



**Dipartimento di Informatica
Università degli Studi di Bari**

DE_VISU

Programma di ricerca (cofinanziato dal MURST, esercizio 2000)

Specifica, Progetto e Sviluppo di Sistemi Interattivi Visuali

Progettazione di Sistemi Visuali Usabili

RAPPORTO TECNICO: N.03 ANNO 2001

BA-R03

30 novembre 2001

Sommario

Questo articolo illustra una metodologia per la progettazione di sistemi visuali interattivi, considerando in modo particolare la necessità di accessibilità al sistema da parte degli utenti finali. Lo scopo della metodologia è di soddisfare tre principi: a) l'utente deve sempre capire gli effetti dell'attività del sistema rispetto all'esecuzione del suo compito; b) l'utente deve sempre avere il controllo della computazione interattiva, evitando di perdersi nello spazio virtuale; c) il sistema deve evitare e/o bloccare gli errori degli utenti mantenendosi in un insieme prevedibile di stati. Il progetto del sistema visuale è effettuato attraverso vari passi, partendo dall'analisi delle notazioni che gli utenti finali hanno sviluppato nei loro ambienti di lavoro, e producendo ad ogni passo un nuovo linguaggio visuale che deve essere verificato ma anche valutato dal punto di vista dell'usabilità. A tale scopo, la metodologia di progetto è integrata con la metodologia di valutazione dell'usabilità SUE.

Titolo ricerca unità	Progettazione di Sistemi Visuali Usabili
Codice	BA-R03
Data	30 novembre 2001
Tipo di prodotto	Rapporto tecnico
Unità responsabile	BA
Unità coinvolte	BA – BS – RM
Autori	Maria Francesca Costabile
Autore da contattare	Maria Francesca Costabile Dipartimento di Informatica Università degli Studi di Bari Via Orabona 4, 70125 Bari, Italia costabile@di.uniba.it

1. Introduzione

Nella società dell'informazione, c'è un bisogno crescente di sistemi che garantiscano accessibilità e alta qualità dell'interazione da parte di una popolazione di utenti finali sempre più ampia. Gli utenti finali sono ora essenzialmente persone non esperte di informatica, che usano i sistemi interattivi per eseguire i compiti di cui sono responsabili in un preciso contesto e in base alla loro cultura ed esperienza di lavoro. Gli utenti finali hanno necessità e conoscenze differenti e operano in contesti differenti. Di conseguenza il nostro punto di vista rispetto a "universal design" non implica che una singola interfaccia utente possa essere adatta per tutti gli utenti. Come progettisti ci sforziamo di proporre soluzioni che siano adatte agli specifici utenti di un particolare sistema.

Dall'inizio degli anni 80, la comunità di ricercatori che lavora in Linguaggi Visuali ha dedicato molto sforzo nel progettare ambienti che possano facilitare il modo di interagire delle persone con i computer. Un approccio visuale non è certamente adatto a persone con problemi di vista, ma per molte altre persone l'uso di formalismi visuali è stato un grande successo in diverse situazioni, così si pensi solo all'interfaccia visuale per lavorare con un sistema operativo proposto dalla Apple negli anni 80 e poi adottata anche da Microsoft Windows. I linguaggi visuali sono stati sfruttati con successo in ambienti di programmazione per bambini e in vari settori applicativi come ad esempio generazione di strumenti per laboratori virtuali.

Membri delle unità di Bari, Brescia e Roma hanno costituito da alcuni anni il Pictorial Computing Laboratory (PCL), con sede a Roma. Nel PCL hanno maturato esperienze nel progetto di sistemi visuali interattivi che forniscono ambienti appropriati per differenti tipi di esperti, quali medici [1] o ingegneri meccanici [2]. Questa esperienza ha portato alla proposta di una metodologia per progettare sistemi interattivi visuali, che enfatizzi la necessità che i sistemi siano accessibili da parte degli utenti finali [3]. Tale metodologia è descritta in questo articolo, dopo aver discusso l'approccio del PCL al progetto di sistemi usabili. L'integrazione di tale metodologia di progetto con la metodologia di valutazione di usabilità SUE è infine illustrata.

2. Approccio al Progetto di Sistemi Usabili

L'approccio del PCL al progetto di sistemi interattivi usabili per l'utente finale è motivato da tre principi: a) l'utente deve sempre capire le conseguenze dell'attività del sistema rispetto all'esecuzione del suo compito; b) l'utente deve sempre avere il controllo della computazione

interattiva, evitando di perdersi nello spazio virtuale; c) il sistema deve evitare e/o bloccare gli errori degli utenti mantenendosi affidabile e cioè in un insieme prevedibile di stati.

L'approccio del PCL ha generato una metodologia di progetto, presentata in questo articolo, che mira a soddisfare i tre principi descritti precedentemente sulla base delle seguenti assunzioni: a) *adeguatezza della comunicazione utente-calcolatore* [4] e *comunicabilità del sistema* [5] sono prerequisiti per l'accettazione del sistema da parte dell'utente, quindi sono dimensioni importanti per tutta l'usabilità del sistema; b) *le notazioni degli utenti* sono il modo in cui gli utenti esprimono la loro cultura, di conseguenza lo sviluppo di sistemi usabili deve capitalizzare su queste notazioni [6]; c) la *formalizzazione* è un prerequisito per verificare validità, completezza e coerenza dei risultati ottenuti progressivamente nelle fasi di sviluppo del progetto [7]; d) esistono due *semantiche*, una relativa all'utente e l'altra implementata nel sistema: la validazione del sistema è un'attività che mira a valutare la corrispondenza tra queste due semantiche; e) un uso coordinato delle attività di *validazione* e *verifica* è necessario per assicurare l'usabilità e l'affidabilità di un sistema interattivo visuale. L'attività di validazione del sistema è basata principalmente sulla valutazione dell'usabilità, stressando le dimensioni dell'adeguatezza della comunicazione e della comunicabilità. L'attività di verifica è basata su una teoria formale per la specifica ed il progetto di sistemi interattivi visuali, derivata dalla teoria delle sentenze visuali, sviluppata negli ultimi cinque anni [8].

Nell'approccio del PCL, l'interazione tra utente e calcolatore è un processo in cui l'utente ed il calcolatore comunicano materializzando e interpretando una sequenza di messaggi a istanti di tempo successivi, l'uomo usando i suoi criteri cognitivi, il calcolatore usando i criteri in esso programmati [8]. Un sistema interattivo visuale può essere considerato come un generatore di messaggi dal progettista all'utente del sistema [4].

In ogni interazione sono sempre definite implicitamente due semantiche; una interna al sistema, in cui ogni messaggio è associato al significato computazionale, come definito dal progettista e implementato nel sistema, ed una propria dell'utente che sta eseguendo il compito, e che dipende dal suo ruolo nell'esecuzione del compito, così come dalla sua cultura, esperienza, ecc. Come osservato in [4], perché il compito dell'utente possa essere eseguito con successo, occorre che un significato simile sia associato dall'utente e dal sistema ad ogni messaggio, e quindi che si raggiunga una comunicazione adeguata. Nel caso di interfacce 2D a manipolazione diretta, che è lo stile di interazione che considereremo in questo documento, i messaggi scambiati sono le immagini rappresentate sullo schermo del calcolatore, composte da testo, grafici, icone, ecc.

Gli esseri umani interpretano tali immagini riconoscendo in esse delle *strutture caratteristiche* (dette implicitamente *strutture* oppure *css* dalle iniziali delle parole inglesi

characteristic structures), cioè insiemi di pixel dell'immagine che sono riconosciute come unità percettive o funzionali. Riconoscere una struttura significa associare ad essa un significato. In generale, le persone esprimono il significato attribuito alla struttura tramite una descrizione verbale. L'identificazione della struttura è influenzata dalla similarità (o dissimilarità) con entità grafiche e costrutti tradizionalmente adottati nella comunità dell'utente.

Anche il sistema associa entità grafiche con costrutti computazionali. E' esattamente questa associazione che rende il computer capace di interpretare le azioni dell'utente (come il click su un bottone) rispetto all'immagine sul video, eventualmente attivando attività computazionali i cui risultati sono materializzati sullo schermo tramite creazione, cancellazione, oppure modificazione delle strutture caratteristiche.

Il problema di ottenere una comunicazione adeguata richiede, inoltre, che sia definita una corrispondenza precisa tra le strutture percepite dagli utenti e quelle previste dai progettisti e implementate nel sistema. In generale, questo richiede la definizione formale di disposizioni di pixel che possano essere considerate *css*, e dell'associazione tra tali strutture e il loro significato percepito o inteso. Si introduce così la nozione di *pattern caratteristico*.

3. Linguaggio Visuale di Interazione

Come abbiamo detto, una *struttura caratteristica* *cs* è un insieme di pixel identificabile sullo schermo. Il significato associato con *cs* è formalizzato come un *simbolo attribuito*, chiamato *u*, che consiste di un tipo di nome - il *simbolo*- e una tupla di proprietà - i valori degli attributi che forniscono una descrizione e una interpretazione di *cs*. L'associazione tra una *cs* e la sua descrizione *u* è espressa da due funzioni, *intcp* e *matcp*. La *funzione di interpretazione* *intcp*, associa la *cs* alla sua descrizione *u*. La *funzione di materializzazione*, *matcp* associa il simbolo attribuito *u* alla *cs*. Un *pattern caratteristico* (characteristic pattern o **cp**) è definito dalla tripla $cp = \langle cs, u, \langle intcp, matcp \rangle \rangle$.

In un'immagine *i*, si possono identificare diversi **cps**. Inoltre, tutta l'immagine *i* può essere associata a un simbolo *u_i* che sintetizza tutte le proprietà e il significato dell'immagine *i*. In ogni caso, l'insieme *d* di tutti i simboli attribuiti che appaiono nei **cps** identificati su *i* costituiscono una descrizione dell'immagine *i*. Due funzioni, *int* e *mat*, possono essere definite sulla base delle funzioni *intcp* e *matcp* nei **cps**, e la relazione tra l'immagine *i* e la sua semantica è riassunta dalla tripla $vs = \langle i, d, \langle int, mat \rangle \rangle$. *vs* è detta *sentenza visuale* o **vs**, *i* è la componente immagine e *d* la componente descrizione di *vs*.

Dal punto di vista dell'utente, ogni immagine è un documento elettronico. Eseguire un compito con il supporto di un sistema visuale interattivo è equivalente a produrre un insieme

di documenti elettronici attraverso l'esecuzione di una sequenza di azioni. In generale, ogni attività interattiva supportata dal sistema può essere ridotta alla produzione di una tale sequenza di documenti elettronici. Gli utenti identificano il sistema con la sequenza di **vss** e le immagini che appaiono sullo schermo sono le componenti immagine delle **vss**. Dal punto di vista del progettista, il sistema è specificato definendo l'insieme di tutte le sequenze ammissibili che possono essere generate durante un processo interattivo. Questo insieme costituisce un linguaggio visuale dinamico, cioè un insieme di **vss** strutturato da una relazione di preordine [9], che viene detto Linguaggio Visuale di Interazione (Interaction Visual Language o IVL). L'interfaccia è la componente del sistema significativa per l'utente, perché è quella che l'utente vede e con la quale interagisce; quindi, dal punto di vista dell'utente, il progetto del sistema visuale interattivo può essere ridotto al progetto dell'IVL.

Le **vss** nell'IVL sono composte da un numero finito di tipi di **cp**. L'insieme di tipi di **cp** è descritto rappresentando ogni tipo con un **cp** esemplare. Questo numero finito di **cps** è chiamato l'alfabeto κ dell'IVL. Una **vs** è composta da un numero finito di istanze di elementi di κ , trasformati e composti opportunamente. Nel nostro approccio, queste trasformazioni e composizioni sono definite su K in termini finiti [8].

4. Linguaggi Visuali in Ambienti di Lavoro Tradizionali

Negli ambienti di lavoro tradizionali, le persone esperte di un certo dominio applicativo (ad esempio gli ingegneri meccanici) eseguono un compito e sulla base della loro conoscenza generano documenti composti da grafici, immagini e testi [6]. In questo modo eseguono le loro attività, costruiscono nuovi prodotti e comunicano la loro conoscenza ad altri. Dal punto di vista della teoria delle sentenze visuali del PCL, le parole nel testo, le strutture nei grafici e nelle immagini sono tutte strutture caratteristiche (**css**), i documenti sono le componenti immagine di sentenze visuali (**vss**), mentre le descrizioni dei loro significati e le relazioni **int** e **mat** non sono necessariamente rese esplicite, ma sono in ogni caso presenti all'esperto nella sua attività. Una procedura di lavoro in un ambiente tradizionale è qui modellata come produzione e trasformazione di documenti, dove un primo documento contiene informazioni circa lo stato iniziale di un processo e i documenti successivi rappresentano gli stadi intermedi dell'attività dell'utente per eseguire un compito specifico, fino all'ultimo che mostra sia il documento che i risultati finali.

La sequenza dei documenti che gli esperti producono durante l'esecuzione di un compito è un risultato osservabile della loro attività. Questa sequenza di documenti cartacei è analoga alla sequenza di documenti elettronici osservati quando un utente lavora con un sistema visuale interattivo. Noi sfruttiamo questa analogia per fare un match tra le semantiche

dell'utente e del sistema, adottando la notazione usata dagli esperti nel loro ambiente di lavoro tradizionale come il nucleo (kernel) della definizione del linguaggio visuale di interazione IVL. Questo è un passo importante per soddisfare i tre principi descritti nella sezione 1, in particolare per raggiungere una comunicazione utente-calcolatore adeguata. Il progetto di un IVL basato sulla notazione dell'utente parte dall'analisi dei processi di produzione di un insieme di tali documenti, eseguita dal progettista insieme con i rappresentanti degli utenti. Insieme con le loro descrizioni, questi documenti formano quello che chiamiamo linguaggio visuale del task osservato o *observed Task Visual Language*, TVLo. L'analisi del processo di produzione dei documenti abilita a stabilire le relazioni tra i documenti stessi.

In questa visione TVLo è un insieme finito di documenti interpretati, che costituisce un campione di documenti che possono essere generati durante l'esecuzione dei compiti degli utenti, che è il vero Task Visual Language. Il progetto di un IVL richiede l'astrazione di una notazione da TVLo, e la derivazione successiva di una definizione formale intensionale di TVL. Tuttavia, quando l'attività eseguita dagli utenti è più astratta e/o la loro cultura è orientata al ragionamento formale, gli esperti spesso sviluppano una notazione visuale - e qualche volta una definizione formale intenzionale dei linguaggi visuali attraverso cui essi generano i documenti con una procedura rigorosa, anche se spesso non algoritmica. Esempi tipici sono le Reti di Petri, per le quali l'annotazione è proprio un linguaggio visuale [8]. Tra queste due situazioni estreme, diversi casi intermedi possono essere trovati studiando varie comunità di esperti. Si possono riassumere tre tipi differenti di situazioni:

1. Gli esperti producono documenti per specificare alcune situazioni o prodotti. TVLo è quindi un insieme di documenti, che non descrivono l'intero compito, ma identificano solo alcuni momenti e decisioni importanti durante l'esecuzione del documento. I documenti sono prodotti senza seguire alcuna strategia formale e le persone imparano come produrre un documento osservando le attività di colleghi con più esperienza [1].
2. Gli esperti trascrivono sistematicamente ogni passo delle loro attività, come accade nei laboratori, dove esiste un logbook, o libro mastro. I documenti del TVLo descrivono in questo caso le strategie seguite dagli esperti per eseguire i loro compiti. In alcuni campi, esistono linee guida e glossari standardizzati per la produzione del documento (per esempio disegni tecnici [2]).
3. Gli esperti sistematicamente trascrivono le loro attività, sviluppando i loro documenti secondo una notazione formale, in senso matematico. In tal caso, non solo esiste già un buon TVL – i cui documenti descrivono le strategie seguite dagli esperti nello svolgimento dei loro

compiti – ma è usata anche una notazione che definisce le strategie seguite dagli esperti, come nel caso della costruzione delle Reti di Petri [8].

Tradizionalmente, gli esperti eseguono i loro compiti senza il supporto del computer. Per progettare un sistema che aiuti gli utenti a migliorare le loro prestazioni, il progettista collabora con gli utenti lavorando nel loro ambiente di lavoro, per identificare le strategie con cui gli utenti eseguono i compiti e li organizzano in sotto-compiti (*task analysis*). Il progettista identifica anche i documenti dove queste strategie sono descritte. Tali documenti sono ordinati in base alla fase del compito durante la quale sono stati prodotti. Insieme con le loro interpretazioni, i documenti costituiscono le vss del TVLo. In più, i progettisti possono identificare la struttura ordinata imposta sulla produzione delle vss, e possono derivare un metodo per permettere solo la produzione delle sequenze ammissibili di vss.

5. Metodologia per il Progetto di Linguaggi Visuali di Interazione

Seguendo un approccio centrato sugli utenti, nel progetto di IVL partiamo dall'analisi degli utenti e dei compiti e dall'osservazione dei documenti prodotti (Figura 1). Il progettista organizza tutte le osservazioni in un modo sistematico in un'ontologia esplicita del dominio, un'organizzazione che l'utente non è sempre capace di ottenere. Spesso, da questa organizzazione si riesce a ricavare un alfabeto visuale e un insieme di regole espresse nel linguaggio utente, che costituiscono il TVLo (rettangolo in alto in Figura 1). L'adeguatezza di un tale glossario – e delle regole – rispetto ai costrutti significativi dell'utente richiede una prima fase di verifica, per dimostrare che tutte le sentenze nel TVLo possono essere prodotte e che nessuna sentenza che violi i vincoli richiesti dagli utenti possa essere creata. Inoltre, una prima fase di validazione è anche necessaria per controllare che l'alfabeto visuale e le regole possano essere capiti e gestiti dall'utente.

Dall'osservazione dei documenti prodotti e dalle attività degli utenti, astraiano la definizione del Task Visual Language (TVL). Questo processo è semplice se già da TVLo si era riusciti ad identificare un possibile alfabeto e un insieme di regole, altrimenti si cerca ora di identificare l'alfabeto visuale e le regole implicite usate dagli esperti nel costruire e trasformare i loro documenti. Questo processo può implicare la definizione di diversi tentativi di generalizzazione del TVLo prima che si raggiunga un accordo. Ciò perché è possibile che i costrutti grafici tipici impiegati dagli utenti non siano sufficientemente espressivi e non-ambigui per costruire una notazione formale. Sono quindi richiesti diversi passi per raggiungere una forma adeguata che può essere formalizzata, eliminando le ambiguità e le inconsistenze identificate.

Una volta raggiunto l'accordo su che cosa deve appartenere al TVL, è necessaria una definizione formale di questo linguaggio (rettangolo 2 di Figura 1). In [10], sono forniti i dettagli riguardo a ciò. Una verifica semi-formale è possibile controllando che ogni documento osservato possa essere prodotto con l'uso del sistema di riscrittura che specifica il linguaggio, che documenti non corretti non possono essere composti, e che sentenze legali corrispondano a passi intermedi durante la costruzione e la trasformazione del documento. La notazione formale così ottenuta deve essere valutata rispetto alle semantiche dell'utente. In particolare, è necessario verificare che gli utenti capiscano correttamente la definizione del TVL derivato e il suo uso.

Il TVL così definito è poi accresciuto per sfruttare le capacità interattive del sistema (rettangolo 3 in Figura 1). Infatti, le sequenze di **vss** del TVL descrivono lo sviluppo dell'attività degli esperti. TVL è un passo intermedio nel processo di costruzione dell'IVL. Tale processo continua arricchendo il TVL con la definizione dei **cps** attraverso cui il sistema visuale comunica all'utente lo stato dell'interazione e fornisce l'accesso alle sue funzionalità. Tuttavia, il TVL stesso può essere arricchito per sfruttare le possibilità dell'interazione e dell'animazione possibili in un computer.

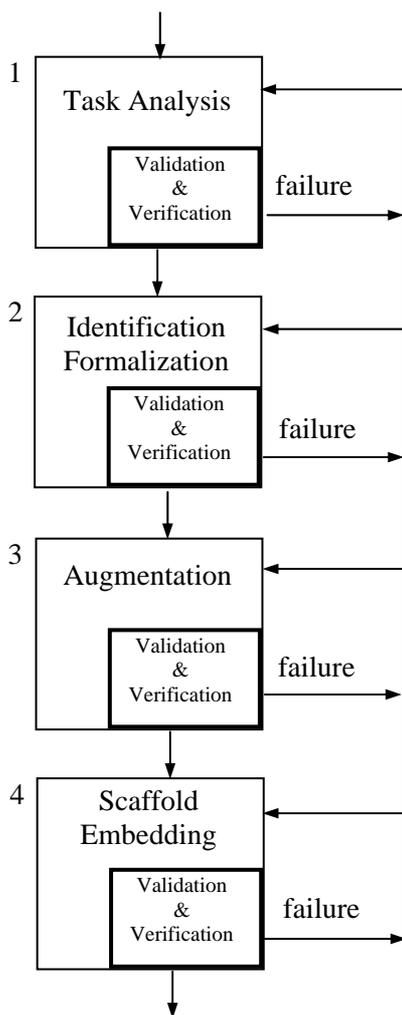


Figura 1. Passi nel progetto dell'IVL

Il progettista può anche usare meccanismi per attirare l'attenzione, meccanismi di feedback, effetti di animazione. Per esempio, per fornire feedback all'azione dell'utente di puntare a una **cs** sul video, tale **cs** è evidenziata cambiando il suo colore, generando così una nuova **cs**. La notazione dovrebbe anche essere adattata alle tecnologie digitali che generano l'immagine sul video così come alle dimensioni stabilite e limitate del video.

La validazione del TVL arricchito (ATVL) richiede un'implementazione parziale o mock-up del sistema, così che la coerenza della dinamica può essere controllata con quella attesa dall'utente. In particolare, la dinamica dell'interazione può essere simulata con mock-up e semplici prototipi per assicurare che i comportamenti dei **cps** sono comprensibili all'utente e che l'utente può accedere a tutti i dettagli necessari resi impliciti dalle notazioni sintetiche. La verifica dell'ATVL richiede che tutte le **vss** nel TVL siano ottenibili tramite ATVL.

Infine, l'Interaction Visual Language (IVL) è specificato definendo come le **vss** dell'ATVL possono essere integrate nelle sentenze visuali con supporto aggiuntivo per l'interazione, quali i **cps** che forniscono il contesto per l'orientamento dell'utente nello spazio interattivo (si vedano i concetti dei frame e scaffold in [9]) e quelli che danno accesso alle funzionalità del sistema, i vari strumenti che operano sulle stesse **vss**, ecc. La definizione dell'IVL deve anche tener conto dei vincoli posti dalle strategie di costruzione dei documenti identificati durante la Task Analysis.

Come rappresentato in Figura 1, ogni passo del progetto dell'IVL consiste di: i) progetto ed implementazione degli strumenti astratti o concreti; ii) verifica di completezza e coerenza; iii) validazione della loro adeguatezza rispetto alla comunicazione utente-calcolatore, tenendo conto delle semantiche degli utenti.

Durante questo processo, ogni passo o sequenza di passi può essere iterata diverse volte se la verifica o la validazione dei risultati di alcuni passi lo richiede. Ogni passo nel progetto di IVL richiede la scelta preliminare degli strumenti necessari per eseguire quel passo e di una tecnica di valutazione dei risultati ottenuti, che sono nuovi strumenti oppure nuovi insiemi di dati. Nella scelta ed uso dello strumento, sorgono problemi per l'esistenza delle due semantiche, quella dell'utente e quella del programma. La valutazione dei risultati deve essere eseguita rispetto ad entrambe le semantiche e richiede verifica e validazione sperimentale.

6. SUE per la Valutazione di Usabilità

Nel nostro lavoro, noi affrontiamo gli aspetti di usabilità con l'adozione della metodologia SUE (*Systematic Usability Evaluation*). La principale assunzione di SUE è che nella valutazione di usabilità di un sistema si dovrebbe tener conto degli aspetti strettamente

dipendenti dalla natura specifica del sistema. Quindi, contrariamente ai principi generali di usabilità, SUE richiede di considerare criteri di usabilità più specifici, cioè capaci di considerare meglio le caratteristiche di quella categoria di sistemi. Un processo di valutazione dell'usabilità basato sulla metodologia SUE combina ispezione e test con gli utenti. Durante l'ispezione, condotta per prima, valutatori esperti analizzano la versione corrente del sistema, cercando di identificare possibili violazioni dei criteri di usabilità [10]. Nel caso in cui si renda necessaria la verifica dell'impatto sugli utenti delle violazioni trovate dagli ispettori e/o l'efficacia dei cambiamenti che devono essere adottati in successive versioni del sistema, possono essere condotte delle sessioni di valutazione che coinvolgono gli utenti.

In particolare, esperimenti con gli utenti sono raccomandati nella fase di specifica del TVL (fase 2 in Figura 1), perché la notazione formale ottenuta dall'analisi dei task deve essere valutata rispetto alle semantiche dell'utente, in modo da essere certi che gli utenti capiscano correttamente la definizione del TVL ed il suo uso.

L'ATVL che è generato nella fase 3 di Figura 1 può, nella gran parte dei casi, essere valutato con il solo uso di metodi di ispezione, in quanto valutatori esperti dovrebbero essere sufficienti per determinare l'appropriatezza dei vari meccanismi usati per sfruttare le capacità interattive del sistema, quali uso di feedback, effetti di animazione, ecc. Per valutare l'IVL finale è invece opportuno affiancare l'ispezione con tecniche di valutazione che coinvolgono gli utenti.

Nel primo anno di attività si è lavorato per rifinire la tecnica di ispezione proposta in SUE e per validarla. I risultati sono riportati nei lavori [10, 11, 12, 13].

7. Conclusioni

Abbiamo presentato una metodologia per progettare sistemi interattivi visuali, che enfatizza il bisogno di sistemi accessibili dagli utenti finali. Tale metodologia mira a soddisfare tre principi: a) l'utente deve sempre capire gli effetti dell'attività del sistema rispetto all'esecuzione del suo compito; b) l'utente deve sempre avere il controllo della computazione interattiva, evitando di perdersi nello spazio virtuale; c) il sistema deve evitare e/o bloccare gli errori degli utenti mantenendosi in un insieme prevedibile di stati. Per soddisfare questi principi, il progetto del sistema è diviso in tre passi principali:

1. Dall'analisi dei task è derivato il linguaggio TVL che consiste di tutte le sentenze visuali richieste per compiti specifici.
2. Il linguaggio ATVL è derivato per sfruttare le possibilità di interazione e di animazione in un sistema informatico.

3. Il linguaggio visuale di interazione (IVL) è ottenuto specificando i pattern caratteristici che comunicano all'utente lo stato del processo d'interazione, fornendo anche l'accesso alle funzionalità del sistema.

In ognuno di questi passi si effettuano valutazioni di usabilità secondo quanto descritto dalla metodologia SUE.

8. Ringraziamenti

La metodologia presentata è stata parzialmente illustrata nel lavoro [3], scritto in collaborazione con membri delle unità di Brescia e Roma, che si ringraziano per la collaborazione.

Riferimenti

[1]	A. Bianchi, M. D'Enza, M. Matera, A. Betta, "Designing Usable Visual Languages: the Case of Immune System Studies". Proc. IEEE Symposium Visual Languages '99, pp.254-261, 1999.
[2]	P. Bottoni, U. Cugini, P. Mussio, C. Papetti, M. Protti, "A system for form-feature-based interpretation of technical drawings", Machine Vision and Application, vol. 8, n.5, pp. 326-335, 1995.
[3]	P. Bottoni, M.F. Costabile, S. Levialdi e P. Mussio, "From User Notations to Accessible Interfaces through Visual Languages", Proc. UA-HCI 2001, New-Orleans, USA, pp.252-256.
[4]	S.K. Chang, P. Mussio, "Customized Visual Language Design". Proc. of Eighth Int'lConference on Software Engineering and Knowledge Engineering, SEKE'96, pp. 553-562, 1996.
[5]	R. Prates, C. De Souza, and S. Barboza, "A Method for Evaluating the Communicability of User Interfaces". Interactions, Vol.7, No.1, pp.31-38, 2000.
[6]	B. Latour, "Visualization and Cognition: Thinking with Eyes and Hands". In: Knowledge and Society: Studies in the Sociology of Culture Past and Present, Vol. 6, pp.1-40, 1986.
[7]	A. Dix, J. Finlay, G. Abowd, R. Beale, Human Computer Interaction, Prentice Hall, London, 1998.

[8]	P. Bottoni, M.F. Costabile, P. Mussio, "Specification and Dialogue Control of Visual Interaction through Visual Rewriting Systems". <i>ACM TOPLAS</i> , Vol. 21, N. 6, pp. 1077-1136, 1999..
[9]	P. Bottoni, S.K. Chang, M.F. Costabile, S. Levialdi, P. Mussio, "On the Specification of Dynamic Visual Languages". <i>Proc. IEEE Symposium Visual Languages'98</i> , pp.14-21, 1998.
[10]	M. Matera, M.F. Costabile, F. Garzotto e P. Paolini, SUE Inspection: an Effective Method for Systematic Usability Evaluation of Hypermedia, <i>IEEE Transactions on Systems, Man and Cybernetics- Part A</i> , in print.
[11]	M.F. Costabile, M. Matera, Guidelines for Hypermedia Usability Inspection", <i>IEEE Multimedia</i> , Vol 8, N. 1, 2001, 66-69.
[12]	M.F. Costabile, A. De Angeli e M. Matera, Guiding Usability Evaluators During Hypermedia Inspection, <i>Proc. of Human-Centric Computing Symposia HCC 2001</i> , Stresa, Italy, September 5-7, 2001, IEEE CS Press, pp. 332-333.
[13]	A. De Angeli, M. Matera, M.F. Costabile, F. Garzotto e P. Paolini, On the Advantages of Systematic Inspection for Evaluating Hypermedia Usability, <i>Int. Journal on HCI</i> , in print.