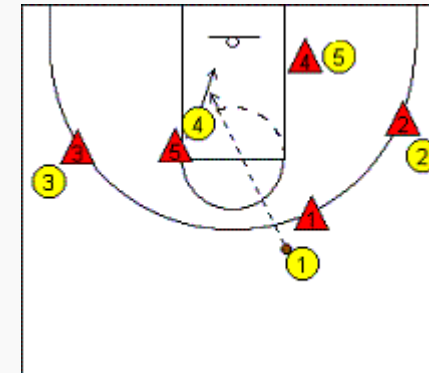
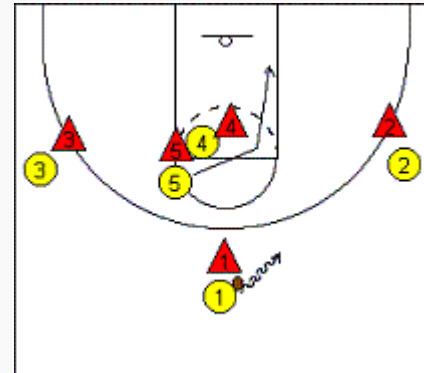
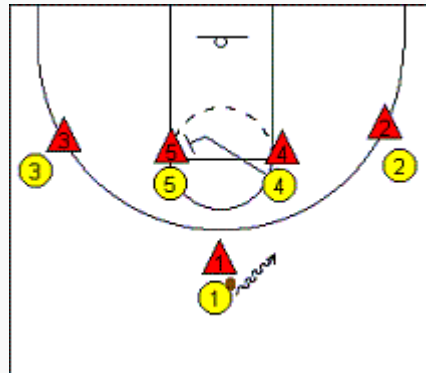
- I giocatori di scacchi, i giocatori di Go, i giocatori di basket si affidano tutti a "schemi"



A star-point Joseki in Go



The Queen's Gambit



1-4 offense Pick-n-roll from <http://www.coachesclipboard.net>

# I DESIGN PATTERN VS. LE LINEE GUIDA DI DESIGN

- Le linee guida di design possono essere astratte o concrete
  - ❖ Le linee guida astratte di solito non suggeriscono come risolvere un problema
  - ❖ Le linee guida concrete sono di solito troppo 'cucite addosso' a una specifica interfaccia
- Le linee guida di solito assumono una validità assoluta mentre i design pattern enfatizzano l'efficacia in un particolare contesto
- Le linee guida sono più utili per descrivere i requisiti laddove i pattern sono strumenti utili per coloro che devono tradurre i requisiti in soluzioni software specifiche.

# Design Pattern

Una breve storia ...



# Design patterns nell'Architettura

Opera fondamentale di Christopher Alexander's et al. nel 1977

- pattern, definiti come regole che mettono in relazione **un problema comune ricorrente in un determinato contesto** con la sua soluzione
- un potente strumento offerto agli architetti per adattare soluzioni di progettazione appropriate, **sfruttando un patrimonio di conoscenze** ed esperienze derivate dall'architettura tradizionale
  - ❖ esempio: *luce su due lati di ogni stanza*

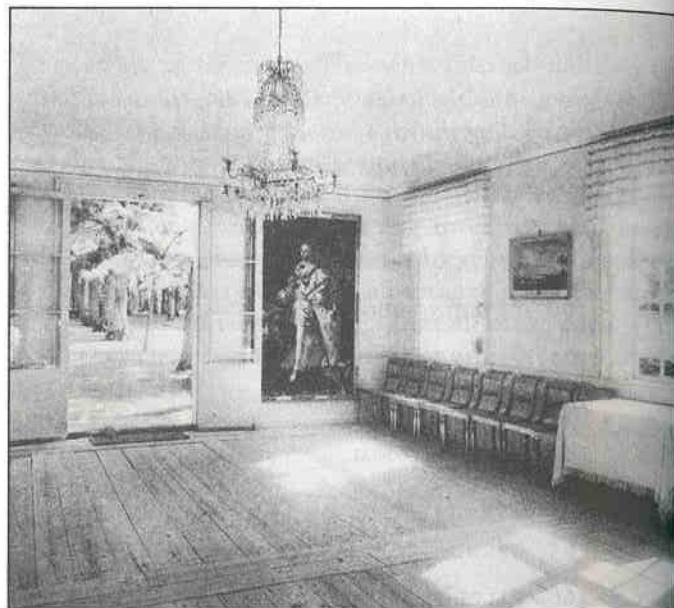
Dalla metà degli anni '90, l'idea è stata adottata nel campo dell'ingegneria del software [Gamma95]

- pattern come un notevole corpus di conoscenze che catturano le migliori pratiche di progettazione del sistema
- pattern come conoscenza di riferimento in altre fasi del ciclo di vita del software, ad esempio test del software, gestione del progetto e architettura del software

## *La struttura dei pattern di Alexander*

- NOME (di solito descrive l'effetto dell'uso del pattern)
- UNA FOTOGRAFIA che mostra un esempio archetipico del pattern in uso
- UN PARAGRAFO INTRODUTTIVO che colloca il pattern nel contesto di altri pattern di più ampia scala
- IL TITOLO un incapsulamento del problema (una o due frasi)
- IL CORPO del problema (questo può essere lungo molti paragrafi)
- LA SOLUZIONE il cuore del pattern, sempre indicato sotto forma di un'istruzione
- UN DIAGRAMMA che mostra la soluzione sotto forma di diagramma
- UN PARAGRAFO DI CHIUSURA mostra come questo pattern è consistente con altri pattern più piccoli

## 159 LIGHT ON TWO SIDES OF EVERY ROOM\*\*



... once the building's major rooms are in position, we have to fix its actual shape: and this we do essentially with the position of the edge. The edge has got its rough position already from the overall form of the building—WINGS OF LIGHT (107), POSITIVE OUTDOOR SPACE (106), LONG THIN HOUSE (109), CASCADE OF ROOFS (116). This pattern now completes the work of WINGS OF LIGHT (107), by placing each individual room exactly where it needs to be to get the light. It forms the exact line of the building edge, according to the position of these individual rooms. The next pattern starts to shape the edge.



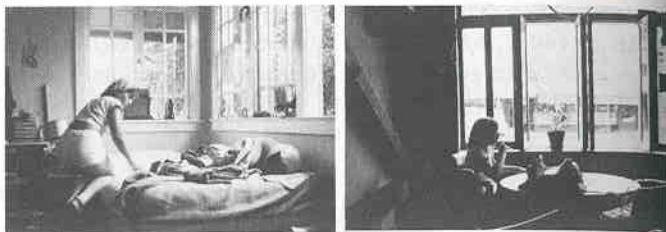
When they have a choice, people will always gravitate to those rooms which have light on two sides, and leave the rooms which are lit only from one side unused and empty.

This pattern, perhaps more than any other single pattern, determines the success or failure of a room. The arrangement of daylight in a room, and the presence of windows on two sides, is fundamental. If you build a room with light on one side only, you can be almost certain that you are wasting your money. People will stay out of that room if they can possibly avoid it. Of course, if all the rooms are lit from one side only, people will have to use them. But we can be fairly sure that they are subtly uncomfortable there, always wishing they weren't there, wanting to leave—just because we are so sure of what people do when they do have the choice.

Our experiments on this matter have been rather informal and drawn out over several years. We have been aware of the idea for some time—as have many builders. (We have even heard that “light on two sides” was a tenet of the old Beaux Arts design tradition.) In any case, our experiments were simple: over and over again, in one building after another, wherever we happened to find ourselves, we would check to see if the pattern held. Were people in fact avoiding rooms lit only on one side, preferring the two-sided rooms—what did they think about it?



We have gone through this with our friends, in offices, in many homes—and overwhelmingly the two-sided pattern seems significant. People are aware, or half-aware of the pattern—they understand exactly what we mean.



*With light on two sides . . . and without*

If this evidence seems too haphazard, please try these observations yourself. Bear the pattern in mind, and examine all the buildings you come across in your daily life. We believe that you will find, as we have done, that those rooms you intuitively recognize as pleasant, friendly rooms have the pattern; and those you intuitively reject as unfriendly, unpleasant, are the ones which do not have the pattern. In short, this one pattern alone, is able to distinguish good rooms from unpleasant ones.

The importance of this pattern lies partly in the social atmosphere it creates in the room. Rooms lit on two sides, with natural light, create less glare around people and objects; this lets us see things more intricately; and most important, it allows us to read in detail the minute expressions that flash across people's faces, the motion of their hands . . . and thereby understand, more clearly, the meaning they are after. *The light on two sides allows people to understand each other.*

In a room lit on only one side, the light gradient on the walls and floors inside the room is very steep, so that the part furthest from the window is uncomfortably dark, compared with the part near the window. Even worse, since there is little reflected light on the room's inner surfaces, the interior wall immediately next to the window is usually dark, creating discomfort and glare against this light. *In rooms lit on one side, the glare which sur-*

*rounds people's faces prevents people from understanding one another.*

Although this glare may be somewhat reduced by supplementary artificial lighting, and by well-designed window reveals, the most simple and most basic way of overcoming glare, is to give every room two windows. The light from each window illuminates the wall surfaces just inside the other window, thus reducing the contrast between those walls and the sky outside. For details and illustrations, see R. G. Hopkins, *Architectural Physics: Lighting*, London: Building Research Station, 1963, pp. 29, 103.

A supreme example of the complete neglect of this pattern is Le Corbusier's Marseilles Block apartments. Each apartment unit is very long and relatively narrow, and gets all its light from one end, the narrow end. The rooms are very bright just at the windows and dark everywhere else. And, as a result, the glare created by the light-dark contrast around the windows is very disturbing.

In a small building, it is easy to give every room light on two sides: one room in each of the four corners of a house does it automatically.

In a slightly larger building, it is necessary to wrinkle the edge, turn corners, to get the same effect. Juxtaposition of large rooms and small, helps also.



*Wrinkle the edge.*

In an even larger building, it may be necessary to build in some sort of systematic widening in the plan or to convolute the edge still further, to get light on two sides for every room.

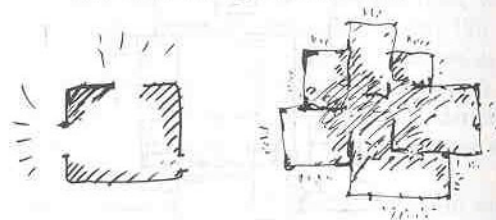
But of course, no matter how clever we are with the plan, no matter how carefully we convolute the building edge, sometimes it is just impossible. In these cases, the rooms can get the effect of light on two sides under two conditions. They can get it, if the room is very shallow—not more than about eight feet deep—with at least two windows side by side. The light bounces off the back wall, and bounces sideways between the two windows, so that the light still has the glare-free character of light on two sides.

And finally, if a room simply has to be more than eight feet deep, but cannot have light from two sides—then the problem can be solved by making the ceiling very high, by painting the walls very white, and by putting great high windows in the wall, set into very deep reveals, deep enough to offset the glare. Elizabethan dining halls and living rooms in Georgian mansions were often built like this. Remember, though, that it is very hard to make it work.

Therefore:

Locate each room so that it has outdoor space outside it on at least two sides, and then place windows in these outdoor walls so that natural light falls into every room from more than one direction.

each room has light on two sides



\*\*\*

Don't let this pattern make your plans too wild—otherwise you will destroy the simplicity of POSITIVE OUTDOOR SPACE (106), and you will have a terrible time roofing the building—ROOF

LAYOUT (209). Remember that it is possible to keep the essence of the pattern with windows on one side, if the room is unusually high, if it is shallow compared with the length of the window wall, the windows large, the walls of the room white, and massive deep reveals on the windows to make quite certain that the big windows, bright against the sky, do not create glare.

Place the individual windows to look onto something beautiful—WINDOWS OVERLOOKING LIFE (192), NATURAL DOORS AND WINDOWS (221); and make one of the windows in the room a special one, so that a place gathers itself around it—WINDOW PLACE (180). Use DEEP REVEALS (223) and FILTERED LIGHT (238). . . .

# Christopher Alexander

## 115 COURTYARDS WHICH LIVE\*\*



The courtyards built in modern buildings are very often dead. They are intended to be private open spaces for people to use – but they end up unused, full of gravel and abstract sculptures

There seem to be three distinct ways in which these courtyards fail.

1. There is too little ambiguity between indoors and outdoors...
2. There are not enough doors into the courtyard...
3. They are too enclosed...



Therefore:

Place every courtyard in such a way that there is a view out of it to some larger open space; place it so that at least two or three doors open from the building into it and so that the natural paths which connect these doors pass across the courtyard. And at one edge, beside a door, make a roofed veranda or a porch, which is continuous with both the inside and the courtyard.

Build the porch according to the patterns for ARCADE (119), GALLERY SURROUND (166), and SIX-FOOT BALCONY (167)...



# *La Natura dei Design Pattern*

- Non troppo generale e non troppo specifico
  - ❖ Usa una soluzione "un milione di volte, senza mai farlo allo stesso modo due volte"
- I design pattern formano un linguaggio condiviso
  - ❖ per "costruire e progettare città, quartieri, case, giardini e stanze".
  - ❖ Ex. La birreria fa parte di un centro per la vita pubblica ...
  - ❖ Ex. La birreria ha bisogno di spazi perché i gruppi siano soli ... **ALCOVES**

## *Formato dei Pattern di Alexander*

- Nome pattern
- Contesto
- forze
- Dichiarazione problema
- Soluzione
  - ❖ Schizzo della soluzione
- Altri schemi da considerare



## *Linguaggio dei Pattern*

- Alexander ha sottolineato **l'importanza dei linguaggi dei pattern** - più che semplici raccolte di pattern.
- Un linguaggio è un insieme di pattern che riempiono uno spazio di progettazione e sono scelti per completarsi a vicenda.
- Le *forze* in ogni pattern possono spiegare le relazioni con altri pattern.

## *Molti pattern formano un linguaggio di pattern*

- I pattern in un determinato dominio dovrebbero essere organizzati in una struttura logica e naturalmente intuitiva. Ciascun pattern dovrebbe indicare la sua relazione con altri pattern e con l'intero linguaggio.
  - ❖ Notare e nominare i problemi comuni in un campo di interesse
  - ❖ Descrivere le caratteristiche chiave di soluzioni efficaci per raggiungere alcuni degli obiettivi dichiarati
  - ❖ Aiutare il progettista a passare da un problema all'altro in modo logico
  - ❖ Consentire molti percorsi diversi lungo il processo di progettazione

# Design Pattern

## dall'architettura all'HCI

Alcuni anni dopo, i pattern hanno raggiunto la comunità HCI:

- l'approccio del pattern più appropriato all'interfaccia utente e alla progettazione di sistemi interattivi [Tidwell97, Bayle98, Borchers2001]
- Nell'ultimo decennio sono state introdotte diverse linguaggi di pattern, come una ***lingua franca*** [Eriksson2000]:
  - ❖ i progettisti di interfacce sono in grado di condividere le loro competenze tra loro
  - ❖ un efficace processo di **design partecipativo** può essere eseguito, grazie a una documentazione chiara e formale delle scelte di progettazione
  - ❖ la maggior parte dei pattern HCI concepiti in modo da consentire agli sviluppatori non esperti di attuare i principi di usabilità anche quando mancano del dovuto background teorico

# *Forte somiglianza tra i Pattern dell'Architettura e i Pattern HCI (\*)*

## Architettura

Il manufatto che l'*architetto* disegna è qualcosa con cui i suoi *abitanti* interagiscono e vivono direttamente.

## HCI

Il manufatto che il *designer dell'interfaccia utente* crea è qualcosa con cui i suoi *utenti* interagiscono direttamente e vivono.

## SE

Il manufatto che l'*ingegnere del software* progetta è qualcosa con cui i suoi *utenti non dovrebbero interagire* direttamente e sicuramente *non vivranno mai*.

(\*) [Borchers, 2000] [http://www.hcipatterns.org/tiki-download\\_file.php?fileId=22](http://www.hcipatterns.org/tiki-download_file.php?fileId=22)

# *Le 12 tesi di Borchers sui Design Pattern*

1. L'architettura è più vicina all'HCI che all'ingegneria del software
2. Il libro della banda dei quattro non contiene pattern
3. HCI deve derivare la sua idea di pattern dalla fonte originale
4. I linguaggi di pattern HCI non sono guide di stile, né regole d'oro, né standard
5. I linguaggi dei pattern dell'architettura devono essere estesi con la nozione di struttura temporale per gestire l'HCI
6. La struttura e i componenti dei singoli pattern devono tenere conto della **dimensione temporale**
7. I pattern dell'HCI devono fornire prove empiriche della loro validità
8. I pattern dell'HCI devono essere leggibili dagli utenti
9. I pattern dell'HCI toglieranno il potere ai progettisti HCI e lo metteranno nelle mani degli utenti
10. I pattern possono modellare molti domini di applicazioni
11. L'uso di pattern nell'architettura software, nella progettazione dell'interazione e nel dominio dell'applicazione di un progetto può migliorare la comunicazione nei team di progettazione interdisciplinare
12. L'uso di pattern nei vari domini può essere mappato verso la maggior parte delle fasi del ciclo di vita dell'ingegneria dell'usabilità

# *I Design Pattern per l'HCI*

- Un approccio per riutilizzare le conoscenze sulle soluzioni di progettazione di successo
- Hanno avuto origine in architettura: Alexander
- Un pattern è una soluzione parzialmente specificata a un problema ricorrente all'interno di un contesto specifico.
- Esempi
  - ❖ *Light on Two Sides of Every Room* (architettura)
  - ❖ *Go back to a safe place* (HCI)
- I pattern non esistono isolati, ma sono collegati ad altri pattern in linguaggi che consentono di creare design completi

## *I Design Pattern HCI (cont.)*

Un Design Pattern HCI cattura l'essenza di una soluzione di successo a un problema di usabilità ricorrente nei sistemi interattivi. Esso consiste di

- ❖ un nome (name),
- ❖ un grado di validità (ranking),
- ❖ un esempio tangibile (image),
- ❖ un contesto (context),
- ❖ una descrizione del problema (problem statement),
- ❖ esempi (examples),
- ❖ la soluzione (solution),
- ❖ uno schizzo (sketch o diagram),
- ❖ riferimenti (references to other patterns)

## *I Design Pattern HCI (cont.)*

- Caratteristiche dei pattern
  - ❖ colgono la prassi della progettazione non la teoria
  - ❖ colgono le proprietà essenziali comuni di un buon progetto
  - ❖ rappresentano la conoscenza della progettazione a vari livelli: sociale, organizzativo, concettuale e di dettagliato
  - ❖ incorporano valori e possono esprimere ciò che è umano nel progetto dell'interfaccia
  - ❖ sono intuitivi e leggibili e quindi possono essere usati per la comunicazione tra tutti gli stakeholder
  - ❖ un linguaggio di pattern dovrebbe essere generativo e perciò utile nello sviluppo di design completi



# Design = Soluzione

- Il design è trovare soluzioni
- Sfortunatamente, i designer spesso reinventano
  - ❖ Difficile sapere come sono state fatte le cose prima
  - ❖ Perché le cose sono state fatte in un certo modo
  - ❖ Come riutilizzare le soluzioni
- I pattern forniscono le informazioni necessarie per riusare soluzioni di successo adattandole alle esigenze del proprio problem space

# I Pattern di Borchers per il design dell'interazione > Immersive Display

## H13 IMMERSIVE DISPLAY \*



Figure 4.31: CAVE in the Ars Electronica Center Linz.

...you have decided to create an exhibit that several people can experience simultaneously—COOPERATIVE EXPERIENCE (H3). Now you need to find a way to design the visual output of such a system.



Typical usage scenarios of standard computer systems often involve only one human interacting with the computer at any time, and the system is only a small part of the real environment of the user. But exhibits are usually visited by groups of people, and when users interact with them, they are ready to immerse themselves into the world of the exhibit.

The CAVE installation in the Ars Electronica Center in Linz uses wall-size projections all around the visitors to immerse them into a virtual reality. Special glasses synchronize with these displays to create a three-dimensional impression.

Virtual Vienna uses a rear-projected display screen of about

CAVE: 3-D walls

Virtual Vienna:  
panorama

Personal Orchestra:  
large projection

1.6 m width, with the users standing at the same distance to the screen. This fills most of the optical viewing field when looking at the screen, and helps people to feel like they are actually standing at the place whose panorama is being displayed.

*Personal Orchestra* uses an even bigger display area of about 2.5 m width, again with a corresponding viewing distance. This conveys the impression of actually standing in front of the Vienna Philharmonic in a far better way than it would on a small computer monitor.

With these systems, this large display not only immerses the main user into the experience, it also allows several bystanders to at least observe the exhibit in action, which many may find already sufficient without becoming an active user.

Therefore:

Prefer a single exhibit with a large-scale display, with a minimum of 1.5 m in display width, over several similar stations with smaller displays, and over other output devices that shield a single user from his co-visitors, such as head-mounted displays. Design for a viewing distance that roughly equals the width of the display.



If you hide the display technology, it can become a "magic image"—INVISIBLE HARDWARE (H14). ...

# I Pattern di Borchers per il design dell'interazione > Immersive Display

## H13 IMMERSIVE DISPLAY \*



Figure 4.31: CAVE in the Ars Electronica Center Linz.

... you have decided to create an exhibit that several people can experience simultaneously—COOPERATIVE EXPERIENCE (H3). Now you need to find a way to design the visual output of such a system.



Typical usage scenarios of standard computer systems often involve only one human interacting with the computer at any time, and the system is only a small part of the real environment of the user. But exhibits are usually visited by groups of people, and when users interact with them, they are ready to immerse themselves into the world of the exhibit.

The CAVE installation in the Ars Electronica Center in Linz uses wall-size projections all around the visitors to immerse them into a virtual reality. Special glasses synchronize with these displays to create a three-dimensional impression.

Virtual Vienna uses a rear-projected display screen of about

name

ranking

picture

context

problem

forces

examples

CAVE: 3-D walls

Virtual Vienna:  
panorama

examples

1.6 m width, with the users standing at the same distance to the screen. This fills most of the optical viewing field when looking at the screen, and helps people to feel like they are actually standing at the place whose panorama is being displayed.

Personal Orchestra:  
large projection

Personal Orchestra uses an even bigger display area of about 2.5 m width, again with a corresponding viewing distance. This conveys the impression of actually standing in front of the Vienna Philharmonic in a far better way than it would on a small computer monitor.

With these systems, this large display not only immerses the main user into the experience, it also allows bystanders to at least observe the exhibit in a way that many may find already sufficient without becoming an active user.

Therefore:

Prefer a single exhibit with a large-scale display, with a minimum of 1.5 m in display width, over several similar stations with smaller displays, and over other output devices that shield a single user from his co-visitors, such as head-mounted displays. Design for a viewing distance that roughly equals the width of the display.

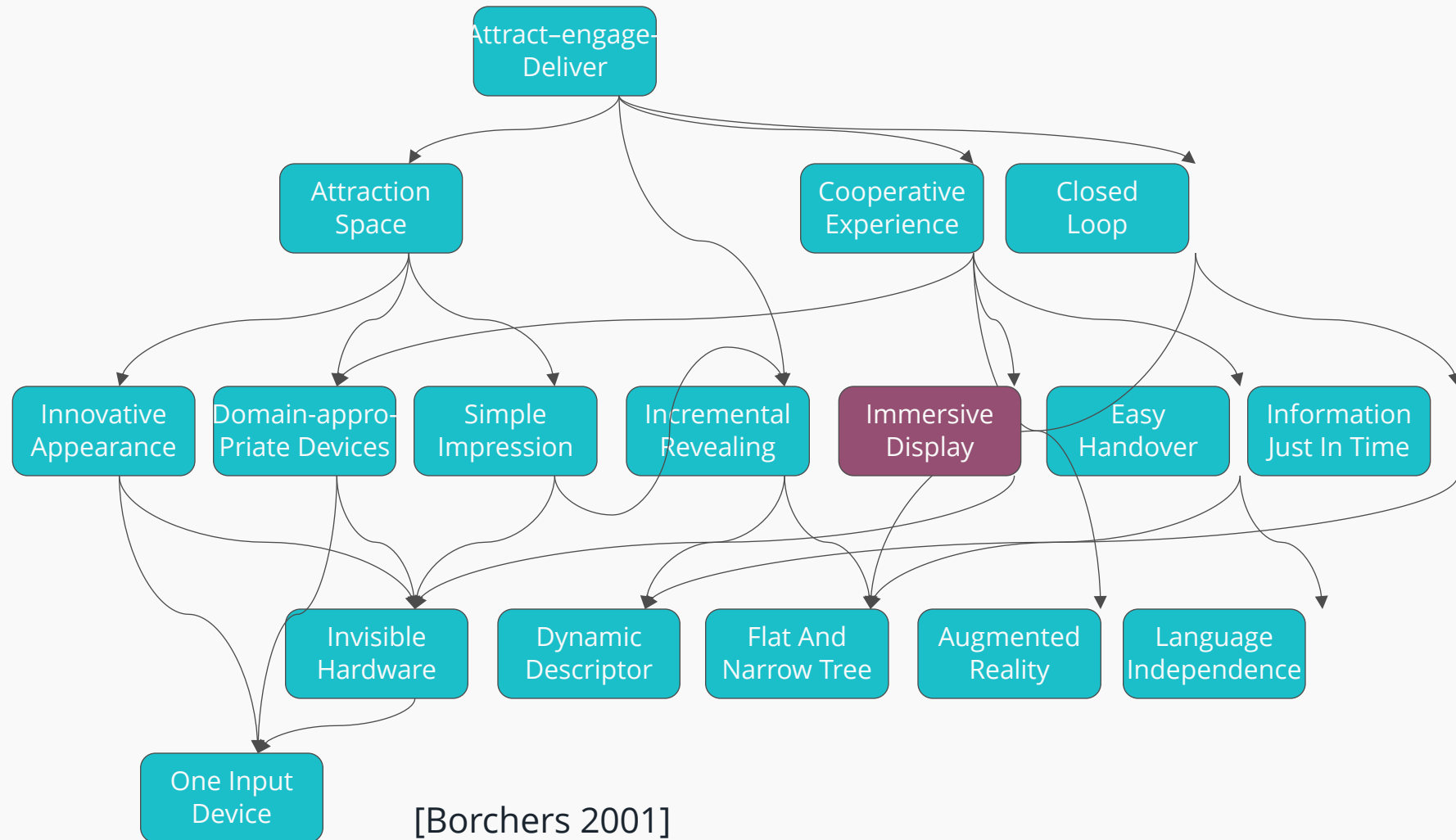
solution



diagram

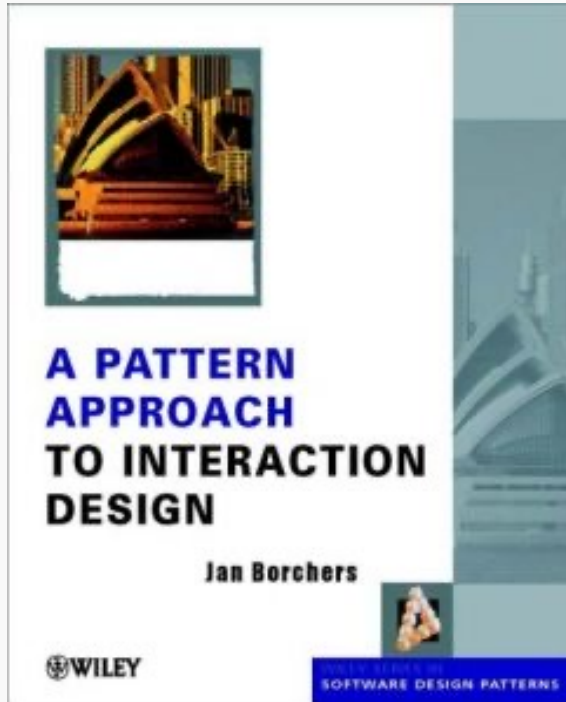
If you hide the display technology, it can become a "magic image"—INVISIBLE HARDWARE (H14). ...

# Un Linguaggio di Pattern



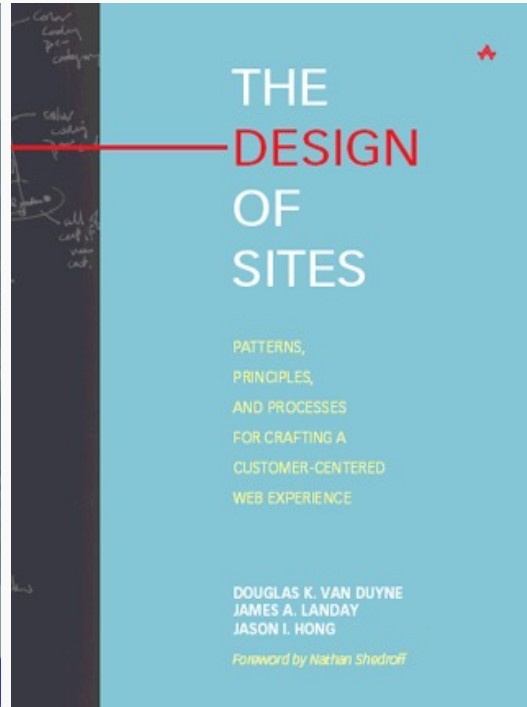


# Pattern nell' HCI/UI Design



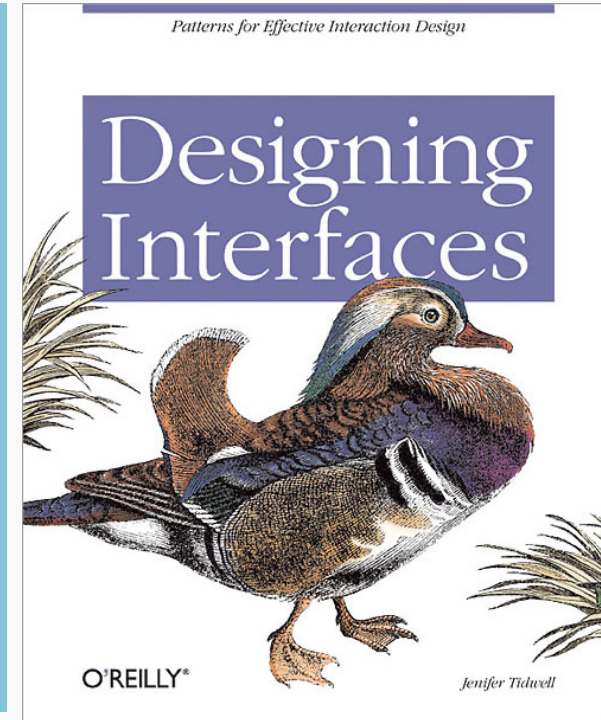
[Borchers 2001]

<http://www.hcipatterns.org/>



[Duyne et al 2002]

<http://www.visi.com/~snowfall/InteractionPatterns.html>



[Tedwell 2005]

<http://developer.yahoo.com/ypatterns/>

# *Design patterns per l'HCI*

Un Design Pattern per l'HCI cattura **l'essenza di una soluzione di successo a un problema di usabilità che spesso ricorre in sistemi interattivi**. La struttura comune di un pattern è:

- *nome*
- *classifica*
- *immagine*
- *contesto*
- *dichiarazione problema*
  - ❖ *forze*
- *esempi*
- *soluzione*
- *schizzo o diagramma*
- *riferimenti ad altri schemi*

## *UI Patterns Classificati da Tidwell*

- Architettura dell'informazione e struttura dell'applicazione
- Navigazione, indicazioni e metodo di ricerca
- Layout degli elementi della pagina
- Azioni e comandi
- Pattern di visualizzazione
- Moduli e controlli
- Costruttori ed editori
- Stile visuale ed estetica

## *Il Common Ground di Tidwell > Go back to a Safe Place*

**Contesto:** l'artefatto consente agli utenti di spostarsi attraverso gli spazi (come nel pattern *Spazi Navigabili*) o attraverso i passaggi (come nel pattern *Istruzioni Passo-Passo*), attraverso una narrazione lineare o fasi discrete; l'artefatto presenta anche uno o più checkpoint in quell'insieme di spazi.

**Problema:** in che modo l'artefatto può rendere la navigazione facile, comoda e psicologicamente sicura per l'utente?

**Esempi:**

- il pulsante "Home" in un browser Web
- per tornare all'inizio di un capitolo di un libro fisico
- la proprietà "ripristina" di alcune applicazioni software



# *Il Common Ground di Tidwell > Go back to a Safe Place*

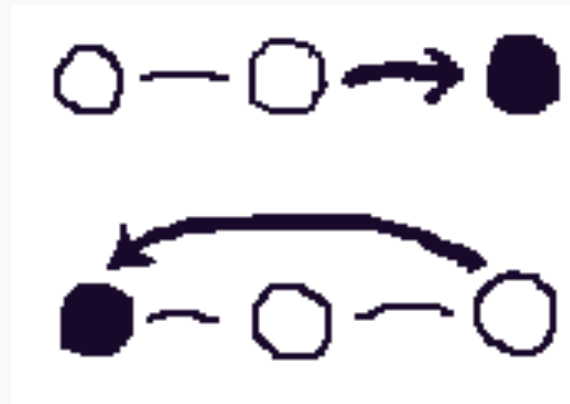
## **Forze:**

- ❖ Un utente che esplora un artefatto complesso o che tenta molte operazioni che cambiano stato, può letteralmente perdersi.
- ❖ L'utente può dimenticare dove si trovava, se smette di usare l'artefatto mentre si trova nel mezzo di qualcosa e non ci riprova per un po'.
- ❖ Se l'utente entra in uno spazio o in uno stato in cui non vuole essere, vorrà uscirne in modo sicuro e prevedibile.
- ❖ È più probabile che l'utente esplori un artefatto se gli viene assicurato che può facilmente uscire da uno stato o spazio indesiderato; questa garanzia genera un senso di sicurezza.
- ❖ Tornare indietro da un lungo percorso di navigazione può essere molto noioso.

## *Il terreno comune di Tidwell > Torna in un posto sicuro*

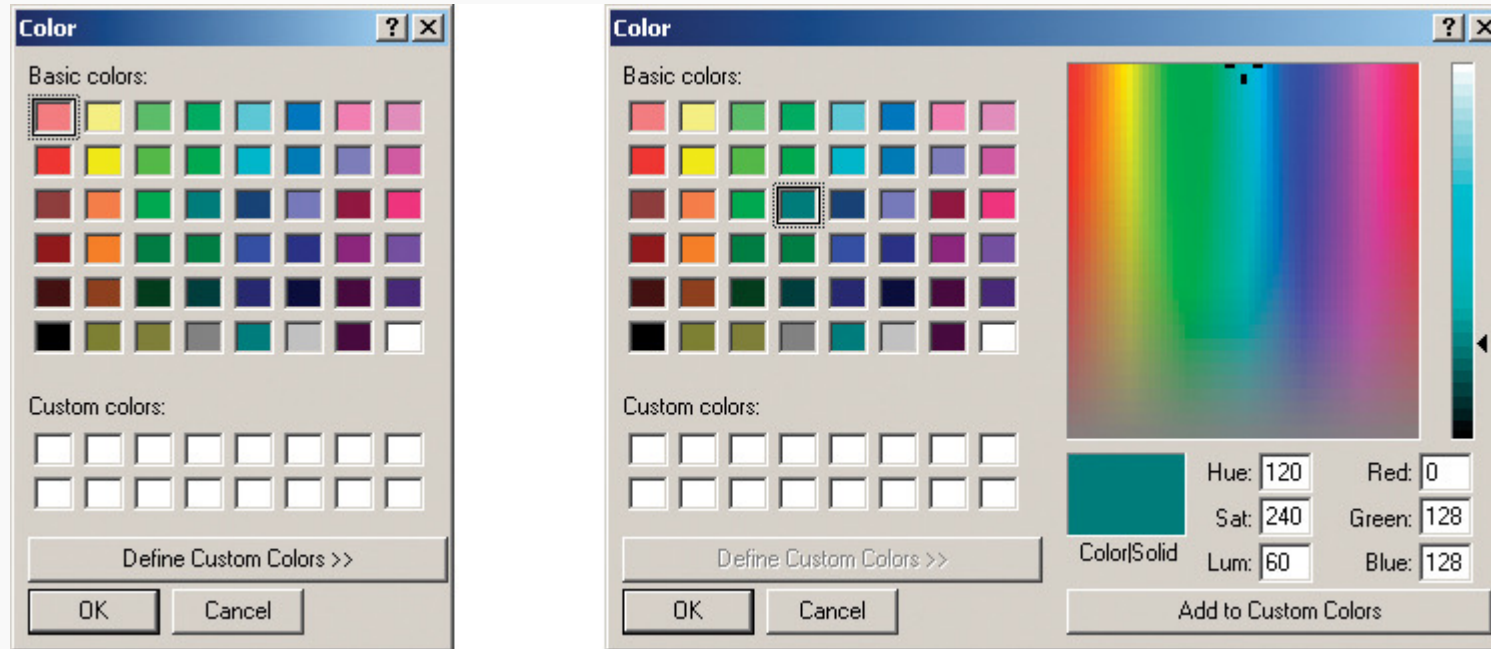
**Soluzione:** fornire un modo per tornare a un punto di controllo a scelta dell'utente

Tale checkpoint può essere una home page, un file o uno stato salvato, l'inizio logico di una sezione narrativa o di una serie di passi. Idealmente, potrebbe essere lo stato o lo spazio che un utente sceglie di dichiarare come checkpoint.



**Contesto risultante:** *Go Back One-Step* è un naturale complemento a questo pattern. Per un uso non narrativo, anche la *Interaction History* è utile, quasi al punto da rendere inutile *Go Back to a Safe Place*: può effettivamente aiutare un utente "smarrito" a capire dove si trova, ad esempio, o ricordare a un utenti che hanno interrotto un'attività dove erano e cosa hanno fatto.

# EXTRA SU RICHIESTA



- Descrizione: mostrare in anticipo i contenuti più importanti, ma nascondere il resto. Consenti all'utente di raggiungerlo con un solo, semplice gesto

# EXTRA SU RICHIESTA

## ➤ Contesto

Ci sono troppe cose da mostrare sulla pagina, ma alcune di queste non sono molto importanti. Preferiremmo avere un'interfaccia utente più semplice, ma abbiamo necessità di mettere tutto questo contenuto da qualche parte

## ➤ Soluzione

Riduciamo drasticamente l'interfaccia utente fino ai suoi elementi più comunemente usati e più importanti. Inseriamo il resto in una pagina o sezione separata.

Nascondiamo quella sezione come impostazione predefinita; sulla nuova interfaccia utente semplificata, inseriamo un pulsante o un collegamento chiaramente contrassegnato al resto, ad esempio "Altre opzioni".

Molte interfacce utente utilizzano frecce ">>", come parte dell'etichetta del collegamento o del pulsante. Altri usano "...", specialmente se il pulsante avvia una nuova finestra o pagina.

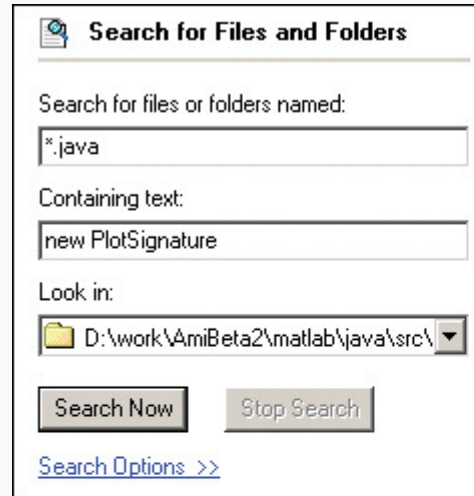
# EXTRA SU RICHIESTA

- Esempi

## **Experts warn of possible Web attack**

Seeing a rise in hacker activity that could be a prelude to a broad Internet attack, security experts urged computer users to protect their machines by installing a free patch. Internet security firms issued similar warnings, saying they've seen increased chatter in hacker discussion groups and chat rooms. "We are expecting something sooner rather than later," said Dan Ingevaldson, engineering director at Internet Security Systems in Atlanta.

[\*\*FULL STORY\*\*](#)



**Search for Files and Folders**

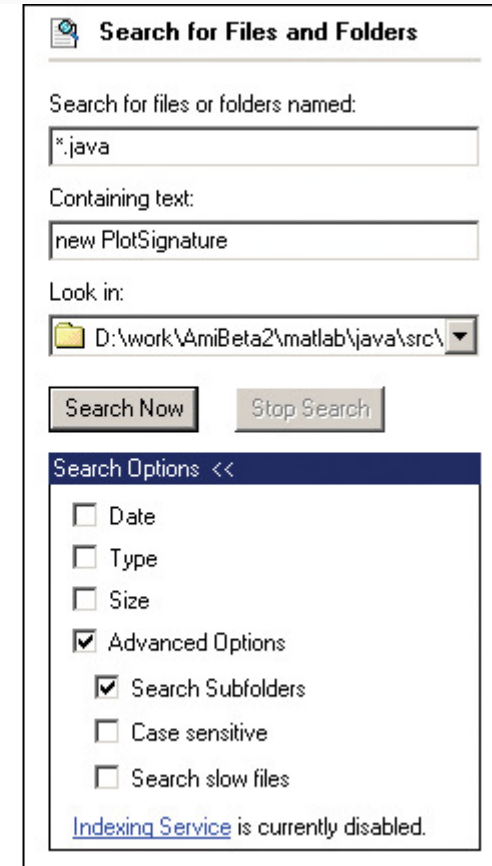
Search for files or folders named:  
\*.java

Containing text:  
new PlotSignature

Look in:  
D:\work\AmiBeta2\matlab\java\src\

[Search Options >>](#)

Buttons: Search Now, Stop Search



**Search for Files and Folders**

Search for files or folders named:  
\*.java

Containing text:  
new PlotSignature

Look in:  
D:\work\AmiBeta2\matlab\java\src\

[Search Options <<](#)

- ☐ Date
- ☐ Type
- ☐ Size
- ☒ Advanced Options
  - ☒ Search Subfolders
  - ☐ Case sensitive
  - ☐ Search slow files

[Indexing Service](#) is currently disabled.

Buttons: Search Now, Stop Search

# BREADCRUMBS



From <http://java.sun.com>

- Descrizione: su ogni pagina di una gerarchia, mostra una mappa di tutte le pagine principali, fino alla pagina corrente

# BREADCRUMBS

## ➤ Contesto

L'applicazione o il sito ha una struttura ad albero semplice, senza molte interconnessioni tra gli elementi dell'albero. Gli utenti si muovono su e giù per questo albero, sia attraverso la navigazione che la ricerca diretta.

## ➤ Soluzione

Nella parte superiore della pagina, inseriamo una riga di testo o icone che indicano il livello corrente della gerarchia. Iniziamo con il livello superiore; alla sua destra, mettiamo il livello successivo; e così via fino alla pagina corrente. Tra i livelli, inseriamo un carattere grafico o di testo, di solito una freccia rivolta a destra, per indicare lo spostamento da un livello all'altro.



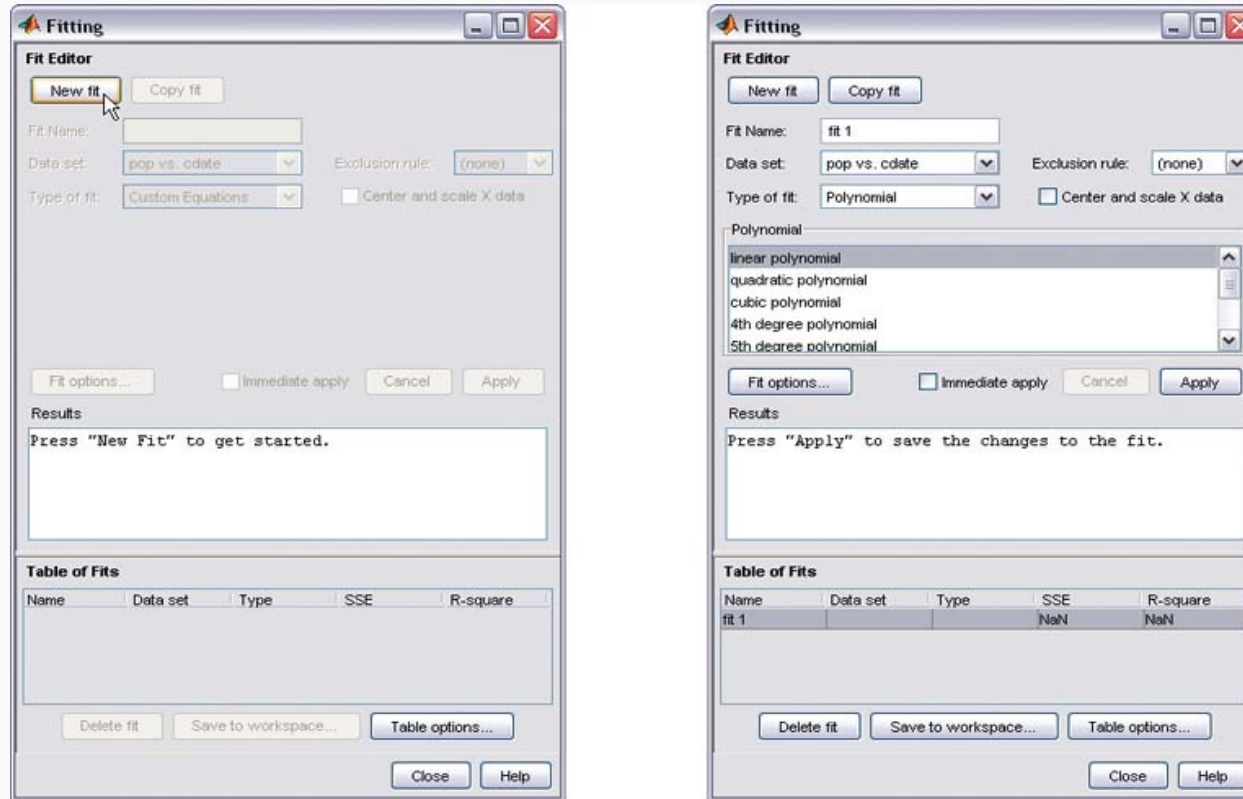
# BREADCRUMBS

- Esempi





# RESPONSIVE ENABLING



- Problema: A partire da una UI molto minimale, guidare un utente attraverso una serie di passi, mostrando qualcosa in più sulla UI al completamento di ogni passo

# RESPONSIVE ENABLING

## ➤ Contesto

L'utente dovrebbe essere accompagnato passo-passo nell'esecuzione di un task interattivo complesso, o perché è inesperto di computer o perché il task è nuovo o eseguito raramente (come in Wizard). Ma non si vuol costringere l'utente a cambiare pagina a ogni passo, piuttosto si vuole mantenere l'intera interfaccia su una sola pagina.

## Soluzione

In alcune applicazioni, la maggior parte delle azioni sull'interfaccia utente iniziano disabilitate: sono disponibili solo le azioni relative al primo passaggio dell'utente. Man mano che l'utente fa delle scelte ed esegue azioni, è necessario abilitare e mettere in gioco più elementi disabilitati

# RESPONSIVE ENABLING

- Esempi

☐ Show the date and time

View in: ☒ Menu Bar ☐ Window

View as: ☒ Digital ☐ Analog

☐ Display the time with seconds

☒ Show AM/PM

☒ Show the day of the week

☐ Flash the time separators

☐ Use a 24-hour clock

☒ Show the date and time

View in: ☒ Menu Bar ☐ Window

View as: ☒ Digital ☐ Analog

☐ Display the time with seconds

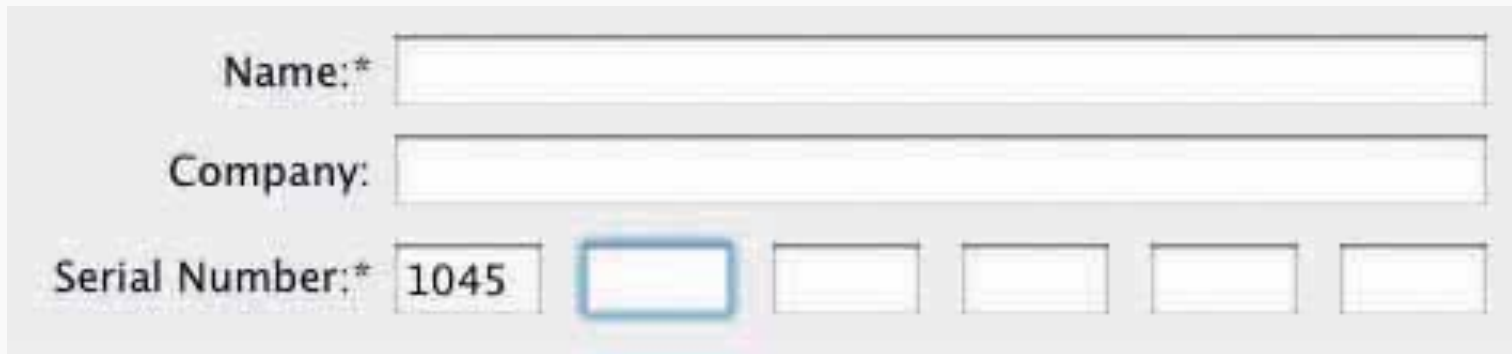
☒ Show AM/PM

☒ Show the day of the week

☐ Flash the time separators

☐ Use a 24-hour clock

# FORMATO STRUTTURATO



Name: \*

Company:

Serial Number: \*

- Descrizione: l'interfaccia richiede all'utente un tipo specifico di input di testo, formattato in un certo modo.

# FORMATO STRUTTURATO

## ➤ Contesto

Il formato di input previsto dal programma è familiare e ben definito e il programma non prevede che gli utenti debbano discostarsi dal formato che ci si aspetta. Esempi includono informazioni sulla carta di credito, numeri di telefono locali e stringhe o numeri di licenza

## ➤ Soluzione

Progettare una serie di campi di testo che riflettano il formato richiesto. Mantenere i campi di testo brevi, come indizi sulla lunghezza dell'input. Una volta che l'utente ha digitato tutte le cifre o i caratteri nel primo campo di testo, se ne dà conferma spostando automaticamente il focus di input sul campo successivo. L'utente può ancora tornare indietro e modificare nuovamente uno dei campi precedenti, ovviamente, ma ora sa quante cifre sono richieste lì

# FORMATO STRUTTURATO

- Esempi

Telephone Number:

Credit Card Number:

Date:

ISBN Number

**Date:**

**Subject:**

# DISCUSSIONE

- Qual è la differenza tra i design pattern e alcuni suggerimenti sul design dati da esperti?
- Perché è necessario il contesto quando si documenta un pattern?
- Possono i design pattern sostituire del tutto buone pratiche di progettazione?

Provate a riflettere autonomamente prima di passare alla slide successiva  
...



# DISCUSSIONE

- Qual è la differenza tra i design pattern e alcuni suggerimenti sul design dati da esperti?

*I design pattern rappresentano un patrimonio di conoscenza alla portata di tutti. Essi sono una codifica dell'esperienza fatta da esperti del design, che altri progettisti possono utilizzare per rendere il proprio prodotto usabile.*

- Perché è necessario il contesto quando si documenta un pattern?

*Il contesto è importante per comprendere il legame tra problema e soluzione descritta e per aiutare il progettista a capire se quella soluzione può essere adottata anche all'interno del proprio problem space e in che modo.*

- Possono i design pattern sostituire del tutto buone pratiche di progettazione?

*No ma possono essere un valido strumento per mettere in atto tutti quei principi che garantiscono che un prodotto sia usabile. Perciò, l'uso dei design pattern va opportunamente integrato nel processo di design centrato sull'utente*

## I design pattern nel vostro progetto

Analizzate la collezione di pattern presente sul sito <http://ui-patterns.com/> e individuare alcuni design pattern iniziali che potreste utilizzare nel vostro progetto.

Potrete riportare all'interno della relazione che consegnerete per il terzo assignment la descrizione dei pattern utilizzati.