

VALUTAZIONE DELL'USABILITA' DI UN PROGRAMMA E PROPOSTA DI UN SUO MIGLIORAMENTO

Veronica Gabusi
gabusi@ing.unibs.it
030-3715456

Elisabetta Piantoni
epiantoni@hotmail.com
0335-5424173
030-711081

Sergio Sigala 026790
ssigala@globalnet.it
0347-4443326
0364-534107

16 luglio 2001

UNIVERSITA' DEGLI STUDI DI BRESCIA
FACOLTA' DI INGEGNERIA
A.A. 2000/2001
Corso di Interazione Uomo-Macchina
Prof. Piero Mussio

Indice

1	Introduzione	7
2	Overview di Vis5D	9
2.1	Interazione col programma	10
2.2	Documentazione	10
2.3	Installazione	12
2.4	Altri programmi	12
2.5	Formato dei data set	12
2.6	Nomi delle variabili	13
2.7	Proiezioni e sistemi di coordinate verticali	13
2.8	Linguaggio Tcl	13
3	Intervista col cliente	14
3.1	Conoscenze base sul dominio	15
3.1.1	Modelli	16
3.2	Classificazione degli utenti	16
3.2.1	Professionisti	17
3.2.2	Ricercatori	18
3.2.3	Studenti	19
3.2.4	Tabella riassuntiva	19
3.3	Problemi del sistema	21
3.4	Finalità dell'intervento	22
4	Definizione dei compiti	23
5	Il concetto di usabilità	30
6	Metodi di valutazione dell'usabilità	31
6.1	I dieci principi di Nielsen	32
6.1.1	Principio 1: visibilità dello stato del sistema	33
6.1.2	Principio 2: adeguare il sistema al mondo reale	34
6.1.3	Principio 3: controllo e libertà all'utente	34
6.1.4	Principio 4: assicurare consistenza	35
6.1.5	Principio 5: prevenzione degli errori	35
6.1.6	Principio 6: riconoscimento piuttosto che richiamo	35
6.1.7	Principio 7: flessibilità ed efficienza d'uso	35
6.1.8	Principio 8: estetica e disegno minimalisti	36
6.1.9	Principio 9: messaggi d'errore	36
6.1.10	Principio 10: aiuti e documentazioni	37
6.2	Metodo Walkthrough	37
6.3	Interviste e questionari	38
7	Scelta dei metodi di valutazione per Vis5D	39

8	Valutazione walkthrough di Vis5D	40
8.1	Prototipo del sistema	40
8.2	Prototipo utente	40
8.3	Compiti	40
8.4	Liste di azioni	41
8.5	L'analisi in dettaglio	41
8.5.1	Azione 1: partenza del programma.	42
8.5.2	Azione 2: attivazione del pulsante Hwind1	42
8.5.3	Azione 3: selezione della quota.	43
8.5.4	Azione 4: cambio del punto di vista.	43
8.5.5	Azione 5: selezione dell'ora.	43
8.5.6	Azione 6: salvataggio di un'immagine.	44
8.5.7	Azione 7: chiusura del programma.	45
9	Valutazione euristica di Vis5D	46
9.1	La valutazione in dettaglio	46
9.2	Principio 1: visibilità dello stato del sistema	47
9.3	Principio 2: adeguare il sistema al mondo reale	47
9.4	Principio 3: controllo e libertà all'utente	47
9.5	Principio 4: assicurare consistenza	48
9.6	Principio 5: prevenzione degli errori	49
9.7	Principio 6: riconoscimento piuttosto che richiamo	49
9.8	Principio 7: flessibilità ed efficienza d'uso	50
9.9	Principio 8: estetica e disegno minimalisti	50
9.10	Principio 9: messaggi d'errore	51
9.11	Principio 10: aiuti e documentazioni	51
10	Risultati delle valutazioni in economia	52
10.1	Cosa il nostro Nielsen non trova	52
10.2	Cosa il nostro walkthrough non dice	52
11	Esperimento di valutazione di Vis5D	53
11.1	Struttura	53
11.1.1	Prima fase: training	53
11.1.2	Seconda fase: il test	54
11.1.3	Terza fase: il questionario	55
11.2	Risultati dell'esperimento	55
11.3	Osservazioni	55
12	Valutazione di Vis5D tramite questionario	59
12.1	Scelta del questionario	59
12.2	Tipi di domande usati	59
12.3	Struttura del questionario	60
12.4	Questionario di valutazione di Vis5D	61

12.5	Risultati del questionario	64
12.5.1	PARTE I: informazioni sull'utente	64
12.5.2	PARTE II: domande generali sul Vis5D	65
12.5.3	PARTE III: domande approfondite sul Vis5D	65
12.5.4	PARTE IV: osservazioni e suggerimenti	66
13	Conclusioni sulla valutazione	68
14	Introduzione al prototipo	69
14.1	Osservazione sulla progettazione	69
14.2	Generalità sui prototipi	70
14.2.1	Classificazioni	70
14.2.2	Costo	71
14.3	Il nostro prototipo	71
15	Overview di Qt	73
15.1	La situazione del mercato	73
15.2	Il ruolo di Qt	74
15.2.1	La storia	74
15.2.2	L'architettura	74
15.2.3	Sviluppo multipiattaforma	74
15.2.4	Look and feel	75
15.3	Il designer	75
16	Il prototipo	78
16.1	Installazione	78
16.2	Avvio del programma	79
16.2.1	Selezione categoria utente	81
16.2.2	Apertura di un data set	83
16.3	La barra dei menu in dettaglio	83
16.3.1	Menu F ile	84
16.3.2	Menu D isplay	85
16.3.3	Menu O perations	93
16.3.4	Menu V ariables	97
16.3.5	Menu F rame	98
16.3.6	Menu S cript	100
16.3.7	Menu P references	101
16.3.8	Menu H elp	102
17	Considerazioni sulla realizzabilità informatica	105
17.1	La nostra scelta	105
17.2	Verifica della realizzabilità	105
17.3	Fusione tra interfaccia e codice	106
17.4	Sulla manutenzione	107

Elenco delle figure

1	Fasi del ciclo di vita a stella	7
2	Vis5D avviato senza alcun data set	10
3	Vis5D mentre l'utente analizza il data set LAMPS.v5d	11
4	Il sistema modellistico integrato	17
5	Campo di concentrazione di un'inquinante a 8km	25
6	Campo di vento a 3km	26
7	Clipping del data set	27
8	Probing del data set	28
9	Sounding del data set	29
10	Finestrella che compare nella rappresentazione volumetrica	50
11	Una sessione d'uso del designer	76
12	Avvio di NewVis5D	80
13	Selezione della categoria d'utente	82
14	Aspetto di NewVis5D senza data set caricato	82
15	Apertura di un data set	83
16	NewVis5D dopo aver caricato un data set	84
17	Contenuto del menu F ile	85
18	La finestra 3D display	86
19	Contenuto del menu D isplay	86
20	Salvataggio di un'immagine	87
21	Selezione della manipolazione R otate	88
22	Termine della rotazione del data set	89
23	Selezione dei campi scalari da visualizzare	90
24	Selezione dei campi di vento da visualizzare	91
25	Dialogo di selezione del punto di vista	92
26	Contenuto del menu O perations	93
27	Dialogo di probing	94
28	Il probing in azione	95
29	Dialogo di sounding	96
30	Contenuto del menu V ariables	97
31	Contenuto del menu F rame	99
32	Il dialogo di selezione del fotogramma	100
33	Contenuto del menu S cript	100
34	Contenuto del menu P references	101
35	Contenuto del menu H elp	102
36	L'aiuto in linea offerto da NewVis5D	103
37	Il tooltip di aiuto per l'icona probing	104

Elenco delle tabelle

1	Caratteristiche degli utenti	19
---	----------------------------------------	----

2	Schema di allocazione dei compiti	24
3	Scelta dei metodi di valutazione	39
4	Risultati qualitativi, parte 1/2	56
5	Risultati qualitativi, parte 2/2	57
6	Risultati quantitativi	57

1 Introduzione

Questo elaborato è stato sviluppato nell'ambito del corso di Interazione Uomo-Macchina, tenuto dal Prof. Piero Mussio all'Università degli Studi di Brescia. La finalità è dimostrare di aver assimilato i concetti illustrati durante il corso, attraverso la loro applicazione ad un caso reale. Il problema è di migliorare l'usabilità di un programma di visualizzazione tridimensionale chiamato **Vis5D**.

Come metodologia di progetto è stato adottato il ciclo di vita a stella modificato (Hartson, Hix e Mussio, 1999), composto dalle sette fasi rappresentate schematicamente nella figura 1.

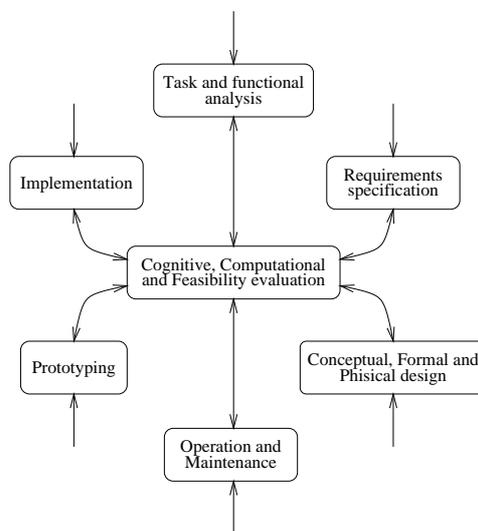


Figura 1: Fasi del ciclo di vita a stella

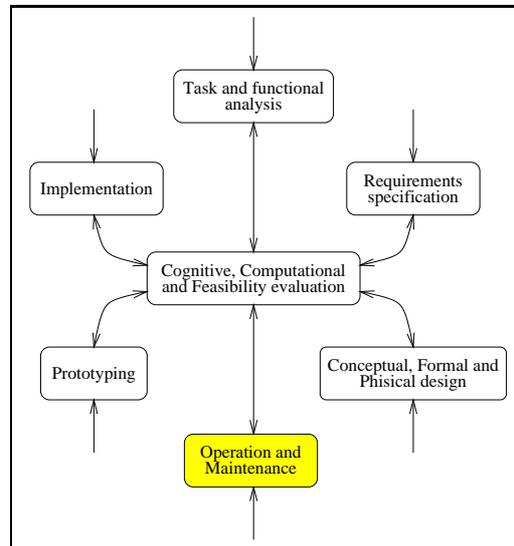
La caratteristica peculiare di questo ciclo di vita è che non esiste una direzione di percorrenza predefinita, come succede invece nei classici (e a volte poco realistici!) cicli di vita a cascata ed a spirale. Da ogni attività è possibile passare a qualsiasi altra, ma solo dopo aver valutato, cioè considerato in bontà, i risultati parziali dell'attività che si lascia, come indicato dal blocco centrale nella figura. La valutazione diviene l'attività centrale del modello: tutti gli aspetti dello sviluppo e dell'uso di un sistema informatico sono soggetti ad una costante valutazione da parte di utenti ed esperti.

Altra caratteristica importante è che lo sviluppo di un nuovo sistema o la manutenzione di uno già esistente possono iniziare da qualsiasi attività; nella figura questa possibilità è rappresentata dalle frecce entranti nelle sei fasi esterne del ciclo di vita. Nel nostro caso, dovendo migliorare un sistema già esistente, si partirà dalla fase di uso e manutenzione.

La scelta di questo metodo flessibile di descrizione dei processi di sviluppo di un prodotto software è dovuta alla complessità delle organizzazioni ed alla varietà di situazioni in cui e per le quali i sistemi informatici vengono sviluppati. Durante

l'elaborato si terrà traccia delle fasi del ciclo a stella che verranno attraversate, illustrandole mediante una serie di figure simili alla precedente, ma nelle quali la fase o le fasi interessate saranno evidenziate.

2 Overview di Vis5D



Si tratta di un software che permette di visualizzare sia dati localizzati su griglie a spaziatura fissa (**gridded data**) che dati con spaziatura irregolare (**irregularly located data**). I file da visualizzare sono chiamati **data set** e provengono da modelli numerici del tempo oppure da osservazioni superficiali ed atmosferiche o altre sorgenti simili.

Un data set di Vis5D è costituito da un insieme di punti di uno spazio a cinque dimensioni, delle quali le prime tre costituiscono le coordinate spaziali del dato all'interno di una griglia tridimensionale. La quarta dimensione è il tempo; l'ultima identifica la variabile fisica cui il dato si riferisce (temperatura, pressione, ...), in modo che in ciascun file si possano memorizzare gli andamenti di più parametri.

Vis5D è in grado di confrontare più file, tutti passati sulla linea di comando all'avvio dell'applicazione oppure importati individualmente mentre il programma è già in esecuzione. Questi dati possono essere visualizzati sovrapposti nel medesimo **3-D Display**, oppure possono venire listati *side-by-side* in una tabella tipo spread sheet.

Il sistema Vis5D completo comprende il visualizzatore vero e proprio e diversi altri programmi per gestire, analizzare e convertire i data set. E' da sottolineare che il visualizzatore possiede lo stesso nome del pacchetto, perciò nel seguito si useranno spesso i termini "Vis5D" e "visualizzatore" in modo intercambiabile, per riferirsi alla medesima applicazione visualizzatrice di data set; quando ci si riferirà all'intero sistema si utilizzerà frequentemente il termine "pacchetto". Vis5D è distribuito sotto licenza **GNU GPL**; i responsabili hanno allestito una mailing list libera a cui gli utenti possono inviare richieste di aiuto, soluzioni a problemi posti da altri utilizzatori e proposte di miglioramenti al sistema.

La figura 2 riporta le finestre mostrate dal visualizzatore al suo avvio, senza

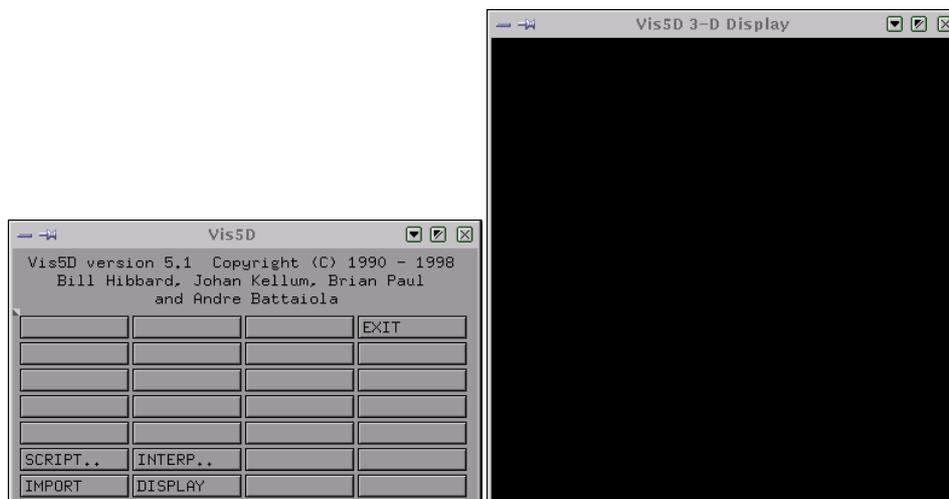


Figura 2: Vis5D avviato senza alcun data set

alcun data set caricato. Questa situazione è ottenibile solamente passando l'opzione **-nofile** sulla linea di comando; altrimenti il programma si aspetta come primo argomento il nome del file contenente il data set e termina brutalmente con un messaggio informativo se tale file non viene specificato.

La figura 3 mostra una situazione tipica in cui protrebbe trovarsi un'utente mentre analizza un data set; da notare il notevole spazio occupato dalle finestre e la grande quantità di comandi disponibili, disposti in modo non strutturato e poco ordinato. L'unico help in linea è il box bianco nella finestra principale. Tale help ricorda all'utente le azioni che può svolgere premendo i tasti del mouse, ma non è completamente sensibile al contesto e pertanto in alcune situazioni può essere fuorviante.

2.1 Interazione col programma

Il visualizzatore, grazie alla sua interfaccia grafica, viene quasi completamente controllato tramite il solo mouse; è necessario però un dispositivo a tre tasti, oppure a soli due se si accetta di emulare il tasto centrale mediante pressione contemporanea dei due laterali. La tastiera si utilizza soltanto per inserire nomi di file, di variabili o per attivare certe funzionalità poco frequenti. In particolare, l'uso dei tasti funzione è estremamente limitato: vengono sfruttati solo **F1** e **F2**, per funzionalità molto marginali. Non si usano combinazioni di tasti.

2.2 Documentazione

La documentazione disponibile è essenzialmente quella contenuta nel file in formato testo chiamato **README**, che accompagna il prodotto. Contiene informazioni sia riguardo all'installazione che all'uso del sistema e non è strutturato per

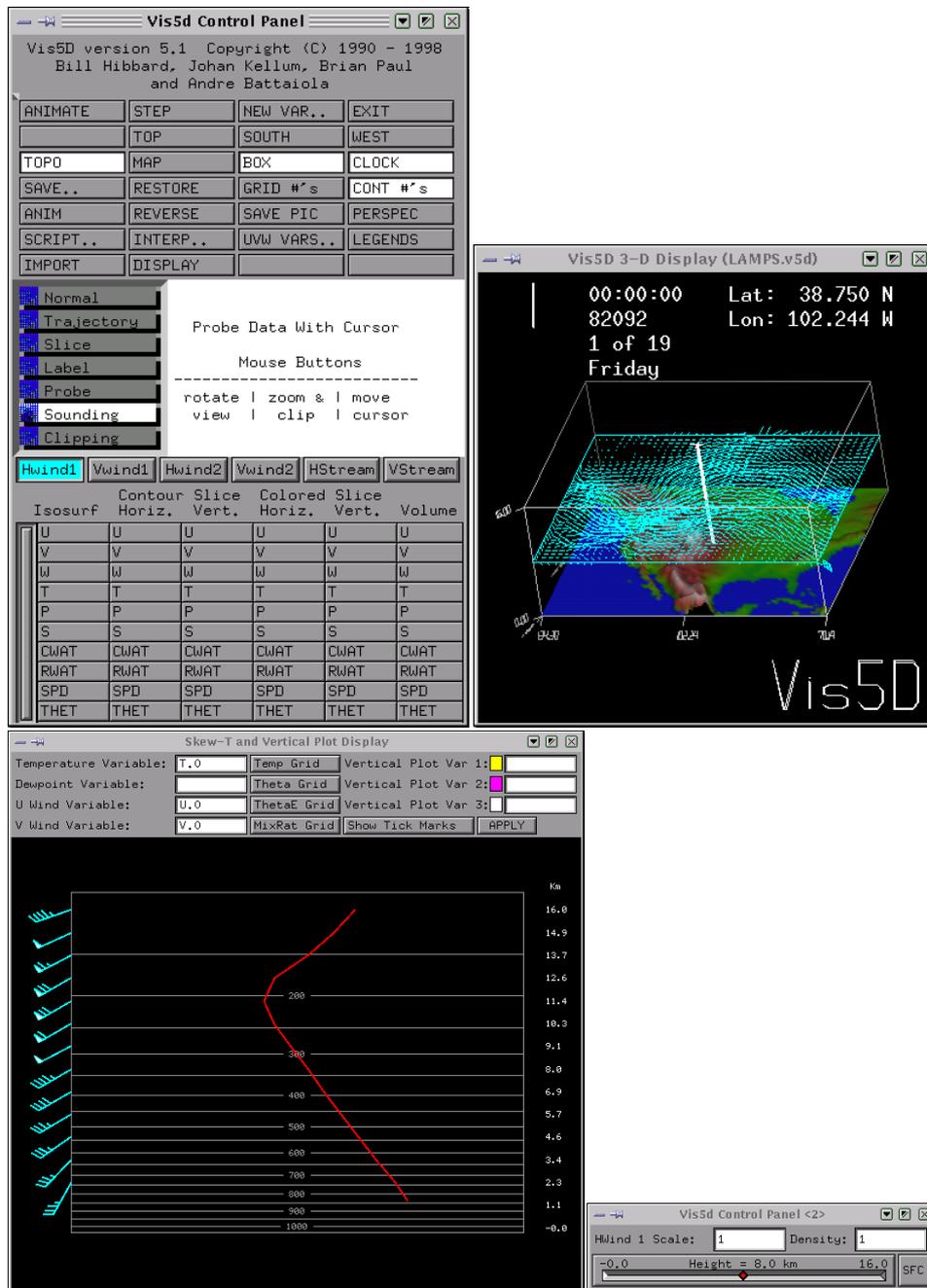


Figura 3: Vis5D mentre l'utente analizza il data set LAMPS.v5d

categorie di utenti (novizi, esperti del dominio, programmatori, manutentori, amministratori, ...). Inoltre non risulta nemmeno completo, in quanto molti comandi del Vis5D restano oscuri anche dopo un'attenta lettura di tutto il manuale. Secondo il manuale il modo migliore per imparare ad usarlo è sperimentando, anche perché non è possibile danneggiare i data set dall'interno del programma.

Leggendo questo file ci si rende conto che l'utilizzatore tipico, per cui il sistema e la documentazione sono stati pensati, è un esperto del dominio con discrete conoscenze di programmazione. Altre fonti di documentazione meno ricche sono alcune pagine html, dei data set d'esempio e delle slide, tutti reperibili sulla rete. L'ultima risorsa d'aiuto è la mailing list.

2.3 Installazione

Il software viene proposto in due forme: come sorgenti da compilare a cura dell'utente e come serie di pacchetti già compilati, per le piattaforme su cui viene prevalentemente utilizzato. In entrambi i casi deve essere prelevato da un sito **ftp** e decompresso; se si sceglie di scaricare i sorgenti il sistema deve anche essere configurato, compilato ed installato.

2.4 Altri programmi

Oltre al visualizzatore, il pacchetto Vis5D include diversi altri programmi per manipolare i data set: **v5dinfo**, **v5dstats**, **v5dedit**, **v5dappend**, **gr3d_to_v5d**, **igg3d**, **igu3d**, **gg3d**, **listfonts**, **comp5d**, **help**, **maketopo.c**, **makemap.c**, **newmap.c** e **v5dimport**.

2.5 Formato dei data set

In aggiunta ai dati puntuali già descritti precedentemente, in ogni file sono registrate anche informazioni riguardanti l'estensione di ciascuna dimensione della griglia, una coordinata geografica di riferimento per i dati, un'orientamento per la griglia, i nomi delle variabili memorizzate e alcune informazioni riguardo al tempo.

Vis5D è in grado di elaborare solo i due formati proprietari compressi **v5d** e **comp5d**; il primo è il nuovo formato ufficiale mentre il secondo serve solo per garantire la compatibilità con le versioni precedenti del sistema. E' compito dell'utilizzatore scrivere dei convertitori da e verso i propri formati.

Vis5D permette di marcare un dato del data set come **mancante**, assegnandogli un valore assoluto maggiore di $1.0 \cdot 10^{30}$. I dati mancanti hanno solitamente due origini: possono essere valori non ammissibili provenienti da simulazioni o misure oppure possono essere dovuti a campagne di osservazione non complete. L'effetto dei dati mancanti è di visualizzare buchi nelle **iso-surfaces** e **contour slices** e regioni nere nelle **colored slices**.

Ci sono alcuni vincoli sulle dimensioni dei data set che possono essere visualizzati da Vis5D: un massimo di 30 variabili fisiche e di 400 istanti di tempo (detti **time steps**).

2.6 Nomi delle variabili

Per convenzione, ad alcuni nomi di variabile sono associati significati particolari:

- **U** è la variabile che individua la componente del campo di vento parallela alle righe;
- **V** individua la componente del campo di vento parallela alle colonne;
- **W** individua la componente del campo di vento lungo la direzione verticale (zenith);
- **T** è il campo della temperatura;
- **P** è il campo della pressione.

Il campo di vento è *vettoriale* e perciò viene scomposto nelle sue tre componenti ortogonali **U**, **V** e **W**. E' possibile scegliere altre variabili per identificarlo, agendo sulla linea di comando o nei menu di Vis5D. La sua unità di misura di default è il **metro al secondo**. I campi **U**, **V**, **W**, **T** e **P** sono tutti *scalari*. Di default temperatura e pressione si misurano in **Kelvin** e **milliBar**, rispettivamente.

2.7 Proiezioni e sistemi di coordinate verticali

Col termine **proiezione** si intende l'insieme di relazioni tra le righe e le colonne della griglia in cui sono memorizzati i dati e le coordinate geografiche (latitudine e longitudine) a cui tali dati corrispondono. Il termine **sistema di coordinate verticali** individua la relazione tra il livello verticale di ciascun dato e l'altitudine nell'atmosfera (o la profondità nell'oceano) che quel dato identifica.

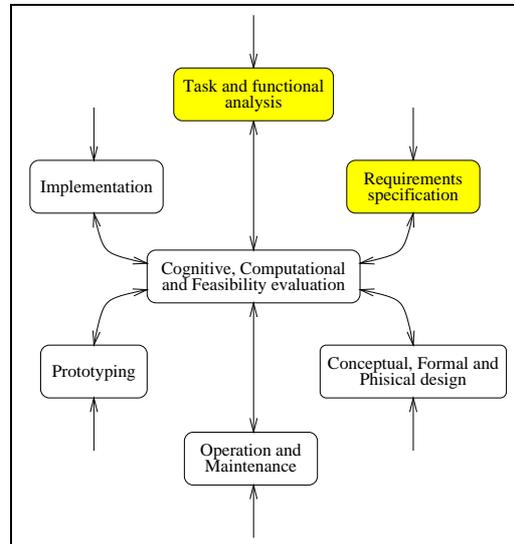
Vis5D supporta varie proiezioni e molteplici sistemi di coordinate verticali.

2.8 Linguaggio Tcl

Il visualizzatore ha integrato un interprete per il linguaggio **Tcl**. Ciò rende possibile controllare Vis5D anche tramite un file di testo, contenente sequenze di comandi in Tcl. Questa attività di scripting viene considerata argomento avanzato dalla documentazione, pertanto la sua descrizione è contenuta in un manuale dedicato, scaricabile anch'esso dalla rete.

La possibilità di creare nuove funzionalità mediante script è ridotta nella versione precompilata di cui disponiamo, perché l'interprete integrato è limitato ai soli costrutti fondamentali del linguaggio Tcl.

3 Intervista col cliente



In questa fase Veronica, utilizzatrice da tempo di Vis5D, agisce sia nelle vesti del cliente che commissiona il lavoro, sia in quelle di esperta del dominio che introduce gli aspiranti esperti di interazione uomo-macchina alla scienza della modellistica e del controllo ambientale. A questo proposito, è essenziale ricordare che cliente e utente sono figure distinte, che solo ben di rado coincidono. In particolare vale questa distinzione:

- **Cliente**, o committente, è colui che commissiona e paga il lavoro di analisi, progettazione e implementazione di un nuovo sistema o di miglioramento di uno già esistente.
- **Utente** è ogni individuo la cui attività è influenzata dal sistema. Sono utenti diretti coloro che operano in prima persona sul sistema; tutti gli altri, che si limitano a subirne l'influenza indiretta, sono utenti indiretti. Nel progetto di un sistema è bene considerare entrambe le tipologie, evitando di fermarsi solo sugli utenti diretti.

L'intervista col cliente ha diversi obiettivi, ora solamente elencati e approfonditi meglio in seguito:

1. Acquisire conoscenze di base sul dominio. Il cliente deve fornire alcune nozioni sul dominio nel quale opera e spiegare come viene utilizzato Vis5D nel suo particolare ambiente lavorativo.
2. Capire chi sono gli utenti del sistema. I clienti andranno partizionati in diverse categorie, caratterizzando bene ciascuna categoria in base a questi punti:

- frequenza di utilizzo del sistema;
 - livello di istruzione minimo necessario;
 - eventuale necessità di corsi di training;
 - possibilità di utilizzo di manuali;
 - profilo psicologico.
3. Raccogliere conoscenze circa il numero e la tipologia degli errori commessi più frequentemente, circa le mancanze del sistema esistente e i requisiti che il nuovo sistema dovrà soddisfare.
- Questo passo aiuterà i progettisti ad individuare le parti mancanti da introdurre nel nuovo sistema, le parti da rimuovere perché superflue e le parti da degradare o ottimizzare perché inefficienti.
4. Definire le finalità dell'intervento. Sono alcune direttive che guidano il lavoro, risultanti da varie considerazioni che tengono conto delle informazioni ricavate nei passi precedenti dell'intervista, delle volontà del cliente, delle capacità dei progettisti di intervenire sul sistema e delle risorse disponibili per compiere l'intervento.

Seguono in dettaglio i risultati dell'intervista.

3.1 Conoscenze base sul dominio

La visualizzazione scientifica nel settore della meteo-climatologia e dell'oceanografia è di fondamentale importanza, in quanto da essa deriva una chiara e completa comprensione dei fenomeni studiati. Ciò che l'utente richiede ad una piattaforma di visualizzazione è, generalmente, la possibilità di gestire interattivamente una grande quantità di dati e di eseguirne un'analisi dettagliata attraverso opportune tecniche di visualizzazione.

Tali tecniche sono ad esempio l'estrazione di sezioni dal volume di dati, la rappresentazione di isolinee e di isosuperfici, il volume rendering, l'uso efficiente dei colori per discriminare variabili diverse o per rappresentare l'insieme dei valori assunti da una variabile. Deve inoltre esser possibile visualizzare molte variabili contemporaneamente, senza deteriorare la qualità delle immagini prodotte e senza aumentare insostenibilmente la complessità.

Infine, la possibilità di animare rispetto al tempo le sequenze prodotte fornisce un valore aggiunto di grande importanza e, allo stesso tempo, facilita la comprensione di molti fenomeni. Poiché il lavoro di visualizzazione non costituisce solo la fase finale di una ricerca, ma è in realtà elemento di grande importanza di ogni tappa della sperimentazione, è particolarmente importante che anche le più complesse visualizzazioni possano esser gestite in maniera semplice ed in tempo reale.

Il software di visualizzazione Vis5D è appunto una valida risposta a tutte queste esigenze; offre, infatti, la possibilità di animare in tempo reale ed in 3D grandi

insiemi di dati e permette all'utente di utilizzare abbastanza facilmente strumenti quali la costruzione di isolinee e isosuperfici, visualizzazioni di campi vettoriali e la costruzione all'interno di quest'ultimi di traiettorie (funzionalità dette **particle tracing** e **ribbon tracing**).

Il cliente usa Vis5D per visualizzare ed analizzare il risultato di complesse simulazioni, applicate a modelli di qualità dell'aria ed osservazioni reali.

3.1.1 Modelli

I modelli di qualità dell'aria forniscono uno strumento per la simulazione dell'evoluzione dinamica degli inquinanti primari coinvolti nei processi di diffusione e nelle interazioni chimiche al fine di analizzare e ipotizzare l'impatto spazio-temporale di diversi scenari emissivi su scala regionale. Occorre dunque disporre di un sistema integrato di modelli, composto da un pre-processore delle emissioni, un modello meteorologico che integri le misure locali e le informazioni su grande scala e un modello di trasporto fotochimico tridimensionale a scala metropolitana.

Per un'analisi corretta del complesso fenomeno della diffusione e delle trasformazioni chimiche che interessano gli inquinanti primari emessi e i loro secondari è infatti necessario integrare le informazioni fornite dalle reti di monitoraggio dell'intensità del traffico e della qualità dell'aria con i dati meteorologici e chimici di interesse (ad esempio, misure al suolo e profili verticali di ozono nelle aree urbane e nell'hinterland possono essere utilizzate per assegnare le condizioni iniziali e al contorno delle simulazioni).

Grazie ad un sistema di modelli di questo tipo e a data set messi a punto in modo opportuno, è possibile simulare e confrontare gli effetti e le diverse modalità di impatto sull'atmosfera urbana di strategie alternative di controllo delle emissioni.

Il sistema modellistico

Il sistema di modelli selezionato per supportare l'analisi dell'impatto sulla qualità dell'aria è solitamente composto da un pre-processore meteorologico, un pre-processore chimico, un modello chimico di trasporto e diffusione e un modello per la stima delle emissioni, che interagiscono come illustrato nella figura 4.

3.2 Classificazione degli utenti

Tra tutto il vasto pubblico d'utenza di Vis5D sono state individuate tre possibili categorie, qui nel seguito sinteticamente presentate:

1. **Professionisti** del settore ambientale, ma non del sistema specifico Vis5D;
2. **Ricercatori** universitari, che conoscono sia il dominio che il sistema Vis5D;
3. **Studenti**, che conoscono parzialmente entrambi gli ambiti.

Per ognuna di queste classi d'utenza verrà ora fornita una breve descrizione, atta a caratterizzare in maniera più puntuale ogni categoria. Come obiettivo dell'elaborato ci proponremo il miglioramento dell'interfaccia di Vis5D nel rispetto delle esigenze della categoria degli studenti.

3.2.1 Professionisti

Visto l'accresciuto interesse del mondo politico verso le tematiche dell'inquinamento atmosferico e di conseguenza all'emanazioni di leggi per la tutela della salute dei cittadini e dello stato dell'ambiente, sono state istituzionalizzate alcune strutture tecnico-scientifiche aventi i compiti connessi appunto alla protezione dell'ambiente.

Tra queste strutture menzioniamo l'ANPA (Agenzia Nazionale per la Protezione dell'Ambiente) che, come l'acronimo stesso suggerisce, opera a livello nazionale, e l'ARPA (Agenzia Regionale per la Protezione dell'Ambiente) che svolge invece attività a favore di Regioni, Provincie, Comuni, Comunità montane ed altri enti pubblici ai fini dell'espletamento di funzioni nel campo della prevenzione e tutela dell'ambiente. Le agenzie nazionali, in particolare, sono chiamate a svolgere queste attività tecnico-scientifiche:

1. **Supporto tecnico** ai livelli istituzionali competenti nelle materie identificate dal testo legge: limiti di accettabilità di sostanze e agenti inquinanti, uso razionale di risorse, metodologie per il rilevamento dello stato dell'ambiente e per il controllo dei fattori di rischio.
2. **Controllo ambientale e segnalazione** alle autorità competenti delle violazioni in materia ambientale: controlli di fattori fisici, chimici, biologici di inquinamento acustico, dell'aria, dell'acqua, del suolo; analisi e controllo di eventi di rischio geologico, idrogeologico e sismico; controlli fitosanitari

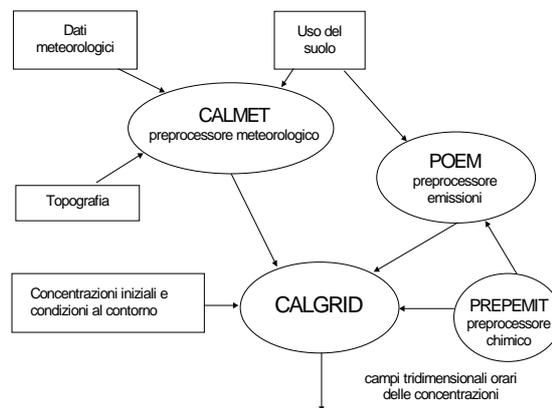


Figura 4: Il sistema modellistico integrato

per quanto concerne l'effetto di inquinamento diffuso derivante dall'uso di pesticidi.

3. **Informazione ambientale:** raccolta sistematica, anche informatizzata, di tutti i dati riguardanti la situazione ambientale regionale o di specifici territori; realizzazione di sistemi di monitoraggio ambientale in accordo con i servizi tecnici nazionali e il sistema informativo regionale; acquisizione, elaborazione e diffusione di informazioni e previsioni sullo stato delle variabili meteorologiche e sul loro impatto sull'ambiente e sulle attività agricole, industriali e civili.
4. **Promozione della ricerca** e diffusione delle informazioni di tecnologie ecologicamente compatibili, di prodotti e sistemi di produzione a ridotto impatto ambientale.
5. **Attività educative** e di formazione ambientale.

Alla luce di quanto sopra accennato, soprattutto per quanto riguarda la terza attività di informazione ambientale, i tecnici appartenenti a tali organismi possono, in alcune situazioni, avere necessità di uno strumento quale Vis5D che, come è stato illustrato nella sua presentazione, risulta essere un software particolarmente adatto per visualizzare insiemi di dati complessi e permetterne eventuali confronti.

L'utilizzo da parte di questi professionisti di Vis5D non è sistematico e richiede uno sforzo di apprendimento del sistema dovuto alle scarse competenze informatiche di queste persone, esperte invece del dominio di applicazione trattato. Infatti, generalmente non vengono organizzati corsi di formazione finalizzati all'apprendimento progressivo del software e non esistono manuali veri e propri che possano aiutare l'utente (tra i file di Vis5D è incluso un solo manuale, che descrive brevemente e sommariamente le varie potenzialità di questo strumento).

3.2.2 Ricercatori

La sensibilità verso problematiche ambientali ha sicuramente incentivato negli ultimi anni la crescita di ricerche universitarie finalizzate ad affiancare alla strumentazione per il monitoraggio della situazione ambientale anche sistemi informativi avanzati, che consentano in particolare sia di effettuare previsioni di episodi critici in tempo reale a scopi preventivi sia di valutare a priori l'impatto ambientale di nuove attività industriali o insediamenti urbane.

La categoria d'utenza dei ricercatori universitari è simile a quella dei professionisti sopra descritti, sia dal punto di vista sia della frequenza d'uso, non costante, sia della preparazione ambientale posseduta. Le conoscenze di base sono forse anche più approfondite dal punto di vista teorico, però vi è una certa deficienza di quel senso pratico peculiare solo a chi concretamente lavora per progetti ambientali; di contro i ricercatori possiedono sicuramente migliori competenze informatiche e quindi si trovano facilitati nell'approccio con un sistema abbastanza complesso come quello di Vis5D.

Categoria d'utenza	Frequenza d'utilizzo	Livello d'istruzione necessario	Necessità d'addestramento	Documentazione di supporto
Professionisti	saltuaria	medio	auspicabile	scarsa
Ricercatori	periodica	elevato	autonoma	scarsa
Studenti	occasionale	base	indispensabile	scarsa

Tabella 1: Caratteristiche degli utenti

Anche all'interno dell'università non è promossa nessun tipo di formazione, né esistono manuali specifici a disposizione dell'utente: l'unico aiuto offerto è la possibilità di confrontarsi con altri utilizzatori più o meno esperti, presenti nell'ambiente universitario stesso o consultati attraverso una mailing-list molto specifica.

3.2.3 Studenti

L'ultima categoria individuata è quella degli studenti, ai quali questo software viene presentato all'interno di corsi specifici per il territorio e l'ambiente, quali ad esempio sono, nell'ambito del programma di Ingegneria della Facoltà degli Studi di Brescia, *Modellistica* e *Controllo ambientale*. Si tratta di utilizzatori che possiedono conoscenze di base sufficienti per comprenderne l'utilizzo, ma che effettivamente sfrutteranno Vis5D solo a scopo propedeutico senza approfondirne mai, se non per interesse puramente personale, tutte le potenzialità.

Dal punto di vista della formazione offerta, la situazione è naturalmente leggermente differente rispetto ai casi precedenti: i docenti stessi forniranno infatti le informazioni necessarie per garantire una comprensione sufficiente anche se parziale del Vis5D.

3.2.4 Tabella riassuntiva

Per ogni categoria d'utenza sono state individuate alcune caratteristiche, raccolte nella tabella 1, che mettono in evidenza le differenti esigenze. Seguono alcuni commenti sulle varie colonne della tabella.

- **Frequenza d'utilizzo:** i professionisti utilizzano Vis5D raramente ma in questi brevi periodi vi lavorano in maniera molto intensa. I ricercatori ne abbisognano in maniera periodica, soprattutto verso il termine di progetti, nella fase in cui serve cioè uno strumento in grado di visualizzare i risultati della ricerca effettuata. Per quanto riguarda gli studenti, i corsi da loro seguiti richiedono un apprendimento abbastanza superficiale del Vis5D.
- **Livello d'istruzione necessario:** ai fini di un utilizzo adeguato ai propri scopi, i ricercatori devono possedere conoscenze modellistiche molto approfondite, mentre ai professionisti sono sufficienti conoscenze ambientali discrete. Per gli studenti, invece, è indispensabile soltanto aver appreso i concetti base illustrati durante i corsi da loro seguiti.

- **Necessità d'addestramento:** sarebbe auspicabile che per i professionisti venissero organizzati corsi di istruzione per l'uso di Vis5D, in modo da rendere omogenea la preparazione dello staff addetto al settore ambientale. I ricercatori, già più inclini ad approfondire autonomamente ricerche e studi, operano da autodidatti e, vivendo nell'ambiente universitario, sono aiutati dalla possibilità di confronto con altre persone interessate allo stesso dominio. Per quanto riguarda gli studenti: è indispensabile applicarsi nello studio degli argomenti affrontati a lezione, sufficienti comunque per conseguire una preparazione adeguata.
- **Documentazione di supporto:** la documentazione disponibile per l'utilizzo del Vis5D è, come si vedrà, scarsa e mal organizzata. E' concentrata tutta in un unico manuale in formato testo.

Più avanti si deciderà di concentrare le forze nel lavoro di miglioramento dell'interfaccia del Vis5D per i soli studenti. In previsione di ciò si è ritenuto molto utile definire ancora più in dettaglio questa categoria di utenti, considerando anche i loro aspetti psicologici, le caratteristiche fisiche e l'esperienza. Queste informazioni sono raccolte schematicamente nel profilo che segue.

Caratteristiche psicologiche

- **Stile cognitivo:** prevalentemente visivo. Il campione d'utenza è abbastanza eterogeneo, ma dovrebbe essere perlopiù abituato a ragionare rappresentando i concetti per immagini.
- **Attitudine:** elevata, si tratta di studenti frequentanti corsi di laurea ad indirizzo tecnico-scientifico.
- **Motivazione:** media; il raggiungimento di un buon livello di utilizzo del Vis5D non è fondamentale ai fini del superamento dell'esame.

Conoscenza ed esperienza

- **Livello di lettura:** elevato.
- **Abilità di battitura:** elevata, il campione di studenti è abituato ad utilizzare strumenti software di office automation.
- **Educazione:** tecnico-scientifica.
- **Esperienza informatica:** mediamente bassa, non tutti gli studenti appartengono al ramo di Ingegneria Elettronica.
- **Esperienza nel compito:** bassa, probabilmente è da poco che lo svolgono.
- **Esperienza dell'applicazione:** nessuna.

- **Esperienza nel dominio:** media. Dal punto di vista teorico le conoscenze acquisite durante le lezioni del corso dovrebbero esser abbastanza complete da permetterne un'applicazione pratica.
- **Linguaggio nativo:** italiano.
- **Uso di altri sistemi:** parziale. Gli studenti avranno utilizzato strumenti simili, come **winast** o **pcl2**, ma con potenzialità sicuramente minori rispetto a quelle offerte da Vis5D.
- **Conoscenza del calcolatore:** media;

Caratteristiche del compito

- **Frequenza d'uso:** occasionale.
- **Preparazione:** buona, formata attraverso una serie di esercitazioni in laboratorio assistite.
- **Uso del sistema:** a discrezione dello studente.
- **Importanza del compito:** poca, svolgere il compito è opzionale.

Caratteristiche fisiche

- **Capacità visive:** distribuite come nella popolazione.
- **Sesso:** maschile e femminile.
- **Gestione motoria:** sia destri che mancini.

3.3 Problemi del sistema

Il cliente considera il programma ostico e “difficile”, per vari motivi:

- il suo funzionamento non è intuitivo ma va imparato;
- l'assetto grafico è ostile e la grande quantità di comandi, tutti disponibili contemporaneamente, è disorientante;
- le abbreviazioni usate spesso non richiamano il significato e talvolta possono essere fuorvianti;
- no vi è aiuto in linea ed il manuale è disorganizzato.

Riassumendo in poche parole: **Vis5D ha notevoli problemi di usabilità.**

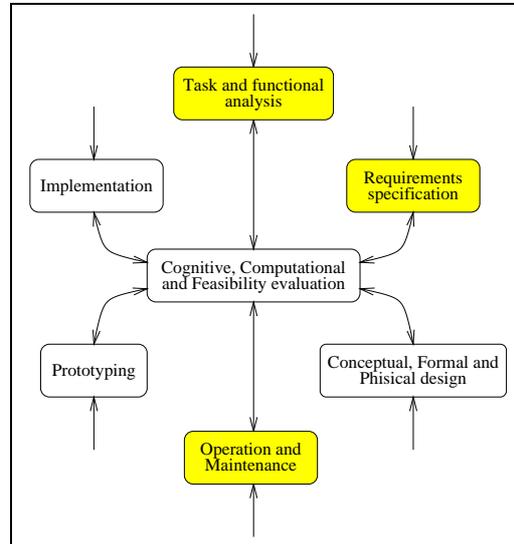
3.4 Finalità dell'intervento

L'intervento ha lo scopo ultimo di migliorare l'usabilità del sistema. Per raggiungere questo obiettivo il lavoro verrà suddiviso in molte tappe, attraverso le quali si cercherà di conseguire queste due finalità:

- **Capire il mondo reale** nel quale gli utenti operano, per essere in grado di produrre una lista approfondita di problemi del sistema.
- **Migliorare il sistema esistente**, usando come guida la lista di problemi individuata al passo precedente da esperti ed utenti, ulteriormente arricchita dalle nostre valutazioni.

Considerazioni quantitative, come percentuale di errori commessi, potranno aiutare a vedere questo secondo punto come un particolare *problema di ottimizzazione*, risolvibile con criteri scientifici.

4 Definizione dei compiti



L'intervista col cliente è stata fondamentale sia per classificare gli utenti del sistema che per identificarne i rispettivi compiti; la tabella 2 riporta schematicamente i risultati dell'indagine. La costruzione di una tabella del genere è essenziale per poter individuare con un certo dettaglio gli insiemi di funzionalità usati da ciascuna categoria di utenti e di fatto costituisce il nocciolo dell'analisi dei compiti.

Nelle righe sono riportate le tre categorie di utenti, mentre nelle colonne sono riassunti i compiti più frequenti che essi eseguono. Un segno in una cella indica che quell'utente svolge quel compito; per descrivere in modo più completo la frequenza o l'importanza di un certo compito in generale potrebbero essere usati anche segni multipli o perfino dei numeri. Due osservazioni:

- Le categorie dei professionisti e dei ricercatori sono state ulteriormente suddivise, dando luogo alle figure dei **professionisti chimici**, **professionisti meteorologi**, **ricercatori chimici** e **ricercatori meteorologi**, individuate dalle lettere **C** e **M**.

Ovviamente queste due suddivisioni non costituiscono delle partizioni: esisteranno infatti alcuni professionisti e diversi ricercatori interessati ad entrambi gli ambiti. Questa ulteriore specializzazione degli utenti è stata ritenuta utile perchè Vis5D dispone di alcune funzionalità rivolte chiaramente ai chimici e di altre che hanno senso solo se applicate a problemi meteorologici.

Un esempio di funzionalità del primo tipo è illustrato nella figura 5, dove è rappresentata la concentrazione di un'inquinante alla quota di 8km.

Per contro, la figura 6 illustra il campo di vento a 3km di altitudine, in una rappresentazione a vettori molto utile ai meteorologi e quindi da noi classificata come funzionalità del secondo tipo.

	Predisposizione data set	Visualizzazione dati		Clipping	Creazione nuova variabile	Analisi dati		Analisi comparata dei dati	Scripting	Salvataggio di una immagine	Produzione di una animazione
		C	M			P	S				
Professionisti	C	X		X	X	X				X	X
	M		X	X	X	X	X			X	X
Ricercatori	C	X		X	X	X		X	X	X	X
	M		X	X	X	X	X	X	X	X	X
Studenti		X	X	X	X	X				X	

Tabella 2: Schema di allocazione dei compiti

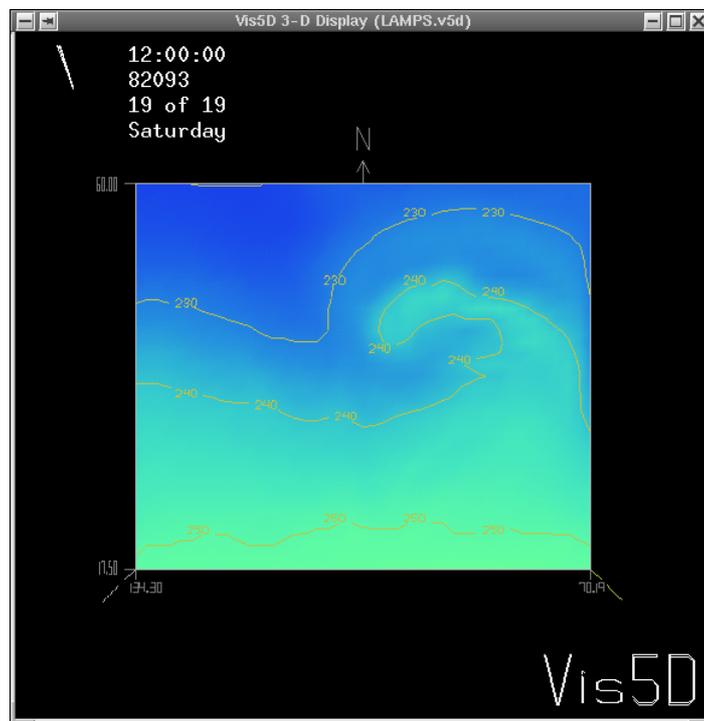


Figura 5: Campo di concentrazione di un'inquinante a 8km

- Gli studenti sono probabilmente interessati ad esplorare le funzionalità di entrambi gli ambiti.

Seguono alcune considerazioni sui compiti che sono stati individuati:

- La **predisposizione del data set** è l'insieme di attività che hanno come obiettivo la produzione di un data set. Sono attività che richiedono molte conoscenze di alto livello e una discreta esperienza, perciò si è pensato di attribuirle ai soli ricercatori. Il ricercatore decide la struttura del data set e successivamente lo riempie con dati opportuni.

La costruzione di un data set può essere effettuata sia con programmi esterni all'ambiente Vis5D, sia con alcuni degli strumenti compresi nel pacchetto.

- La **visualizzazione dati** comprende le attività svolte dall'utente al fine di visualizzare alcune variabili del data set, sotto certe rappresentazioni.

Comporta la selezione di una o più variabili di interesse fra quelle disponibili, il cambio del punto di vista e la scelta del tipo di grafico da visualizzare.

- Il **clipping** può essere sfruttato dall'utente per selezionare un sottovolume di dati dal dominio originario. Gli permette di concentrare l'attenzione solo

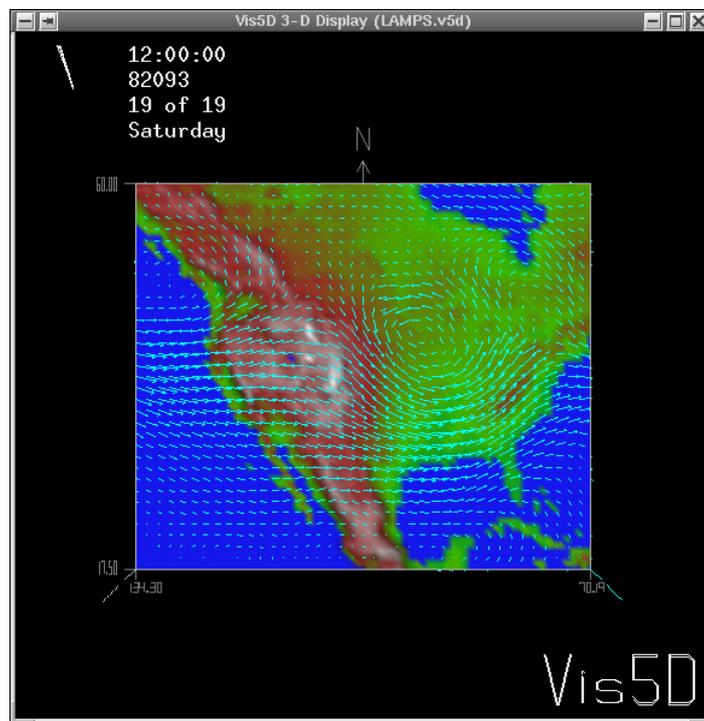


Figura 6: Campo di vento a 3km

sull'area di interesse, sopprimendo la visualizzazione dell'output relativo ai dati esterni a tale regione.

La figura 7 illustra il funzionamento del clipping su una rappresentazione volumetrica della velocità del vento. I rettangoli verdi delimitano la regione di interesse.

- La **creazione di nuova variabile** permette di definire nuove variabili sulla base di quelle già esistenti. A tal fine è necessario premere il pulsante **NEW VAR..**, presente nel pannello principale.

A questo punto si può clonare una variabile già esistente oppure costruirne una originale, attraverso l'introduzione di una funzione matematica arbitrariamente complessa che opera sulle variabili già esistenti.

- L'**analisi dei dati** si basa sulla visualizzazione quantitativa del contenuto del data set. Vis5D mette a disposizione due strumenti per questo compito: il **probing** e il **sounding**.

La figura 8 illustra il funzionamento del probing: vengono mostrati i valori che tutte le variabili assumono nel punto dello spazio tridimensionale che l'utente ha appena selezionato con l'aiuto del mouse. Il punto è indicato dalla crocetta, al centro della figura.

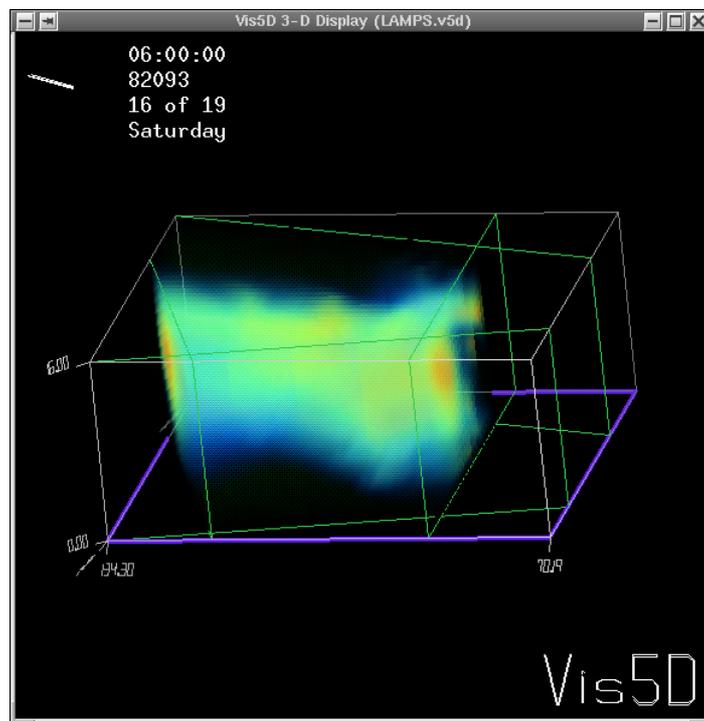


Figura 7: Clipping del data set

La figura 9 invece mostra il sounding in azione: nella finestra di sinistra l'utente effettua un "carotaggio" del dominio dall'alto e di conseguenza nella finestra a destra il sistema visualizza i rilievi pertinenti. Da notare l'output in stile prettamente meteorologico, contenente il punto di rugiada, diverse bandierine del vento e una curva di temperatura.

L'attività di probing (simbolo **P**) è stata attribuita ai chimici, mentre il sounding (simbolo **S**) è chiaramente una funzionalità da meteorologi.

Gli studenti sperimentano solo il probing.

- L'**analisi comparata** consiste nell'esaminare contemporaneamente due o più data set, cercando di individuare delle relazioni causa-effetto tra di essi. A questo scopo Vis5D permette la visualizzazione di più data set, sovrapposti o affiancati.

Secondo gli autori chi opera in questo modo e con queste finalità deve essere un esperto conoscitore delle dinamiche del dominio, dunque essenzialmente un ricercatore.

- Lo **scripting** è l'attività che coinvolge la scrittura da parte dell'utente di file di script in Tcl, che successivamente vengono fatti interpretare al Vis5D. I file di script possono automatizzare funzionalità già esistenti oppure crear-

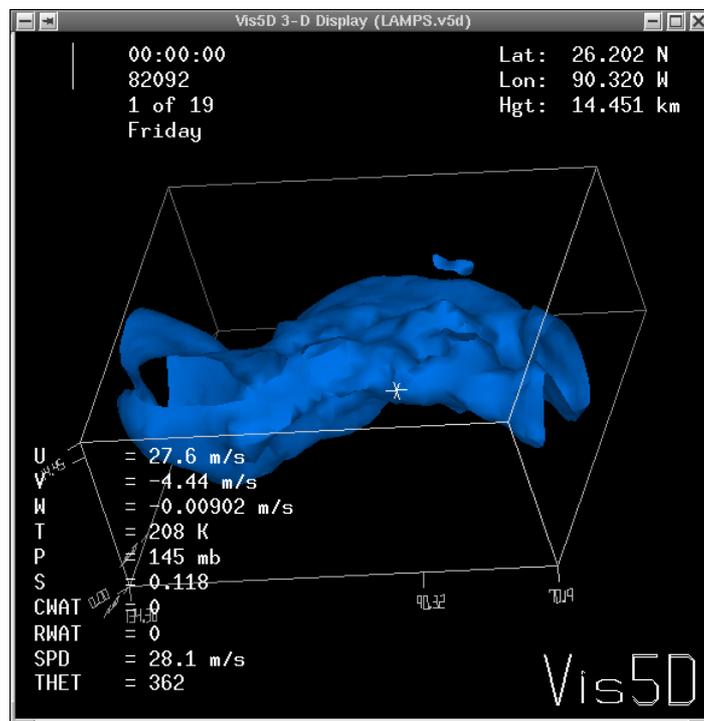


Figura 8: Probing del data set

ne di totalmente nuove. Dal visualizzatore l'interprete Tcl viene invocato premendo il pulsante **SCRIPT...**

- Attraverso il **salvataggio di un'immagine** l'utente può memorizzare il contenuto corrente del display 3D. Questo compito si realizza facilmente premendo il pulsante **SAVE PIC**. Il sistema si limita a mostrare un semplice dialogo, che chiede all'utente la selezione del tipo di formato e il nome del file per l'immagine che verrà creata.
- La **produzione di un'animazione** consiste generalmente nel creare un filmato salvando una sequenza di fotogrammi ad intervalli costanti di tempo, preservando la vista ed il tipo di grafico. Anche se quest'attività viene svolta sia dai professionisti che dai ricercatori, Vis5D non mette a disposizione alcuna funzionalità dedicata invocabile direttamente sotto forma di pulsante. E' possibile realizzarla in due modi; il primo è completamente manuale e prevede che l'utente salvi i fotogrammi necessari, uno alla volta tramite il pulsante **SAVE PIC**, e successivamente li ricomponga attraverso un programma adatto fino ad ottenere il filmato voluto. Il secondo modo è quello di utilizzare uno script che automatizzi il processo e invocarlo tramite il pulsante **SCRIPT...**

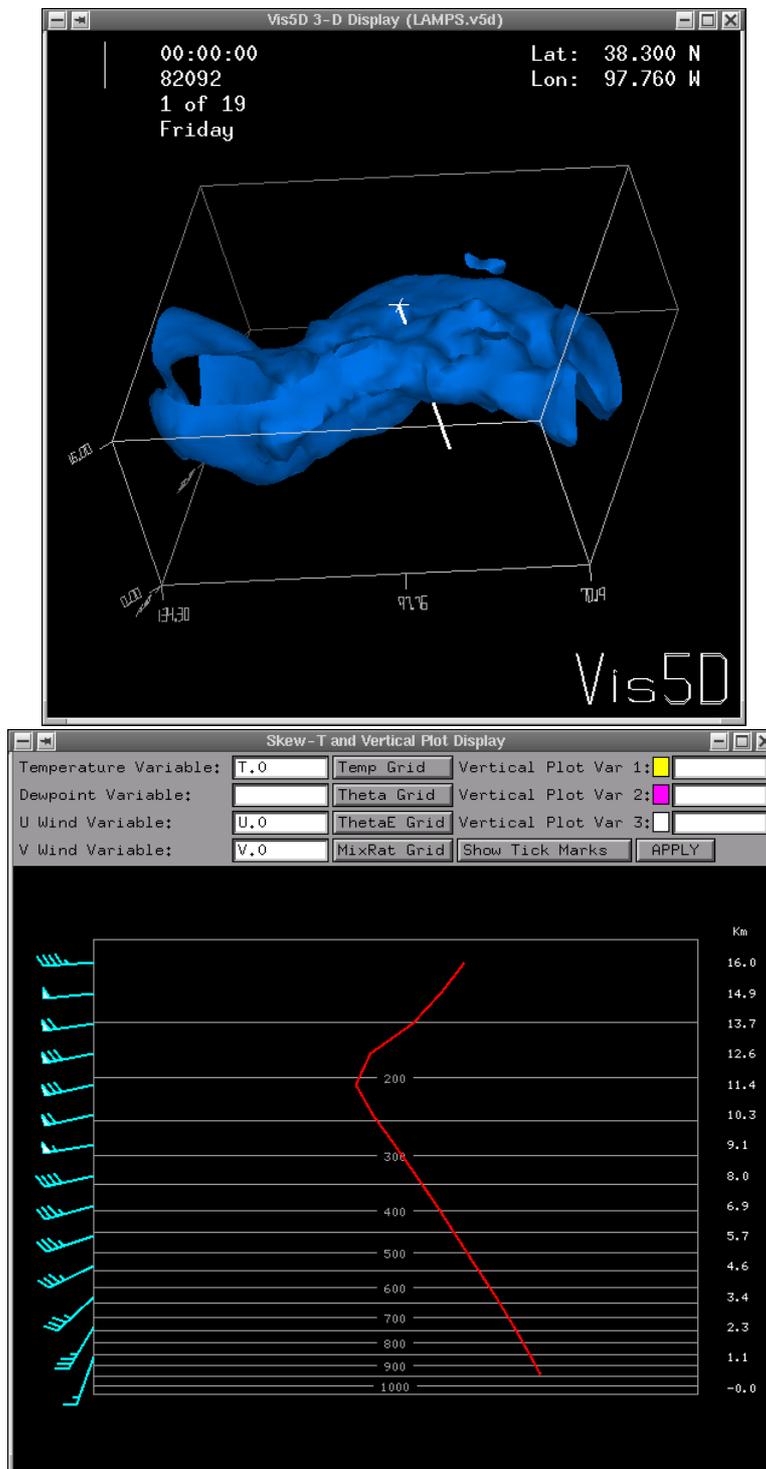


Figura 9: Sounding del data set

5 Il concetto di usabilità

Un sistema usabile soddisfa questi requisiti:

- è facile da capire;
- è facile da apprendere;
- è facile e piacevole da usare;
- fornisce le funzionalità di cui l'utente necessita;
- il suo utilizzo giova all'utente, perché lo agevola nel suo lavoro.

I fattori che determinano l'usabilità di un sistema comprendono:

- facilità di apprendimento, memorizzazione e uso;
- efficienza: la produttività deve aumentare, una volta a regime;
- basso numero di errori, che devono essere intercettati e gestiti adeguatamente e mai mandare in crash il sistema;
- soddisfazione: l'utente deve apprezzare il sistema.

Riassumendo, l'usabilità è una misura della facilità con la quale un sistema può essere imparato o usato, della sua sicurezza, efficacia ed efficienza e dell'attitudine che gli utenti hanno verso di esso. Perciò l'usabilità non è una grandezza fisica tradizionale e quindi non è quantificabile univocamente attraverso i metodi classici di misura diretta.

Ci si limiterà a verificare quanto un sistema rispetta i requisiti sopra esposti applicando delle procedure e dei principi dettati dal buon senso. In particolare, verranno realizzati questi passi:

- definizione delle caratteristiche dell'utente, attraverso l'**analisi dell'utente**;
- descrizione del contesto in cui l'utente opera, con l'**analisi dei compiti**;
- applicazione di uno o più **metodi di valutazione** dell'usabilità.

L'analisi dell'utente e l'analisi dei compiti sono necessarie perché l'usabilità è spesso un concetto relativo, dipendente dall'ambiente specifico nel quale il sistema viene adoperato. Entrambe sono già state svolte nei capitoli precedenti.

6 Metodi di valutazione dell'usabilità

Si tratta di alcune procedure di valutazione proposte da esperti di HCI, *human-computer interaction*, che aiutano ad identificare e discutere molti dei difetti e delle lacune che affliggono un certo sistema e che verranno genericamente denominati **problemi**.

La valutazione ricava un *feedback dagli utenti*, che permette di determinare *cosa vogliono e quali problemi sperimentano*. In questo modo gli esperti e l'intero gruppo di progettazione saranno in grado di giudicare se il sistema sotto esame si comporta come atteso e se soddisfa le necessità degli utenti; si potrà anche stimare la mole di informazioni che gli utilizzatori devono acquisire prima di poterlo adoperare con profitto. Idealmente si dovrebbe eseguire una fase di valutazione ogni volta che si completa un passo nel ciclo di vita a stella.

Vi sono molti metodi di valutazione, ma ciascuno di essi è riferibile ad una di queste tre categorie:

- Metodi che non richiedono la presenza dell'utente.

Sono *economici*, perché sono necessari solo alcuni esperti di HCI. Di questa categoria si vedranno un **metodo euristico**, basato sulla verifica del rispetto di certi principi da parte del sistema in esame (o del suo prototipo), ed un **metodo predittivo**, basato sulla simulazione da parte di esperti di HCI del comportamento previsto dell'utente.

Entrambi questi metodi consentono di costruire una lista particolareggiata e specifica di problemi.

- Metodi che richiedono la presenza dell'utente.

I metodi precedenti, non coinvolgendo l'utente, vanno bene sia nella fase di valutazione dell'implementazione, sia in quella di valutazione di un prototipo. Però non possono rimpiazzare totalmente gli esperimenti con gli utenti, per cui il sistema è inteso.

Esiste un nutrito gruppo di metodi che coinvolgono l'utente e tutti richiedono la disponibilità di un'implementazione del sistema, sotto una qualche forma.

Cadono in questa categoria i *metodi empirici (esperimenti)*, i *metodi osservativi (thinking aloud, cooperative evaluation, paper and pencil, audio and video recording, computer logging e user notebooks)* e infine i *metodi indiretti*, che si basano sulla raccolta delle opinioni degli utenti (**interviste e questionari**).

- Metodi misti, ottenuti da combinazioni dei precedenti; richiedono la partecipazione di esperti ed utenti.

I parametri che guidano nella scelta di un particolare metodo di valutazione piuttosto che di un'altro includono:

- tipo di informazione richiesta;
- caratteristiche degli utenti, delle loro attività e nell'ambiente in cui operano;
- natura del sistema o del tipo di specifica che si sta considerando;
- stadio nel ciclo di vita che si sta valutando;
- il fatto se si voglia o meno ottenere validità statistica;
- le risorse disponibili, in termini di esperienza e disponibilità di valutatori, equipaggiamenti, tempo e denaro.

6.1 I dieci principi di Nielsen

E' un metodo di **valutazione euristica**, cioè basato su alcuni principi ragionevoli dettati dal buon senso. Questi principi non sono universalmente riconosciuti: nella relazione si analizzeranno e applicheranno quelli proposti da Nielsen, uno studioso di HCI.

I principi vengono verificati da un esperto di HCI, che prima prende familiarità con l'interfaccia del sistema e successivamente la analizza sistematicamente. L'aderenza ai principi aiuta l'esperto a focalizzare l'attenzione sulle problematiche chiave dell'usabilità. E' accettabile che l'esperto conosca parzialmente il dominio in cui opera il programma, ma è necessario che non sia a conoscenza del contesto specifico nel quale viene utilizzato, in modo tale che le sue osservazioni non vengano polarizzate.

Per come è stato definito, il processo di valutazione non richiede la presenza dell'utente. L'esperto svolge una o più esplorazioni dell'interfaccia, con i due fini di ispezionare il suo flusso da una configurazione alle successive e analizzare ogni singola configurazione. Il risultato previsto è un rapporto contenente l'elenco di problemi di usabilità riscontrati, ognuno dei quali reca una motivazione precisa e la lista dei principi euristici violati.

In genere un singolo esperto non trova tutti i problemi, ma solo il 35% di essi. Per avere una maggiore copertura è possibile interpellare più esperti, con la condizione essenziale che essi non si comunichino le rispettive osservazioni. Ovviamente più esperti possono identificare lo stesso problema, dunque i loro risultati saranno parzialmente sovrapposti. Si è mostrato sperimentalmente che cinque esperti riconoscono congiuntamente circa il 75% degli errori che affliggono un sistema.

In questo lavoro e' necessario *essere specifici*, identificando ogni problema *separatamente*, altrimenti si può incorrere di nuovo nello stesso problema, anche se si riprogetta l'intero sistema. In ogni caso avere una lista organica permette di confinare localmente i problemi, nelle sedi esatte dove si presentano.

Questi principi vanno considerati sia durante la fase di valutazione, sia durante quella di progettazione, per rendere remota la possibilità di ottenere un sistema inusabile già dalle prime fasi del progetto. Quando tutte le informazioni sono state

raccolte, il team di progettazione potrà determinare quali sono le più importanti, che riceveranno la sua attenzione.

Prima di vedere i principi in dettaglio è utile soffermarsi brevemente sul concetto di tempo di risposta.

Il tempo di risposta

Chiamiamo **tempo di risposta** l'intervallo di tempo che passa tra l'istante in cui l'utente esegue una certa azione sul sistema e l'istante in cui il sistema *reagisce* a tale azione.

In un programma *interattivo*, come un word processor, il tempo di risposta può essere quello trascorso tra la pressione di un tasto sulla tastiera e l'apparizione del corrispondente carattere sullo schermo. Nei programmi *batch* il tempo di risposta è quello che intercorre tra l'immissione dei dati nel sistema e la fornitura dei relativi risultati, ed è esattamente pari al loro tempo di esecuzione.

Alcune considerazioni sull'ordine di grandezza di questo parametro:

- Quando l'utente opera in un ambiente interattivo è necessario che il tempo di risposta sia particolarmente breve (minore di un secondo), in modo che l'utente rimanga coinvolto e non si distraiga.
- E' tuttavia bene che il tempo di risposta non sia troppo piccolo, perché in alcune situazioni l'utente potrebbe non accorgersi di certe informazioni che il sistema gli invia, perdendole definitivamente (ad esempio il rapido lampeggio della voce **Copy** nei menu dei sistemi Mac potrebbe non essere avvertito). Dunque la reazione del sistema dovrebbe permanere almeno per alcuni decimi di secondo.

Si dice che il tempo di risposta è *ragionevole* se entrambe le regole sono soddisfatte. Ci sono particolari situazioni in cui il tempo di risposta è un parametro critico per avere un'interazione efficace col sistema, ad esempio nella *realtà virtuale*.

6.1.1 Principio 1: visibilità dello stato del sistema

Bisogna garantire che l'utente riceva un *feedback adeguato* sullo stato del sistema, in ogni situazione. Il feedback è costituito dall'insieme di segnali che il sistema invia all'utente per informarlo degli esiti dei comandi appena eseguiti e, più in generale, dello stato corrente della macchina.

- Se l'utente sospende il lavoro al calcolatore avere ben visibile lo stato del sistema gli permette di ottenere una veloce ripresa, senza obbligarlo a ricordare in che punto esatto aveva interrotto.

- Soluzioni che consentano un feedback *continuo nel tempo* sono da preferire perché l'utente viene continuamente aggiornato della situazione corrente e, se necessario, può correggerla rapidamente.

In quest'ottica, per operazioni che richiedono almeno qualche secondo, è ormai prassi comune quella di visualizzare una finestra contenente un'indicatore di progresso e un tasto **Annulla**, per dare la possibilità di bloccare l'elaborazione.

L'indicatore di progresso di solito mostra la percentuale di tempo trascorsa dall'inizio dell'elaborazione, calcolata rispetto al tempo presunto necessario per portarla a termine.

- In caso di fallimento di un'operazione deve esserci un *feedback negativo* verso l'utente, con specifica precisa della causa del fallimento (ad esempio "Questo bancomat è fuori uso perché le banconote sono esaurite").

6.1.2 Principio 2: adeguare il sistema al mondo reale

Deve colloquiare con l'utente attraverso il *suo dialetto*, evitando equivoci. Per realizzare questa situazione è necessario vedere il colloquio dal punto di vista dell'utente. Secondo quest'ottica il messaggio "Lei ha appena comperato 1000 azioni" è senz'altro migliore di "La banca le ha appena venduto 1000 azioni".

- Eventuali regole usate dal sistema per troncane i nomi devono essere semplici e chiare.
- E' bene sfruttare conoscenze che l'utente già possiede e utilizzare analogie e metafore a lui note.
- Ci deve essere una corrispondenza diretta tra i documenti cartacei del mondo reale e le *fill-in form* del mondo virtuale.
- L'utente deve disporre di *diverse alternative* in ogni stadio del lavoro, ma non troppe. In questo contesto per più alternative si intende la possibilità di attivare uno fra più comandi, più voci di menu o più strumenti, nello stesso istante di tempo.

6.1.3 Principio 3: controllo e libertà all'utente

Lasciare una via di fuga chiara all'utente: in ogni situazione l'utente deve poter interrompere immediatamente l'operazione appena intrapresa e possibilmente anche annullare gli effetti di quelle eseguite precedentemente. Dunque:

- predisporre un comando di **Undo**;
- se l'operazione richiede molto tempo prevedere un indicatore di progresso con un pulsante di arresto;
- lasciare il più possibile libertà di controllo all'utente.

6.1.4 Principio 4: assicurare consistenza

Con **consistenza** si intende l'uniformità tra applicazioni e nella stessa applicazione di certe caratteristiche, di certi comportamenti o di certe proprietà. La consistenza ha dunque aspetti multidimensionali e perciò si parlerà di consistenza:

- *nella notazione*, cioè nei nomi e nelle abbreviazioni;
- *nel linguaggio*, sia riguardo alla sintassi che ai suoi aspetti semantici;
- nella filosofia di dialogo con l'utente, compresi gli aspetti iconologici;
- *nelle azioni*, per operazioni simili;
- nell'*organizzazione funzionale*;
- nell'*organizzazione spaziale*;
- nell'*organizzazione temporale*.

La consistenza aiuta sia il lavoro dell'utente che quello del progettista del sistema. Il principio di consistenza può essere violato se l'azione intrapresa è *distruttiva*: ad esempio è bene che il trascinarsi di un file sul cestino causi la richiesta di conferma da parte del sistema, comportamento da non tenere durante i normali trascinalamenti.

6.1.5 Principio 5: prevenzione degli errori

Adottare tutti gli accorgimenti e tutte le tecniche convenienti affinché:

- il sistema non induca l'utente in errore;
- gli errori non causino danni irreversibili.

6.1.6 Principio 6: riconoscimento piuttosto che richiamo

Il sistema dovrebbe essere progettato in modo che l'utente abbia chiaramente visualizzate, in ogni istante, tutte le possibilità che ha a disposizione (cioè tutte le funzionalità che può sfruttare). Evitare il *richiamo*, cioè il ricorso alla memoria.

6.1.7 Principio 7: flessibilità ed efficienza d'uso

I modi in cui l'utente può interagire con il sistema dovrebbero essere vari e studiati tenendo conto che *l'utente impara*. Dunque:

- abbreviare i messaggi che vengono visualizzati;

- predisporre degli *acceleratori* (ad esempio la voce di menu **Chiudi finestra**, richiamabile anche direttamente tramite la combinazione di tasti **Alt-F4**) in modo che l'utente possa invocare velocemente le relative funzionalità non appena li memorizza, senza annoiarsi;
- dare la possibilità di associare *macro* (sequenze di comandi base) a certi tasti o a certe combinazioni di tasti;
- fare in modo che le funzionalità usate più di frequente possano venire invocate con i metodi più rapidi, come la pressione di un singolo tasto;
- permettere il *type ahead*, cioè dare la possibilità di accodare in un buffer i comandi che l'utente immette nel sistema in modo che egli possa inserire sequenze di comandi più velocemente di quanto il sistema li esegua;
- sfruttare l'ipertestualità;
- permettere il riuso delle storie passate (ad esempio in un editor di testo listare gli ultimi file elaborati sotto forma di voci nel menu **File**, in modo che possano essere riaperti più velocemente, senza dover passare attraverso il comando **Apri**).

6.1.8 Principio 8: estetica e disegno minimalisti

La comunicazione con l'utente, intesa come l'insieme dei contenuti testuali che gli vengono mostrati, deve essere:

- semplice e naturale;
- non ambigua e sintetica: non deve contenere informazioni irrilevanti che diminuiscano la leggibilità;
- strutturata e organizzata in modo che le scritte vengano raggruppate logicamente;
- sensibile al contesto, cioè che rifletta la situazione e le possibilità correnti ad ogni istante (ad esempio realizzata in modo che le scritte vengano aggiunte e tolte dinamicamente).

6.1.9 Principio 9: messaggi d'errore

I messaggi d'errore che vengono presentati all'utente devono essere:

- scritti in un linguaggio semplice e senza codici oscuri;
- precisi circa l'errore rilevato (ad esempio "Non puoi aprire il documento perché non esiste") e non vaghi;

- costruttivi, cioè studiati in modo che aiutino a risolvere il problema;
- educati e non accusatori (ad esempio “Lei ha commesso un errore fatale”).

Eventualmente prevedere la possibilità di mostrare messaggi a più livelli per gli errori più impegnativi, in modo tale che l’utente possa limitare la propria indagine ai livelli di sua comprensione.

6.1.10 Principio 10: aiuti e documentazioni

Teoricamente un’interfaccia utente ben realizzata dovrebbe rendere inutili gli aiuti, ma l’evidenza sperimentale dimostra che una situazione del genere è impossibile a verificarsi. Gli aiuti devono essere sempre ben focalizzati sulla situazione nella quale sono stati invocati.

E’ necessario tenere conto che l’utente medio non legge i manuali ma preferisce piuttosto mettersi subito al lavoro con il sistema e procedere per tentativi, seguendo la filosofia dell’*imparare facendo*. Pertanto è utile interagire con l’utente cercando di proporgli dei *percorsi di apprendimento*.

Anche il sistema di help e la documentazione vanno valutati rispetto alla loro usabilità.

6.2 Metodo Walkthrough

E’ un metodo di **valutazione predittiva**, nel quale uno o più esperti simulano l’utente, cercando di compiere le stesse azioni che egli compirebbe. Prerequisiti essenziali per procedere sono:

1. Disponibilità del prototipo del sistema, anche sotto forma di pura descrizione. Non deve essere completo, solo abbastanza dettagliato.
2. Esistenza del prototipo dell’utente o almeno di una sua descrizione approfondita, proveniente dall’**analisi dell’utente**. Deve illustrare in particolare la sua esperienza e le sue conoscenze.
3. Esistenza del compito o di un insieme di compiti che l’utente esegue, risultato dell’**analisi dei compiti**. Devono essere compiti rappresentativi, che la maggior parte degli utenti intende svolgere.
4. Esistenza della lista di azioni elementari necessarie per svolgere ciascun compito, costruita considerando il prototipo dell’utente specificato al secondo punto.

L’esperto riproduce ogni azione e la valuta ponendosi queste domande:

1. L’utente ha problemi nel capire cosa fare ?
2. L’utente ha problemi nell’individuare l’azionamento ?

3. L'utente ha problemi nell'eseguire l'azione ?
4. L'utente ha problemi nel recepire il feedback dell'azione ?

Durante queste prove il valutatore redige un documento nel quale descrive le attività svolte e in cui raccoglie organicamente tutte le sue osservazioni. Il documento contiene:

- il nome dell'esperto;
- il prototipo del sistema;
- il prototipo dell'utente;
- l'insieme dei compiti simulati;
- una lista di azioni per ciascun compito;
- un'elenco dei problemi riscontrati.

E' vitale documentare l'analisi, specificando bene ciò che è buono e ciò che deve essere migliorato. Può essere utile stimare la severità e la frequenza di ciascun problema, accompagnandole anche da una impressione della sua serietà dal punto di vista degli utenti. Queste informazioni aiuteranno i progettisti nel decidere le priorità degli interventi per la correzione del sistema.

Questa valutazione è poco costosa, ma richiede parecchio tempo per essere portata a termine, se si intendono analizzare tutti i compiti previsti per l'utente. Perciò a volte si preferisce analizzare un sottoinsieme dei compiti che l'utente esegue, considerato sufficiente perché ritenuto particolarmente efficace nel portare alla luce eventuali problemi.

6.3 Interviste e questionari

Una evidente mancanza dei metodi di valutazione in economia è che, non richiedendo la presenza fisica degli utenti, non possono raccogliere le *opinioni soggettive* che questi hanno nei confronti del sistema o di un suo prototipo. Tenere conto delle loro opinioni durante le varie fasi dello sviluppo è molto utile perché può fare risparmiare risorse, in quanto funzionalità non usabili, non necessarie, o inutili non verranno implementate. In altre parole, avere un feedback dagli utenti distribuito nel tempo durante la progettazione assicura che il prodotto finale sarà accettato da loro; inoltre si sarà tenuto conto anche della loro attitudine, parametro che avrà grande influenza nel decretare il successo del sistema.

Le interviste ed i questionari sono entrambi metodi per raccogliere informazioni soggettive sulle preferenze degli utenti, ma differiscono nell'esperienza richiesta agli esperti che vogliono organizzarli, nello stile di presentazione necessario e, infine, nella flessibilità con cui è possibile porre le domande. In particolare, le interviste danno informazioni qualitative, mentre i questionari danno indicazioni un poco più quantitative.

	Osservazioni	Interviste e questionari	Esperimenti	Metodi interpretativi	Metodi predittivi
Migliorare il sistema	x	X	X		x
Capire il mondo reale	X	X		X	
Confrontare progetti	X	X	x		X
Verificare rispetto degli standard			X		

Tabella 3: Scelta dei metodi di valutazione

7 Scelta dei metodi di valutazione per Vis5D

Durante il corso sono stati esaminati molti metodi di valutazione, alcuni dei quali rivisti brevemente in precedenza. Ora è il momento di applicarli, scegliendone uno o più. Come visto nel capitolo precedente, la scelta dei metodi di valutazione da impiegare si svolge considerando: le caratteristiche degli utenti, della loro attività, dell'ambiente nel quale operano, la natura del sistema, le risorse disponibili, l'esperienza dei valutatori nonché le finalità che si intendono perseguire. Quest'ultime, decise in conclusione dell'intervista col cliente, sono:

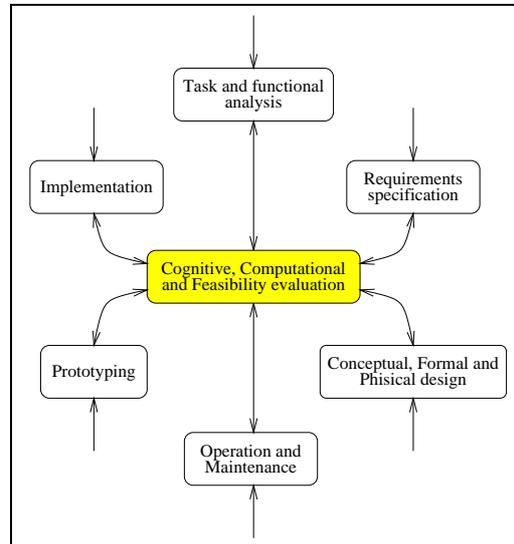
- **capire il mondo reale** nel quale gli utenti operano, per essere in grado di produrre una lista di problemi approfondita;
- **migliorare il sistema esistente**, usando come guida la lista individuata al passo precedente da esperti e utenti.

La tabella 3, estratta da uno¹ dei libri di testo del corso e al quale si rimanda per approfondimenti, suggerisce i metodi di valutazione da usarsi, in funzione delle ragioni che giustificano la valutazione stessa. Le **X** contrassegnano le scelte consigliate, mentre delle **x** individuano altre scelte possibili, ma meno indicate. In accordo con la tabella, considerando le risorse disponibili e le esigenze di capire il mondo reale e migliorare il sistema esistente, sono stati scelti questi quattro metodi di valutazione:

- metodo euristico;
- metodo predittivo;
- esperimenti integrati da questionari.

¹J. Preece, Y. Rogers, H. Sharp, D. Benyon, S. Holland e T. Carey (1994), Human-Computer Interaction, Addison-Wesley

8 Valutazione walkthrough di Vis5D



Prima di procedere nella fase di analisi vera e propria si definiscono con precisione gli ingredienti necessari all'applicazione del metodo, come richiesto dai suoi prerequisiti.

8.1 Prototipo del sistema

E' l'implementazione di Vis5D di cui si dispone.

8.2 Prototipo utente

Sono degli studenti, il cui profilo è stato definito chiaramente in precedenza nell'analisi dell'utente. Si rammenta solo che essi hanno seguito un corso universitario ed una serie di esercitazioni: durante il corso hanno appreso i concetti teorici fondamentali del dominio, sia dal punto di vista chimico che meteorologico, mentre durante le esercitazioni gli sono state illustrate le funzionalità fondamentali di Vis5D.

8.3 Compiti

Poiché una valutazione di questo tipo è sì poco costosa da un punto di vista economico, ma richiede molto tempo per essere completata, si è preferito limitare l'analisi ad un solo compito, descritto di seguito. Secondo gli autori questo compito richiede un'interazione abbastanza approfondita col sistema e dunque dovrebbe dare risultati significativi. Il compito è sintetizzabile nei seguenti quattro obiettivi:

- avvio del programma ed apertura del file **LAMPS.v5d**;

- visualizzazione del campo di vento orizzontale alla quota di 2km, da un arbitrario punto di vista 3D, a tre diverse ore della giornata;
- salvataggio di un'immagine per ciascuna ora;
- terminazione del programma.

8.4 Liste di azioni

Ecco la lista di azioni in cui si decompone l'unico compito previsto. Per maggiore chiarezza, le azioni logicamente correlate sono state raggruppate, giungendo alla formazione di sette gruppi. Come *stato iniziale* è stato scelto l'ambiente X Window, senza nessuna shell aperta.

1. Lanciare Vis5D: aprire una shell, digitare il nome del programma, specificare come unico parametro sulla linea di comando il file **LAMPS.v5d** e confermare con **Invio**.
2. Selezionare il campo di vento orizzontale, premendo il pulsante **Hwind1**.
3. Portare il puntatore nella finestrella che compare una volta premuto il pulsante **Hwind1** e manipolare la scroll bar **Height** ivi contenuta, sino ad imporre un'altezza di 2km al grafico.
4. Cambiare il punto di vista, portando il puntatore sul **3-D Display** e premendo opportunamente i suoi tre tasti mentre lo si muove.
5. Utilizzando i tasti del mouse, agire sul pulsante **STEP** fino a visualizzare sul **3-D Display** la prima delle tre ore di interesse.
6. Premere il pulsante **SAVE PIC**, selezionare il formato **ppm**, inserire il nome del file e premere il pulsante **SAVE**. Ripetere questo passo ed il precedente per gli altri due orari.
7. Chiudere Vis5D, tramite la pressione del pulsante **EXIT**.

8.5 L'analisi in dettaglio

Seguono, in dettaglio, i passi simulati dai tre esperti. Ogni situazione problematica è identificata da un **SI** e vengono specificati il numero di esperti che l'hanno individuata e il numero totale di esperti che hanno esaminato l'intera azione alla quale si riferisce. Un breve commento spiega con precisione il problema.

8.5.1 Azione 1: partenza del programma.

Problemi nel capire cosa fare:

SI (1/2): è necessario che l'utente conosca le basi dell'interazione in una shell Unix, ed in particolare il funzionamento della linea di comando. Si prevede che solo pochi degli studenti presi in esame saranno in grado di padroneggiare questi argomenti.

Per sapere che è necessario passare come parametro il nome del file contenente il data set, lo studente ha tre possibilità:

1. ricordarselo (uso della memoria);
2. cercare nella documentazione (**README**);
3. interpretare il messaggio di errore che il programma mostra prima di terminare, se avviato senza parametri.

Problemi nell'individuare l'azionamento:

NO: deve semplicemente inserire il nome del file e premere **Invio**.

Problemi nell'eseguire l'azione:

NO.

Problemi nel recepire il feedback:

SI (1/2): il programma parte solo se il file è leggibile, altrimenti viene visualizzato il messaggio di errore **File not found**, che è piuttosto ambiguo in quanto non chiarisce bene il problema.

8.5.2 Azione 2: attivazione del pulsante Hwind1.

Problemi nel capire cosa fare:

SI (2/3): il fatto che si debba selezionare il pulsante **Hwind1** non è ovvio, l'utente deve ricordarselo o almeno fermarsi e consultare la documentazione.

Problemi nell'individuare l'azionamento:

SI (3/3): perché deve cercare il pulsante giusto tra i molti disponibili (disorientamento).

SI (2/3): perché nel pannello principale sono presenti sia **Hwind1** che **Hwind2** (confusione).

Problemi nell'eseguire l'azione:

NO.

Problemi nel recepire il feedback:

SI (2/3): non è immediato capire il significato dei controlli **Scale**, **Density**, **Height** e soprattutto **SFC**, presenti nella finestrella che appare una volta attivato il pulsante **Hwind1**.

8.5.3 Azione 3: selezione della quota.

Problemi nel capire cosa fare:

NO: l'utente sa che deve cambiare l'altezza del grafico.

Problemi nell'individuare l'azionamento:

SI (1/3): l'utente deve cercare, tra le tante finestre aperte, quella che contiene la scroll bar **Height** che controlla l'altezza, che compare una volta selezionato il pulsante **Hwind1** (disorientamento).

SI (2/3): la finestrella può risultare nascosta dalle altre.

Problemi nell'eseguire l'azione:

NO: si opera con i controlli in modo convenzionale.

Problemi nel recepire il feedback:

SI (2/3): anche se il livello del grafico viene aggiornato mentre l'utente altera il valore della scroll bar, l'effetto può essere invisibile da certi punti di vista, ad esempio in **Top** (dall'alto verso il basso, impostazione di default all'avvio).

8.5.4 Azione 4: cambio del punto di vista.

Problemi nel capire cosa fare:

NO: si tratta di una manipolazione diretta nel **3-D Display**.

Problemi nell'individuare l'azionamento:

SI (1/3): il modo di procedere non è ovvio, nonostante la presenza del box bianco di help nel mezzo del pannello principale.

Problemi nell'eseguire l'azione:

SI (2/3): non è facile spostarsi nel mondo virtuale della rappresentazione.

SI (2/3): è impossibile impostare un punto di vista esatto, in termini di certi angoli e certe coordinate.

SI (2/3): E' necessario utilizzare tutti i tre tasti del mouse; gli utenti che sono sprovvisti di mouse a tre tasti sono costretti ad emulare il tasto centrale premendo contemporaneamente i due laterali.

Problemi nel recepire il feedback:

SI (1/3): il display viene aggiornato continuamente mentre l'utente agisce su di esso, però si leggono poco i numeri, la data è indecifrabile e il logo di Vis5D può dare fastidio.

8.5.5 Azione 5: selezione dell'ora.

Problemi nel capire cosa fare:

SI (1/3): l'utente preferirebbe inserire l'ora direttamente, in una apposita input box. Dopo alcuni istanti di meditazione, diversi tentativi di trovare un comando adeguato, oppure una rapida ricerca nel manuale,

probabilmente converrà che ciò non è possibile e che può selezionare l'ora d'interesse soltanto indirettamente, premendo ripetutamente un certo pulsante.

SI (1/3): l'utente può tentare di interagire direttamente con l'orologio.

Problemi nell'individuare l'azionamento:

SI (2/3): deve cercarlo tra i molti pulsanti.

SI (1/3): il nome **STEP** non è chiarificatore della funzionalità che scatena.

SI (1/3): la presenza dei pulsanti **ANIMATE** e **ANIM** può disorientare l'utente, che potrebbe selezionarli.

SI (1/3): il pulsante **CLOCK** è fuorviante.

SI (1/3): il pulsante **STEP** è fisicamente lontano dall'orologio e dal **3-D Display**, sui quali agisce.

Problemi nell'eseguire l'azione:

SI (1/3): l'utente deve premere il tasto sinistro del mouse fino a visualizzare l'ora di interesse sull'orologio. Se sbaglia, avanzando troppo, è probabile che ritenti facendo passare tutti i time step rimanenti e ricominciando dal primo, perché non ricorda le funzionalità associate agli altri due tasti del mouse. Sono due *comandi invisibili*, che fanno tornare al primo fotogramma del data set oppure a quello precedente, rispettivamente.

SI (1/3): lo stesso risultato si può ottenere col pulsante **ANIMATE**, ma questa alternativa richiede precisione nella disattivazione del controllo, pena il sorpasso del time step desiderato.

Problemi nel recepire il feedback:

SI (1/3): anche se **3-D Display** e orologio si aggiornano immediatamente, si può avere poca percezione dell'avanzamento del tempo. L'orologio è particolarmente antipatico.

8.5.6 Azione 6: salvataggio di un'immagine.

Problemi nel capire cosa fare:

SI (1/3): la funzionalità associata al pulsante **SAVE PIC** dovrebbe essere ovvia, tuttavia la mancanza dei due puntini alla fine del nome del pulsante può indurre incertezza nell'utente circa il fatto che questo sia effettivamente il pulsante giusto da attivare. Infatti, convenzionalmente i due puntini indicano la presenza di un'ulteriore, successivo dialogo, necessario in questo caso per poter inserire il nome ed il tipo del file da creare.

SI (2/3): il pulsante **SAVE PIC** può essere confuso con il pulsante **SAVE..**, che invece salva il contesto di Vis5D (colori, tipi di grafici attivi e altri settaggi).

Problemi nell'individuare l'azionamento:

SI (2/3): il pulsante **SAVE PIC** va cercato; la sua posizione è particolarmente infelice perché distante dal **3-D Display**.

Problemi nell'eseguire l'azione:

SI (2/3): è possibile salvare immagini in soli due formati, **xwd** e **ppm**, entrambi poco conosciuti.

SI (2/3): non è chiaro in quale punto del file system verrà memorizzata l'immagine, e non risulta possibile cambiare path.

SI (1/3): il fatto che, per il file da salvare, venga riproposto lo stesso nome usato per il precedente salvataggio comporta un rischio di sovrascrittura dell'immagine precedente.

SI (1/3): la pressione del tasto **Invio** non invoca nessuna azione di default. Non si capisce se l'estensione del file venga aggiunta automaticamente dal sistema oppure se è necessario specificarla.

Problemi nel recepire il feedback:

SI (2/3): non è presente alcuna forma di feedback, soprattutto nel caso esista già un file con quel nome.

8.5.7 Azione 7: chiusura del programma.

Problemi nel capire cosa fare:

NO.

Problemi nell'individuare l'azionamento:

SI (1/3): anche se il pulsante **EXIT** è autoesplicativo, esso è disperso tra i molti presenti nel pannello principale, posizionati senza alcun criterio apparente.

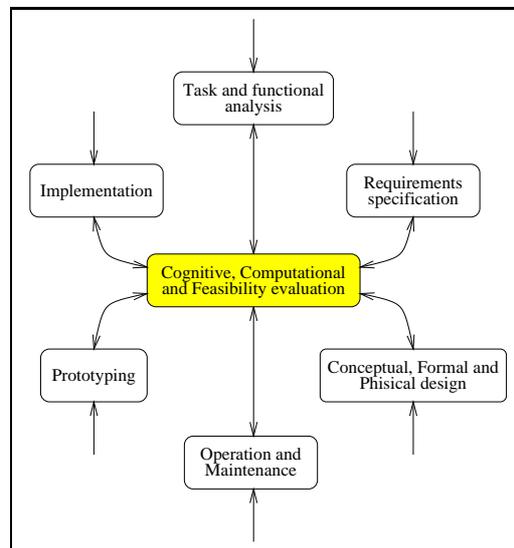
Problemi nell'eseguire l'azione:

NO.

Problemi nel recepire il feedback:

SI (1/3): il programma termina immediatamente, senza alcuna richiesta di conferma.

9 Valutazione euristica di Vis5D



Si rammenta che la valutazione euristica prevede che gli esperti esaminino separatamente il sistema basandosi su di un insieme di principi guida atti ad evidenziare problemi di usabilità durante l'uso del software.

Tali principi possono essere scelti a discrezione degli esaminatori, ma vanno comunque concordati prima di iniziare l'esame. Affinché la valutazione risulti la più obiettiva possibile, è fondamentale che ogni esperto analizzi il sistema singolarmente, evitando così di subire inconscie polarizzazioni da parte degli altri esaminatori. Al termine della valutazione, i risultati ottenuti dai vari esperti dovranno essere aggregati.

Ai fini della significatività complessiva del lavoro compiuto è opportuno avere a disposizione diversi esperti. Nielsen, infatti, ha osservato sperimentalmente che maggiore è il numero degli esperti migliore è la percentuale di problemi riscontrati. Tuttavia tale miglioramento non è direttamente proporzionale al numero degli esperti coinvolti, ma tende a raggiungere un valore di saturazione oltrepassati i sei esaminatori.

Il nostro gruppo di lavoro è composto da soli tre esaminatori.

9.1 La valutazione in dettaglio

Per la valutazione di Vis5D si è deciso di utilizzare alcuni principi già esistenti, conosciuti correntemente sotto il nome di **dieci principi di Nielsen**, ben illustrati precedentemente. Per ogni problema è prevista una spiegazione e vengono riportati il numero di esperti che l'hanno individuato e il numero totale di esperti che hanno considerato il principio.

9.2 Principio 1: visibilità dello stato del sistema

- (1/3) Stato del sistema molto frammentato, distribuito tra controlli spazialmente troppo lontani e in tre finestre diverse:
 - sui pulsanti del pannello principale, che cambiano aspetto quando la loro funzionalità è attiva;
 - sui controlli delle finestrelle secondarie, quando sono aperte;
 - nel **3-D Display**.
- (2/3) Mancanza di riferimenti spaziali continui: sarebbe auspicabile disporre di coordinate spaziali in termini di latitudine e longitudine, ogni volta che l'utente senta l'esigenza di orientarsi nel display 3D.
- (2/3) I miseri riferimenti spaziali presenti sono oltretutto poco leggibili: numeri piccoli e caratteri non adeguati.
- (3/3) I riferimenti temporali presenti sono visualizzati in formato poco immediato. La data è mostrata in formato Giuliano e la sua posizione sullo schermo è scomoda.
- (2/3) Il logo di Vis5D, presente in basso a destra nel **3-D Display**, è decisamente ingombrante, oltre che inutile.
- (1/3) Manca un'indicazione percentuale dell'intensità dello zoom eventualmente utilizzato.
- (1/3) Quando l'utente seleziona una visualizzazione abbastanza pesante da calcolare, il sistema *si congela* e spende diversi secondi in elaborazioni, prima di mostrare il risultato. Possibile induzione di confusione nell'utente.

9.3 Principio 2: adeguare il sistema al mondo reale

- (1/3) E' impossibile inserire coordinate e angoli esatti.
- (2/3) Il significato dei pulsanti non è sempre chiaro, ad esempio le azioni scatenate da **GRID #'s**, **CONT #'s** e **SFC** sono imprevedibili.
- (2/3) Esistono coppie di pulsanti con significati equivoci: **SAVE..** e **SAVE PIC**, **CLOCK** e **STEP**, oppure **ANIM** e **ANIMATE**.

9.4 Principio 3: controllo e libertà all'utente

- (2/3) Non esiste la possibilità di annullare l'ultima azione intrapresa (undo).
- (2/3) Non è possibile compiere azioni utilizzando anche scorciatoie via tastiera.

- (1/3) Operazioni temporalmente dispendiose, come il grafico volumetrico, non sono interrompibili.
- (1/3) Una volta attivata una delle funzionalità presenti nella parte centrale del pannello principale, per disabilitarla è necessario selezionare il pulsante **Normal**: via di fuga non chiara.
- (2/3) La gestione del software è quasi completamente operata via mouse; è negato anche l'utilizzo delle usuali frecce, che in alcuni casi renderebbe i movimenti più precisi.
- (1/3) A volte le vie d'uscita sono impraticabili: se l'utente seleziona il pulsante **DISPLAY** e decide di non voler operare alcun cambiamento può rimanere incastrato nel grande dialogo mostrato, se lavora in bassa risoluzione e non riesce a raggiungere il pulsante di chiusura posto in fondo alla finestra.

9.5 Principio 4: assicurare consistenza

- (3/3) L'organizzazione dei pulsanti è concettualmente disordinata: risultano troppo ammassati e quelli riguardanti funzionalità eterogenee sono spazialmente vicini.
- (2/3) I controlli sono distanti dal luogo dove producono effetti. Ad esempio i pulsanti **TOP**, **SOUTH** e **WEST** dovrebbero stare più vicini al **3-D Display**.
- (1/3) Sono presenti funzionalità anche quando sono inutilizzabili: ad esempio il pulsante **Hwind1** è visibile ed attivabile anche quando un campo di vento non è definito.
- (2/3) Non è facile riconoscere la tipologia dei pulsanti: mutuamente esclusivi, opzionali, sovrapponibili. Vi è inconsistenza con il *look and feel* delle applicazioni moderne in ambiente X Window.
- (2/3) L'utente non viene informato circa la gerarchia di "importanza" dei controlli.
- (1/3) Non è chiaro se la pressione di un pulsante innesca subito la relativa funzionalità, oppure se si limita ad aprire una finestra per chiedere informazioni. La convenzione di usare puntini per indicare ulteriori dialoghi non è rispettata: inconsistenza con le altre applicazioni, che può generare equivoci.
- (3/3) Quando l'utente chiude una qualsiasi finestra utilizzando l'icona di chiusura nella sua cornice, in realtà provoca la terminazione del programma: inconsistenza con le altre applicazioni.
- (1/3) Il pulsante **EXIT** è situato in una posizione poco adeguata al suo scopo.
- (3/3) Gran parte dello spazio disponibile sul video non è sfruttato.

- (1/3) Il **3-D Display** è troppo piccolo ed è difficile da ingrandire. La sua posizione è scomoda.
- (1/3) Il parametro d'avvio **-full**, che aumenta le dimensioni del **3-D Display**, provoca anche la sovrapposizione delle varie finestre, rendendo ingestibile la situazione.
- (3/3) La gestione delle finestre è pessima: sono visualizzate in zone spazialmente scomode che inducono l'utente ad ignorarle; non sempre presentano una chiara possibilità d'uscita.
- (2/3) Le finestrelle hanno posizioni infelici: la loro comparsa non cattura l'attenzione dell'utente. A volte sono parzialmente nascoste dalle tool bar del window manager di X Window; in caso ve se siano diverse aperte, esse si nascondono vicendevolmente.
- (2/3) Il box bianco riservato alla descrizione delle azioni associate ai tre tasti del mouse, variabili in funzione del pulsante attivo, è posto in un punto scomodo per l'occhio. E' poco chiaro e risulta pesante, se sempre presente. Sarà ridondante quando l'utente avrà acquisito confidenza.

9.6 Principio 5: prevenzione degli errori

- (1/3) Cattive abbreviazioni possono indurre l'utente in errore: ad esempio i pulsanti **ANIM** e **ANIMATE** possono essere scambiati.
- (2/3) Non vengono mai chieste conferme in seguito a decisioni importanti, come il salvataggio di un file che provoca una sovrascrittura.
- (2/3) Nelle finestrelle associate ad alcuni pulsanti, non sono indicati i range che possono assumere i valori introducibili dall'utente.
- (3/3) Non vi sono messaggi d'errore scritti.
- (1/3) Non sono sfruttati né suoni né colori per avvertire l'utente di eventuali situazioni di errore o di elaborazione in corso;
- (1/3) Se l'utente è un mancino può darsi che inverta l'uso dei pulsanti esterni del mouse, pertanto l'help dato nel box bianco è deviante.

9.7 Principio 6: riconoscimento piuttosto che richiamo

- (2/3) E' totalmente trascurato l'uso di icone che possano suggerire il significato dei pulsanti o informare sullo stato corrente in cui si trova il sistema.
- (3/3) L'utente deve ricordare molte informazioni a memoria, soprattutto se intende gestire in maniera completa il sistema. Come risultato, chi usa raramente Vis5D continuerà a trovarlo difficile.

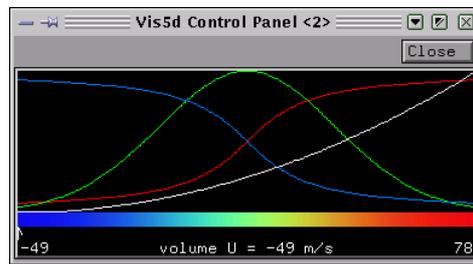


Figura 10: Finestrella che compare nella rappresentazione volumetrica

- (2/3) La finestra di aiuto per la gestione del mouse non è intuitiva e ha poco impatto.

9.8 Principio 7: flessibilità ed efficienza d'uso

- (2/3) Non è possibile disporre di acceleratori che si adattino al livello di familiarizzazione col sistema raggiunto dall'utente mediante la pratica.
- (1/3) Impossibile definire delle macro o riutilizzare storie passate.
- (3/3) La possibilità di creare nuove funzionalità mediante script è ridotta, poiché l'interprete integrato nel Vis5D è limitato ai costrutti fondamentali del linguaggio Tcl, almeno nella versione precompilata di cui disponiamo.

9.9 Principio 8: estetica e disegno minimalisti

- (1/3) Schermo inutilmente inflazionato da controlli.
- (3/3) Dialogo con l'utente ridotto e povero, non naturale.
- (2/3) Icone assenti.
- (1/3) Nella matrice di variabili, visualizzata sul video nella zona in basso a sinistra nel pannello principale, i nomi della terza e quinta colonna sono inspiegabilmente coincidenti (**Slice Vert.**).
- (2/3) Utilizzando la rappresentazione volumetrica, è difficile capire come manipolare la scala cromatica che appare nella finestrella aggiuntiva (vedere la figura 10). E' anche impossibile avere una scala a soli toni di grigio.

Mediante questa scala si specifica la mappatura tra i valori assumibili da una variabile e i colori da usare per il disegno del grafico volumetrico, punto per punto.

- (2/3) Il nome di diversi pulsanti non ha il potere di richiamare la loro funzionalità: tra essi menzioniamo **GRID #'s**, **CONT #'s** e **SFC**. Sono abbreviazioni o contrazioni infelici.

9.10 Principio 9: messaggi d'errore

- (3/3) Non esiste alcuna forma di avvertimento in caso di errori o malfunzionamenti del sistema.

9.11 Principio 10: aiuti e documentazioni

- (3/3) La documentazione cartacea è poca, carente e mal strutturata.
- (3/3) Non è possibile chiedere aiuto on-line.
- (1/3) Il poco help fornito on-line dal sistema, attraverso il box bianco, è solo parzialmente sensibile al contesto.
- (2/3) Esistono molti *comandi nascosti*, la cui esistenza viene conosciuta attraverso manovre casuali, tramite suggerimenti da altri utenti, oppure ancora dedotta dopo analisi approfondita del manuale.

10 Risultati delle valutazioni in economia

Questi sono alcuni dei problemi di usabilità di Vis5D, ricavati aggregando i risultati dell'applicazione dei metodi di valutazione in economia, cioè il metodo dei dieci principi e il metodo walkthrough:

- vi sono problemi di consistenza spaziale e funzionale;
- lo schermo è inutilmente sovraffollato di controlli;
- i pulsanti hanno spesso significati oscuri, che obbligano l'utente all'uso della memoria per ricordare le funzionalità a loro associate;
- l'output generato dal programma è difficile da comprendere, in generale;
- la mancanza di qualsiasi tipo di help in linea obbliga l'utente a perdere tempo nell'apertura del manuale e nella ricerca in esso;
- non vi è alcuna prevenzione degli errori;
- la manipolazione del **3-D Display** è difficoltosa.

E' da notare che i due metodi di valutazione in economia hanno prodotto delle liste di problemi simili, ma non esattamente coincidenti, poiché alcuni problemi sono stati rilevati soltanto da un unico metodo.

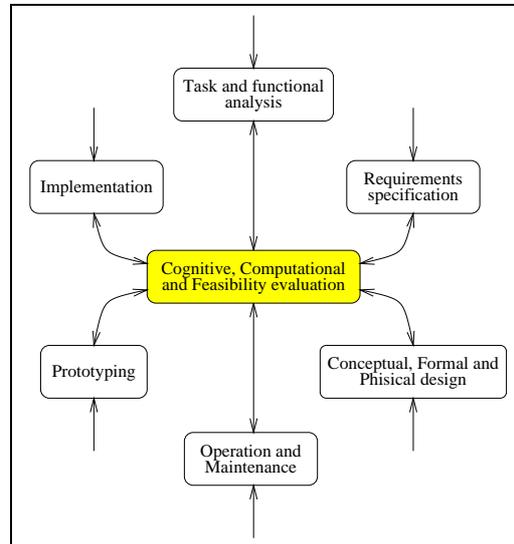
10.1 Cosa il nostro Nielsen non trova

La valutazione euristica secondo Nielsen non tiene conto del prototipo dell'utente e delle sue capacità, dunque non è in grado di identificare il problema di usabilità che si presenta all'avvio del sistema, che come visto richiede una interazione non banale con una shell Unix.

10.2 Cosa il nostro walkthrough non dice

Il metodo walkthrough non ha evidenziato la mancanza di feedback durante lunghe elaborazioni. Questo è accaduto fondamentalmente perché l'indagine è stata limitata solo ad alcune delle funzionalità presenti, che accidentalmente non producono calcoli particolarmente pesanti; un'analisi sistematica avrebbe probabilmente rilevato questo problema.

11 Esperimento di valutazione di Vis5D



Procedendo nella fase di valutazione dell'usabilità di Vis5D, è stato deciso di condurre un esperimento convocando alcune persone che, secondo gli autori, costituissero un campione abbastanza rappresentativo per la categoria degli studenti, per la quale si vuole migliorare il programma.

In tutto erano presenti sette cavie, delle quali cinque studenti di ingegneria e due giovani laureati. Ad essi è stato chiesto di realizzare il medesimo compito che gli esperti hanno simulato durante la valutazione walkthrough, perché ritenuto sufficientemente articolato ed approfondito da fornire risultati utili, nonostante la sua semplicità.

11.1 Struttura

Il test è stato suddiviso in tre parti: una prima parte collettiva di training, durata 20 minuti circa, che vedeva la presenza di tutti gli utenti e degli esperti; la seconda individuale in cui ogni utente, isolato dagli altri ma osservato dagli esperti, ha interagito col software. La terza, infine, prevedeva la compilazione di un questionario studiato al fine di completare la valutazione.

11.1.1 Prima fase: training

Ecco ciò che è avvenuto nella prima parte:

- Ci siamo riuniti tutti in un ufficio ed abbiamo esposto brevemente le finalità dell'esperimento agli studenti, assicurandoli circa le nostre intenzioni

di salvaguardare la riservatezza dei risultati degli esperimenti, qualunque essi fossero. Infine, è stata ricordata loro la possibilità di ritirarsi dal test in qualsiasi momento.

- Sono stati richiamati alcuni concetti base dell'ambito modellistico.
- E' stato introdotto velocemente il Vis5D, spiegandone le possibilità di utilizzo, accennando alle sue funzionalità base e illustrando *staticamente* alcune delle sue schermate più significative, in modo da non mostrare gli azionamenti usati per invocarle.
- Dopo aver fornito ad ogni persona alcuni fogli riassuntivi dei concetti trattati (questa documentazione è presente in appendice), è stata messa a disposizione di ciascuno una copia cartacea del manuale di Vis5D, facendo comunque presente a tutti la possibilità di poterlo consultare anche in modo informatico, visualizzando il file **README** disponibile su ogni postazione.
- Si è proseguito mostrando dettagliatamente e dinamicamente la fase di avvio del software, *partendo da una shell aperta di Linux*, che costituisce lo stato iniziale. Si è insistito particolarmente su questo punto poiché, durante la fase di valutazione in economia, è stato previsto potesse divenire fonte di complicazioni non indifferenti per utenti poco esperti di Linux.
- Dopo aver ringraziato tutti per l'attenzione concessaci, abbiamo spiegato come si sarebbe svolta la seconda parte del test e ci siamo accordati sull'ora e il luogo in cui avremmo premiato la loro disponibilità con un caffè.

11.1.2 Seconda fase: il test

A questo punto ha avuto inizio la seconda parte dell'esperimento: gli esperti si sono divisi in due gruppi e contemporaneamente, ad intervalli della durata di circa 25 minuti, due cavie sono state portate alle rispettive postazioni per il test. Sono stati utilizzati due PC aventi caratteristiche hardware comparabili e circa le stesse prestazioni. In particolare, i monitor avevano uguali dimensioni e i mouse erano entrambi a tre tasti. E' stata scartata la possibilità, concreta, di utilizzare un PC portatile, perché il suo touch pad non è comune.

Nei primi 5 minuti del test è stata offerta ad ognuno la possibilità di interagire liberamente con il Vis5D, in modo da permettergli un minimo di ambientamento sia con la macchina che con l'interfaccia del programma. Allo scadere di questo intervallo, l'utente ha potuto leggere il foglio contenente le indicazioni sul compito da svolgere e subito dopo è partito il cronometraggio del tempo. Si è tenuta traccia dell'ora d'inizio e di conclusione delle operazioni; durante questo periodo abbiamo silenziosamente seguito l'utente, appuntando in una lista le azioni che ha compiuto, includendo anche suoi eventuali commenti ad alta voce. Abbiamo altresì tenuto in conto sue eventuali consultazioni del manuale.

11.1.3 Terza fase: il questionario

Si rimanda la descrizione dettagliata di questa fase al capitolo successivo dove, dopo avere illustrato in generale l'utilità dei questionari come metodo di valutazione, presenteremo il nostro questionario e riporteremo i risultati ottenuti, commentandoli.

11.2 Risultati dell'esperimento

Di seguito sono riportati degli schemi riassuntivi dei risultati qualitativi (tabelle 4 e 5) e quantitativi (tabella 6) dell'esperimento. Le colonne identificate da **s1**, ... **s7** individuano ognuna delle sette cavie. Nelle prime due tabelle compare la lista di tutti i problemi riscontrati durante il test di usabilità; ogni difficoltà viene accompagnata dalla percentuale di studenti che sono incappati in essa. Nella terza (tabella 6) si indica, per ogni studente, il tempo complessivo impiegato per svolgere il compito ed il numero di consigli chiesti agli sperimentatori.

E' importante specificare che noi esperti ci siamo limitati ad osservare in silenzio le cavie, intervenendo solo in situazioni particolarmente critiche. Prima di iniziare l'esperimento avevamo concordato tra noi di aiutare gli utenti solo nei casi in cui si trovassero in situazione di stallo per un tempo maggiore di quello massimo stabilito per ogni azione (fissato in circa 3 minuti). Durante il test questa situazione si è presentata più volte e sempre in occasione della ricerca del pulsante per l'impostazione dell'ora.

11.3 Osservazioni

Esprimiamo alcune considerazioni, che ci sembrano indispensabili per poter trarre dei giudizi significativi da questa esperienza. Queste riflessioni sono frutto di osservazioni personali che gli esperti hanno maturato durante il corso dell'esperimento e che non sono state considerate nelle tre tabelle.

Avvio del programma

Innanzitutto, come si può notare osservando le tabelle precedenti, non si sono notati particolari difficoltà nella fase di avvio del programma. Questo dato riflette l'esperienza informatica che il campione d'utenti possiede: come si può evincere dalla prima sezione del questionario, si tratta infatti di utenti informaticamente preparati, di cui ben l'86% possiede una conoscenza discreta del sistema operativo Linux.

In realtà riteniamo che, per persone meno esperte, la fase di avvio sia molto delicata e possa creare complicazioni. A tal proposito si ricorda che Vis5D può esser lanciato soltanto caricando contemporaneamente il data set che andrà a visualizzare mentre, secondo le valutazioni da noi effettuate, sarebbe meglio permettere all'utente di eseguirlo indipendentemente dai dati che serviranno, mettendo a di-

Problemi riscontrati	s1	s2	s3	s4	s5	s6	s7	Percentuale
Avvio difficoltoso		x						14%
Uscita dal software chiudendo con l'apposita icona una finestra qualsiasi	x	x	x	x	x	x	x	100%
Problemi con la chiusura della finestra 3-D Display		x				x		29%
Problemi di uscita dopo l'attivazione del pulsante INTERPR..						x		14%
Problemi nella comprensione del significato della modalità Normal	x							14%
Difficoltà nel trovare il pulsante adatto a visualizzare il campo Hwind1 ; tentativi casuali con NEW VARS.., U, e V	x	x			x	x		57%
Dubbi riguardo al significato del pulsante Hwind2	x				x			29%
Indifferenza all'apparizione della finestra Control Panel	x	x	x	x	x	x		86%
Difficoltà nell'impostare l'altezza desiderata			x		x			14%
Incomprensione della abbreviazione SFC nel Control Panel					x	x		29%
Problemi nella manipolazione del 3-D Display (traslazione, zoom)			x	x				29%

Tabella 4: Risultati qualitativi, parte 1/2

Problemi riscontrati	s1	s2	s3	s4	s5	s6	s7	Percentuale
Incomprensione iniziale del carattere tridimensionale del 3-D Display ; confusione con un'immagine bidimensionale					x			14%
Incomprensione della funzione del pulsante CLOCK	x	x			x	x	x	71%
Tentativo di settare l'ora premendo sull'orario visualizzato a schermo	x				x		x	43%
Tentativi casuali e disperati per cambiare l'ora	x	x			x	x		57%
Impostazione ora tramite il pulsante ANIM	x					x		29%
Incomprensione della finestra bianca d'aiuto per l'uso del mouse					x	x		29%
Ambiguità tra i pulsanti ANIM e ANIMATE					x		x	29%
Tentativo di confermare il nome di un file tramite Invio da tastiera					x			14%
Salvataggio di un file in un formato non usuale	x		x			x		43%
Ambiguità tra i pulsanti SAVE.. e SAVE PIC		x	x	x	x	x		71%
Sovrascrittura di un file durante il salvataggio			x					14%
Difficoltà a trovare il pulsante EXIT	x		x					29%

Tabella 5: Risultati qualitativi, parte 2/2

Informazioni quantitative	s1	s2	s3	s4	s5	s6	s7
Tempo totale impiegato	10min	8min	15min	7min	15min	10min	12min
Tempo speso nella consultazione dei manuali	1min	-	1min	-	1min	-	1min
Numero di aiuti richiesti agli esperti	1	1	-	1	1	-	-

Tabella 6: Risultati quantitativi

sposizione una funzionalità che permetta l'apertura in un secondo momento di un qualsiasi data set.

Confusione coi pulsanti

Altro punto fondamentale riguarda i pasticci provocati dagli studenti durante la scelta dei pulsanti: tutti gli utenti non hanno eseguito con la necessaria attenzione le azioni nel loro ordine logico, forse perché agitati, forse perché poco motivati ma, con maggior probabilità, perché faticavano a orientarsi nel pannello di controllo che, come loro stessi hanno dichiarato verbalmente e scritto nel questionario, risulta disordinato e non strutturato.

Inoltre, tutte le cavie si sono trovate d'accordo nell'affermare che molti pulsanti possiedono nomi poco significativi e tutt'altro che intuitivi. Tra questi hanno menzionato **SFC**, acronimo infelice, e **INTERPR..**, che lancia l'interprete Tcl ma è facilmente confondibile con **Interpolation**. Alcuni hanno evidenziato le ambiguità tra **SAVE..** e **SAVE PIC**, tra **ANIM** e **ANIMATE**.

Difficoltà colle finestre

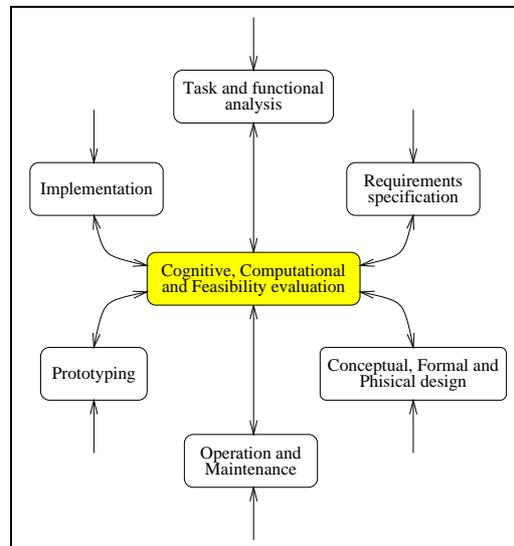
In generale, è stato notato che tutti gli utenti hanno faticato nel trovare la finestra che permette il cambiamento dell'altezza del campo di vento. Ciò è dovuto al fatto che l'apparsa di tale finestra avviene automaticamente, in seguito alla pressione del pulsante **Hwind1**; inoltre, la zona dello schermo interessata è poco osservata dall'occhio, tutto concentrato in quel momento nel percepire il feedback della visualizzazione del campo di vento nel **3-D Display**.

Per di più, la selezione contemporanea di più campi di vento apre naturalmente più finestre, che si coprono vicendevolmente e non riportano una chiara ed evidente dicitura di quale sia il campo vettoriale a cui rispettivamente si riferiscono.

Problemi col manuale

Infine, è importante riportare l'impressione che gli studenti hanno trasmesso riguardo al manuale di Vis5D, che qualcuno ha provato a consultare. Dopo alcuni istanti sprecati per cercare di orientarsi nella documentazione, tutti hanno preferito abbandonare questa strada e hanno optato per un tentativo di risoluzione dei problemi dettato dal buon senso oppure, in alcuni casi di esasperazione (capitati ad esempio durante l'impostazione dell'ora), guidato da scelte casuali dei controlli.

12 Valutazione di Vis5D tramite questionario



Per integrare la lista di problemi ricavata con i metodi in economia e con la sperimentazione si è ritenuto utile realizzare un questionario, da presentare agli utenti alla conclusione dell'esperimento.

12.1 Scelta del questionario

I questionari hanno due fondamentali vantaggi rispetto alle interviste: in breve tempo riescono a raggiungere un grande numero di persone e richiedono poco impegno da parte degli esperti. L'unico vincolo è la necessità di formulare domande chiare e non ambigue.

La scelta dei questionari come approccio di valutazione indiretta, alternativo alle interviste, è stata dettata principalmente dal bisogno contingente di acquisire informazioni da un pubblico d'utenza abbastanza numeroso, costituito da sette studenti con poco tempo da dedicarci.

12.2 Tipi di domande usati

In generale, è possibile porre due tipi di domande: **domande aperte** e **domande chiuse**. Le domande aperte forniscono molte informazioni, ma la loro analisi è pesante; noi abbiamo deciso di proporre solo una domanda di questo tipo.

Per contro, le domande chiuse propongono una lista finita di risposte possibili, tra cui l'utente deve selezionare quella o quelle più conformi al proprio pensiero. Il fatto di suggerire in anticipo le risposte accettabili rende critica la loro scelta; inoltre, la risposta data sarà necessariamente più grossolana rispetto a quella fornita a una domanda equivalente, ma aperta.

Normalmente alle risposte chiuse è associata una sorta di scala di valori, indispensabile all'esperto per trarre conclusioni quantitative. La scelta dei criteri di valutazione delle risposte è un punto molto delicato e deve essere affrontato seriamente per non compromettere la validità dei risultati. Le domande chiuse possono essere classificate in base alla complessità della loro scala di valori:

- Pochi elementi identificano domande poco articolate, dette **checklist**. Le risposte sono del tipo **Sì**, **No** e **Non so**.
- Una scala più complessa è la **multi-point rating scale**, sempre discreta, ma ordinata tra un minimo e un massimo. Gli estremi possono corrispondere a giudizi del tipo **Inutile** e **Molto utile**. L'espressività delle risposte migliora notevolmente, a discapito però della semplicità di effettuare la scelta da parte di chi compila il questionario.
- Una scala **Likert** è come la precedente, ma associa ad ogni cella un giudizio testuale, rendendo più semplice e amichevole la scelta.
- Esistono anche scale più articolate, come la **semantic differential** o la **ranked order**, che però non descriviamo.

Per il nostro questionario sono stati utilizzati solo i primi due tipi di scala, perché semplici e adatti alle nostre esigenze. La natura strutturata del questionario pecca evidentemente di flessibilità, dunque è impossibile captare tutte le informazioni che un utente potrebbe, se intervistato, comunicare. Al fine di permettere all'utente di esprimere al meglio il proprio giudizio, è quindi stato deciso di introdurre anche una domanda finale aperta, che offre maggiore libertà di comunicazione.

12.3 Struttura del questionario

Il questionario propone alcuni gruppi di domande, incentrati su vari aspetti del sistema, ritenuti dagli autori interessanti e significativi. È organizzato in quattro sezioni: la prima raccoglie informazioni sul livello di confidenza informatica dell'utente, la seconda pone domande abbastanza generiche sul Vis5D, la terza approfondisce alcuni aspetti dell'interazione col software e l'ultima lascia spazio a considerazioni e suggerimenti personali. Non è stato fissato alcun limite di tempo per la compilazione del questionario.

12.4 Questionario di valutazione di Vis5D

Ti ringraziamo per la disponibilità offertaci e ricordiamo che la compilazione di questo questionario è totalmente anonima e ha carattere puramente indicativo. Precisiamo, inoltre, che non è indispensabile rispondere a tutte le domande e, se lo desideri, hai la facoltà di ritirarti in qualunque momento.

Il questionario è organizzato in quattro parti: la prima raccoglie alcune informazioni su di te, la seconda pone domande abbastanza generiche sul Vis5D, la terza ne approfondisce alcuni aspetti che noi riteniamo importanti e l'ultima lascia spazio alla tua fantasia. Occhio a non esagerare!!!

PARTE I: informazioni sull'utente

1. Quali tipi di software o ambienti hai già usato in passato? Seleziona una o più tra queste caselle:
 - Elaboratore di testi (ad esempio Word)
 - Database (come Access)
 - Foglio elettronico (tipo Excel)
 - Linguaggi di programmazione (come Pascal, C o Java)
 - Sistemi CAD (per esempio AutoCAD)
2. Come giudichi la tua conoscenza del sistema operativo Linux? Seleziona una sola tra le seguenti caselle:
 - Nulla
 - Superficiale
 - Sufficiente
 - Discreta
 - Buona
 - Ottima
3. In base alle tue esperienze con i diversi tipi di software elencati prima, valuta l'utilità dei seguenti metodi di input. Procedi crocettando una casella per ciascun metodo di input proposto:
 - Selezione tramite menu (ad esempio File, Modifica, Edit, ...)

--	--	--	--	--

Inutile Molto utile

- Selezione tramite tasti funzione (cioé **F1**, **F2**, ... **F12**)

--	--	--	--	--

Inutile Molto utile

- Selezione tramite icone e pulsanti

--	--	--	--	--

Inutile Molto utile

- Selezione tramite combinazioni di due o più tasti (ad esempio **Alt-F**, oppure **Shift-F1**)

--	--	--	--	--

Inutile Molto utile

PARTE II: domande generali sul Vis5D

1. Come hai individuato le azioni da eseguire per portare a termine il compito che ti è stato assegnato (visualizzazione campo di vento, manipolazione del display 3D e salvataggio di un'immagine)? Scegli una o più tra le seguenti caselle:

- Consultando il manuale di Vis5D
- Chiedendo aiuto agli esperti presenti durante l'esperimento
- Azionato casualmente i controlli, fino ad ottenere gli effetti voluti
- Ricordando gli esempi visti durante l'esercitazione
- In altro modo... (specificare meglio)

2. Hai faticato ad individuare i pulsanti che cercavi?

--	--	--	--	--

Per niente Moltissimo

3. Dopo aver azionato un pulsante, hai ottenuto l'effetto che prevedevi?

--	--	--	--	--

Per niente Perfettamente

4. Se hai commesso un'errore, quando te ne sei accorto?

--	--	--	--	--

Subito Molto più tardi

PARTE III: domande approfondite sul Vis5D

1. Hai incontrato problemi nel manipolare il display 3D?

--	--	--	--	--

Nessuno Moltissimi

2. Sei riuscito ad orientarti spazialmente nel display 3D (hai capito sempre a che latitudine e longitudine ti trovavi)?

--	--	--	--	--

Per niente Benissimo

3. Hai capito quali pulsanti erano attivabili e quali invece no?

--	--	--	--	--

Per niente Perfettamente

4. Come ti è sembrata la gestione delle finestre? Scegli una casella per ciascuno dei due gruppi proposti:

- | | |
|-------------------|----------------|
| ◇ Incomprensibile | ◇ Inefficiente |
| ◇ Macchinosa | ◇ Discreta |
| ◇ Intuitiva | ◇ Efficiente |

5. Hai incontrato nomi di pulsanti poco chiari o abbreviazioni ambigue? Puoi scegliere una sola casella tra queste:

- ◇ Non ricordo
- ◇ No
- ◇ Sì: ... (specifica quali se li ricordi)

PARTE IV: osservazioni e suggerimenti

Lasciamo questo spazio bianco per tue eventuali proposte di modifiche da apportare al Vis5D; commenti o suggerimenti saranno altrettanto graditi. Ti ricordiamo che alla fine ti aspetteremo alla Ca Nöa per un delizioso caffè; ogni consiglio accettabile dato qui verrà premiato con una brioche! Grazie.

12.5 Risultati del questionario

Teniamo a puntualizzare che, avendo prevalentemente utilizzato scale multivalore per le risposte alle domande, è molto difficile sia assegnare dei punteggi a ciascuna opzione disponibile, sia definire delle distanze numeriche tra le varie opzioni. Perciò, non essendo abbastanza abili in questo tipo di elaborazioni, si è deciso di non trarre dai dati conclusioni di tipo statistico, ma piuttosto solo indicazioni qualitative dedotte applicando il buon senso. Se si volessero ottenere risultati più completi e statisticamente validi, sarebbe necessario rivolgersi a psicologi e statistici professionisti.

12.5.1 PARTE I: informazioni sull'utente

Background informatico

Gli studenti partecipanti al test, come già anticipato, rivelano di possedere buone competenze informatiche: tutti hanno utilizzato i principali strumenti office disponibili sul mercato (Word, Excel, ...). A parte uno, gli altri in passato hanno programmato in C e/o Pascal o in altri linguaggi; solo due dichiarano di aver fatto esperienza con sistemi CAD.

Conoscenza di Linux

Per quanto concerne la conoscenza del sistema operativo Linux, il questionario svela una sostanziale esperienza da parte di quasi tutti gli studenti: solo uno non l'hai mai utilizzato, due dichiarano di conoscerlo superficialmente, due di possedere sufficiente esperienza e due di saperlo gestire con una buona preparazione. Si tratta insomma di un campione di utenza che nell'86% dei casi ha già interagito con un sistema Unix e, per quanto poco esperto, conosce almeno l'ambiente della shell.

Preferenza di stili di interazione

Nella domanda riguardante la preferenza tra gli stili utilizzati da una generica interfaccia, gli studenti hanno concordemente definito i menu e le combinazioni di tasti molto utili, mentre negli altri casi hanno fornito risposte non omogenee e quindi difficilmente catalogabili.

Per quanto riguarda i tasti funzione, in tre sostengono siano inutili ed in due molto utili (i restanti si sono astenuti dalla risposta); nei confronti delle icone in quattro ritengono siano quasi inutili mentre gli altri tre ne apprezzano moltissimo l'utilizzo.

12.5.2 PARTE II: domande generali sul Vis5D

Strategia di utilizzo di Vis5D

Tutti gli studenti dichiarano di aver cercato casualmente i controlli interessati, sperando di ottenere i risultati desiderati. Tre affermano di aver provato ad utilizzare il manuale, altri tre riconoscono di esser ricorsi ad aiuti richiesti agli sperimentatori che li osservavano.

Difficoltà nel riconoscimento dei pulsanti

In tre sostengono di aver faticato moltissimo per trovare i pulsanti cercati, mentre altri tre ritengono di aver compiuto uno sforzo elevato ma accettabile.

Effetti dei pulsanti

Gli effetti ottenuti una volta attivati i pulsanti opportuni sono stati corrispondenti alle aspettative degli utenti nell'opinione di cinque persone, mentre le altre due dichiarano il contrario.

Tempismo nel riconoscimento degli errori compiuti

Tre persone sostengono di essersi accorte subito degli errori compiuti nel corso delle proprie operazioni, mentre le altre quattro si dimostrano profondamente insoddisfatte del feedback fornito dal Vis5D per situazioni di errore.

12.5.3 PARTE III: domande approfondite sul Vis5D

Manipolazione del 3-D Display

La difficoltà riscontrata nella manipolazione del **3-D Display** per la visualizzazione delle immagini è risultata mediamente bassa; nessuno ha affermato di essersi trovato molto male.

Orientamento spaziale nel 3-D Display

Non tutti hanno risposto a tale domanda, forse perché poco chiara o, con più probabilità, perché il compito assegnato non richiedeva all'utente di orientarsi nel display. Due hanno risposto di non essersi per niente orientati, uno di esserci riuscito alla perfezione e due mantengono una posizione neutrale.

Comprensione dell'attivabilità dei pulsanti

Tre persone dichiarano di non aver mai assolutamente capito quali pulsanti fossero o meno attivabili, due ritengono che la situazione fosse sempre molto chiara e una si mantiene nel mezzo. Un candidato si astiene dalla risposta.

Gestione delle finestre

Tutti coloro che rispondono a tale domanda, forse un po' troppo specifica, si trovano d'accordo nel definire la gestione delle finestre macchinosa. Due persone la considerano inefficiente, mentre altre due la ritengono addirittura discreta.

Nomi e abbreviazioni di pulsanti

All'unanimità tutti gli studenti dichiarano di aver riscontrato difficoltà notevoli nel comprendere le funzionalità dei pulsanti per associazione al loro nome. Alcuni specificano inoltre alcune abbreviazioni che hanno notato essere incomprensibili o ambigue: tra esse citano **SFC**, **INTERP.**, **ANIM** e **ANIMATE**, **SAVE** e **SAVE AS...**

12.5.4 PARTE IV: osservazioni e suggerimenti

Nell'ultima sezione del questionario, come già accennato, è stato lasciato libero sfogo alle impressioni degli studenti, i quali hanno suggerito con cura alcune preziose migliorie che potrebbero esser apportate all'interfaccia di Vis5D al fine di renderla più usabile. Eccole:

- introdurre (estendere) l'uso della tastiera;
- rendere più chiaro il significato dei pulsanti del mouse nelle diverse modalità di utilizzo disponibili;
- ordinare i pulsanti, associando un senso alla loro posizione sullo schermo;
- riorganizzare i pulsanti di controllo in gruppi semanticamente omogenei;
- pensare all'introduzione di etichette, tipo post-it, che collegate ai pulsanti ne illustrino la funzione in caso di bisogno;
- arricchire e riorganizzare il manuale d'uso;
- ideare un pulsante che permetta una chiusura normale delle finestre dei sottomenu;
- in ogni contesto evidenziare solo i pulsanti che sono effettivamente attivabili, ingrigendo tutti gli altri;
- inserire una finestra che migliori la visibilità dello stato del sistema, ad esempio contenente latitudine e longitudine;
- rendere più visibili i pulsanti **Hwind1** e **Vwind1** nella finestra del **Control Panel**;

- introdurre un modo più comodo e chiaro per zoomare, traslare e ruotare le immagini nel **3-D Display**, alternativo a quello già presente, che richiede l'uso dei tre tasti del mouse;
- pensare ad un help on-line;
- permettere la modifica di alcuni parametri più direttamente, magari cliccando sulla loro visualizzazione nella finestra **3-D Display**;
- implementare un sistema di menu.

Da questa ricca lista, prodotta da soli sette studenti, possiamo dedurre che la loro disponibilità è stata molto elevata e particolarmente attenta agli scopi dell'esperimento. Inoltre, ci vantiamo di aver scelto una strategia di gratificazione del loro lavoro molto incentivante: per ogni consiglio riportato, abbiamo infatti promesso un compenso extra corrispondente ad una deliziosa brioche della pasticceria della Ca Nöa.

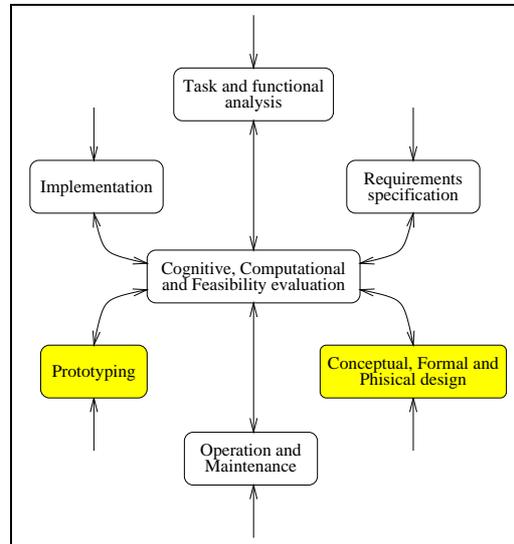
13 Conclusioni sulla valutazione

Accorpendo le singole conclusioni, prodotte da ciascuno dei diversi metodi di valutazione scelti, siamo pervenuti ad una lista di punti fondamentali, sui quali si dovrà necessariamente intervenire se si desidera migliorare l'usabilità di Vis5D. Riassumendo, è necessario:

- Introdurre nuove tecniche di apprendimento e migliorare quelle poche già presenti, in modo da ridurre il periodo di addestramento. Sfruttare la capacità di riconoscimento, ovunque possibile.
- Rendere il sistema più funzionale (efficiente), riorganizzando le funzionalità e rimuovendo quelle non utilizzate dalla categoria di utenti considerata (studenti). Visualizzare solo le informazioni strettamente necessarie, in modo più leggibile e utilizzando meglio lo spazio sullo schermo.
- Adeguare anche il programma di installazione, per ridurre le richieste di assistenza.
- Migliorare la visibilità dello stato del sistema, in modo che l'utente riesca a riconoscere velocemente dove si trova.
- Dialogare con l'utente in un linguaggio chiaro, senza ambiguità, usando solo abbreviazioni ovvie.
- Migliorare la consistenza, sia tra i vari aspetti interni al programma, sia tra il programma e le altre applicazioni.
- Introdurre un help in linea sensibile al contesto.
- Cercare di prevenire gli errori, dando la possibilità di annullarne gli effetti se accadono. Usare messaggi precisi e costruttivi.
- Permettere l'esecuzione delle stesse azioni tramite molteplici meccanismi di input, come tastiera e mouse. Introdurre degli acceleratori.

Rielaborando Vis5D secondo questi obiettivi, è stato ideato un prototipo che illustra come potrebbe essere fatto un visualizzatore più usabile.

14 Introduzione al prototipo



Durante le fasi precedenti sono stati rilevati e chiariti i bisogni degli utenti, raccolti i loro suggerimenti, svolte analisi in economia e quindi tratto delle conclusioni sui miglioramenti da apportare al sistema. Questi risultati hanno anche fornito alcune indicazioni pratiche su come procedere nella progettazione.

Prima di continuare è importante ricordare che si dovrà progettare un sistema *uomo-macchina*, non la sola macchina. Sarà perciò necessario considerare i singoli compiti che l'utente svolge, risultato della fase di definizione dei compiti, decidendo quali azioni, tra quelle che compongono ciascun compito, verranno svolte dall'utente e quali invece sono a carico della macchina (processo di *allocazione dei task*).

14.1 Osservazione sulla progettazione

Durante questa attività si chiarisce come i requisiti possano essere raggiunti attraverso un'opportuna architettura software. In questa fase si giocano le qualità tecniche dell'applicazione che si sta sviluppando; in particolare, la capacità dell'applicazione di evolvere in modo semplice ed affidabile a fronte di cambiamenti nei requisiti, che potrebbero sorgere ad esempio nella fase di uso e manutenzione.

L'attività di progettazione è scomponibile in due sottoattività principali, che possono risultare intercalate temporalmente in molti modi.

- Il progettista deve **capire le specifiche** che il prodotto dovrà soddisfare. La comprensione delle specifiche si realizza e si perfeziona discutendo i bisogni di coloro che useranno il prodotto e analizzando sistemi simili già esistenti. In particolare è utile analizzare sia sistemi esistenti di successo, per carpirne i segreti, sia fallimentari, per scovarne i problemi da non ripetere.

- Il progettista deve **sviluppare il prodotto**. Questo processo può essere visto come una successione di elaborazioni di idee, durante le quali esse prendono forma e si raffinano. Lo sviluppo è un'attività creativa, in cui il progettista impiega la propria esperienza.

Il progettista deve essere abile sia nel comprendere le specifiche che nel rappresentare le idee, in modi diversi durante le varie fasi della progettazione. Scegliere la rappresentazione adatta è importante per esplorare, provare, memorizzare e trasmettere le idee e le decisioni, sia agli altri membri del gruppo di progettazione che agli utenti.

Nel nostro caso il prodotto di riferimento, che dovremo migliorare, è Vis5D: terremo le parti utili, perfezioneremo quelle migliorabili ed elimineremo quelle dannose o inutili. Puntando verso gli obiettivi riassunti nel capitolo precedente, cercheremo di ottenere il prototipo di un visualizzatore più usabile. Il prototipo cristallizza le nostre idee, costituendone la rappresentazione.

14.2 Generalità sui prototipi

Sono sistemi autonomi, che però rappresentano solo alcune delle funzionalità che offriranno i sistemi finali. In particolare un prototipo:

- è funzionalmente incompleto;
- non copre tutti gli scopi che il sistema finale dovrà soddisfare;
- ha prestazioni diverse da quelle che avrà il sistema finale.

Dunque si tratta di un modello approssimato dell'applicazione, il cui obiettivo è fondamentalmente quello di essere mostrato all'utente, o usato da questi, al fine di ottenere una indicazione su quanto il prototipo colga i reali fabbisogni. Il prototipo, una volta validato, diviene una specifica del progetto e, nell'ottica iterativa ma non sequenziale del ciclo di vita a stella, costituisce di fatto un mezzo per realizzare la semantizzazione progressiva tra utenti e progettisti.

14.2.1 Classificazioni

I prototipi sono classificabili secondo diversi criteri ortogonali. Uno di essi li suddivide in due categorie: non attivi e attivi.

- Quelli **non attivi** sono disegni basati su carta o, equivalente informatico, costruiti con un programma di presentazione (per esempio PowerPoint) o un generico programma di disegno (ad esempio CorelDraw). Presentando delle istantanee dell'interfaccia, sono necessariamente statici. L'aspetto dinamico viene introdotto dal presentatore umano, che illustra le azioni che scatenano i singoli controlli, cambiando "fotografia". I prototipi statici costruiti con sistema informatico possono presentare qualche limitato comportamento dinamico.

- I prototipi **attivi** sono basati su calcolatore: in qualche modo lo si programma per mostrare parte del comportamento dinamico dell'interfaccia, oltre a quello statico.

Un'altro criterio di classificazione considera quali funzionalità sono state inserite nel prototipo, in relazione a tutte quelle che saranno disponibili nell'applicazione finale. In questo senso abbiamo tre possibilità:

- **Prototipo orizzontale:** sviluppa solo i livelli alti delle funzionalità di cui il sistema finale disporrà. E' un approccio in ampiezza, con profondità limitata.
- **Prototipo verticale:** approfondisce in dettaglio solo alcune funzionalità che offrirà il sistema. E' un approccio in profondità, ma solo per le funzionalità considerate.
- Per limitare il costo del prototipo, in alcuni casi è possibile limitarsi ad uno **scenario**. Lo scenario dispone solo di alcune funzionalità, approfondite inoltre con poco dettaglio. Va bene quando un utente individuale vuole ottenere una risposta precisa da un sistema che deve risolvere un problema ben circoscritto, in circostanze note e per un tempo limitato (per esempio il Bancomat).

Sviluppare lungo entrambe le direzioni, orizzontale e verticale, equivale di fatto a scrivere l'intero sistema.

14.2.2 Costo

Richiedendo tempo e un gruppo di progettisti, la fase di prototipazione risulta abbastanza costosa; occorre che il prototipo si sviluppi economicamente ed in tempi brevi. Vi sono almeno tre metodi per ridurre i costi:

- Implementazione poco efficiente, usando linguaggi di prototipizzazione.
- Implementazione non affidabile: possono esserci errori di codifica oppure algoritmi semplificati. Non tutti gli errori possono essere rilevati e di questi non tutti sono gestiti.
- Tecnica del mago di Oz, dove la parte del prototipo in realtà è realizzata da una persona nascosta che simula l'output del programma finale, trasmettendolo al calcolatore dove l'utente è seduto.

14.3 Il nostro prototipo

Per il nostro sistema è stato scelto un *prototipo verticale*, nel quale verranno considerate *tutte le funzionalità usate dagli studenti*, come ricavate dalla definizione dei compiti. Si tratta di un prototipo non attivo, la cui interfaccia è stata sviluppata con

Qt designer, un programma molto efficace e specifico per progettare le interfacce dei sistemi a finestre, distribuito con il toolkit grafico multiplatforma **Qt**.

Il prototipo dovrebbe incarnare molte delle migliorie descritte precedentemente. Per quanto riguarda la sua organizzazione, essenzialmente vengono mantenute le stesse funzionalità offerte da Vis5D, disposte però in modo più razionale e chiaro. E' stato introdotto un meccanismo di undo e resa automatica la procedura di creazione di animazioni.

15 Overview di Qt

Questa sezione contiene una rapida panoramica della libreria grafica che è stata scelta per un'ipotetica implementazione del prototipo. Varie ragioni ci hanno spinto a selezionare proprio questo prodotto:

- Poiché avremmo intenzione di rilasciare la nostra applicazione come **free software**, corredandola di una licenza GNU GPL, è bene utilizzare una libreria facilmente reperibile da tutti, a costo nullo e con la licenza d'uso meno incatenante possibile. In tale modo ognuno sarà effettivamente in grado di migliorare il nostro prodotto, apportandovi le modifiche che ritiene opportune e ricompilandolo. Qt è disponibile anche in una versione detta *free edition*, con una licenza d'uso molto simile alla GNU GPL.
- E' **multiplatforma**, e questo assicurerebbe al nostro programma il più ampio pubblico possibile, con minimo sforzo da parte nostra.
- E' **portabile**: non legandosi a sistemi particolari, non ci obbliga a considerare le specificità di ogni piattaforma. Infatti, sebbene le piattaforme possibili siano molte, il codice sarà unico; il porting si realizza completamente con una banale compilazione sui sistemi di destinazione.
- E' senz'altro **moderna**, ben supportata e con rischio di obsolescenza molto remoto.
- E' **usata ampiamente** da molte organizzazioni importanti.
- Dispone di un **designer**, un programma che permette di realizzare facilmente l'interfaccia dell'applicazione attraverso la disposizione drag and drop dei controlli necessari nelle finestre che successivamente diverranno i dialoghi dell'applicazione.

15.1 La situazione del mercato

Per molti anni gli sviluppatori di software si sono dovuti confrontare col problema di come competere in un mercato con molti sistemi operativi e vari sistemi a finestre. Questa situazione permane tuttora, e non c'è ragione per credere che cambierà. Infatti le insinuazioni, incominciate agli inizi anni 90, che davano per imminente l'estinzione dei sistemi compatibili Unix erano esagerate. In realtà, la forte diffusione di sistemi compatibili Unix a basso costo, quali **GNU/Linux** e **FreeBSD**, assicura che nel mercato del software continueranno ad esistere molte diverse piattaforme nel prossimo futuro.

Una sfida importante nel voler supportare diverse piattaforme consiste nel saper affrontare i costi derivanti dallo sviluppo e dalla manutenzione di una particolare applicazione per le varie piattaforme considerate.

Inoltre, a causa delle inconciliabili differenze tra i vari sistemi, il porting di un'applicazione verso un nuovo sistema può rendere necessari costi di ridisegno e reimplementazione non trascurabili.

15.2 Il ruolo di Qt

Qt è un **toolkit**, o *framework grafico*, da impiegare durante la creazione di software. Uno dei suoi obiettivi è aiutare a risolvere molte delle sfide più critiche che si incontrano quando si vuole sviluppare e mantenere un'applicazione multiplatforma. Le note che seguono illustrano le caratteristiche più interessanti di questa libreria.

15.2.1 La storia

Qt è prodotto e commercializzato dalla norvegese Trolltech (www.trolltech.com, oppure www.troll.no). È in vendita fin dal 1995; oggi è in uso presso le principali compagnie che realizzano software, come HP, IBM, Intel, Siemens, Ericsson e Xerox. È disponibile sotto due tipi di licenza: la prima è una classica licenza proprietaria a pagamento, simile a quelle che solitamente accompagnano le librerie commerciali. La seconda è una licenza GNU GPL; il codice è esattamente lo stesso.

Grazie alla disponibilità della versione free, Qt è ampiamente usato nei sistemi GNU/Linux: infatti è la base del popolare desktop environment chiamato KDE, virtualmente incluso in ogni distribuzione.

15.2.2 L'architettura

Qt è un toolkit grafico multiplatforma. È implementato come una libreria di classi C++ e mette a disposizione una ricca API (Application Programmer's Interface) per gli sviluppatori. Offre un'ampio spettro di utili funzionalità, ma si orienta principalmente verso la gestione della GUI (Graphical User Interface). Dal punto di vista dello sviluppatore, Qt prende il posto dei tradizionali toolkit usati per realizzare GUI, tra cui sono da ricordare le classiche librerie Motif e MFC (Microsoft Foundation Classes).

15.2.3 Sviluppo multiplatforma

La libreria di classi Qt è stata implementata per molti sistemi operativi e diversi sistemi a finestre, mentre l'interfaccia è identica per tutte le piattaforme. Questo significa che applicazioni scritte con Qt per girare su una certa piattaforma possono essere portate su un'altra tramite una semplice ricompilazione dei sorgenti, senza alcuna modifica. Dunque, attraverso Qt, i produttori di software possono sviluppare e mantenere un'applicazione multiplatforma attraverso lo sviluppo e la manutenzione di una singola base di codice. Qt supporta correntemente tre tipi di architettura:

- Unix, nelle sue varie reincarnazioni;
- Windows, nelle sue successive edizioni;
- sistemi embedded, per i quali Qt mette a disposizione un completo sistema a finestre, che può essere facilmente modificato per gestire qualsiasi tipo di display o meccanismo di input.

15.2.4 Look and feel

Il toolkit Qt è stato progettato in modo da conformarsi perfettamente con il look and feel del sistema grafico sul quale gira l'applicazione; è comunque importante sottolineare che Qt internamente funziona *emulando dinamicamente gli elementi GUI nativi*. Questo significa che ogni applicazione basata su Qt può mostrare qualsiasi stile di look and feel su qualsiasi piattaforma; la scelta può essere fatta anche mentre il programma è in esecuzione. Per chiarire meglio questa capacità proponiamo un esempio: è possibile fare in modo che un'applicazione che gira sotto Linux abbia l'aspetto di un normale programma Windows. Gli stili che Qt supporta sono Motif, CDE (Common Desktop Environment), Windows, SGI, MotifPlus e Platinum (Macintosh).

15.3 Il designer

Introdotta nelle release recenti di Qt, permette di costruire l'interfaccia dell'applicazione disegnandola; un programma esterno convertirà i disegni nel codice C++ equivalente. Il programmatore ha a disposizione i controlli standard di Qt e li può trascinare, disponendoli in alcune finestre; a runtime queste finestre diverranno i dialoghi usati dall'applicazione per interagire con l'utente. Ogni controllo può avere molte proprietà, ad esempio un flag che indica se è visibile oppure una stringa da visualizzare al suo interno, tutte impostabili tramite un dialogo specializzato del designer.

La figura 11 fotografa una tipica sessione d'uso del designer. Le barre degli strumenti mostrano sia alcune delle funzionalità del designer più usate, sia i controlli di Qt che possono essere trascinati nei dialoghi. Le due finestre a sinistra visualizzano parte dell'albero dei controlli già inseriti e gli attributi di uno specifico controllo, rispettivamente. Il dialogo più grande è l'area in cui il progettista disegna l'interfaccia; è l'equivalente del classico foglio di carta. I puntini sullo sfondo formano una griglia, sulla quale i controlli possono essere allineati facilmente. Da notare la grande quantità di controlli inseriti, disposti peraltro in modo piuttosto articolato. La finestra in basso a destra contiene il codice associato a un metodo di un certo controllo.

Il designer ha queste caratteristiche essenziali:

- gestisce il meccanismo dei segnali e degli slot, che è il particolare sistema utilizzato in Qt per scambiare messaggi tra i controlli;

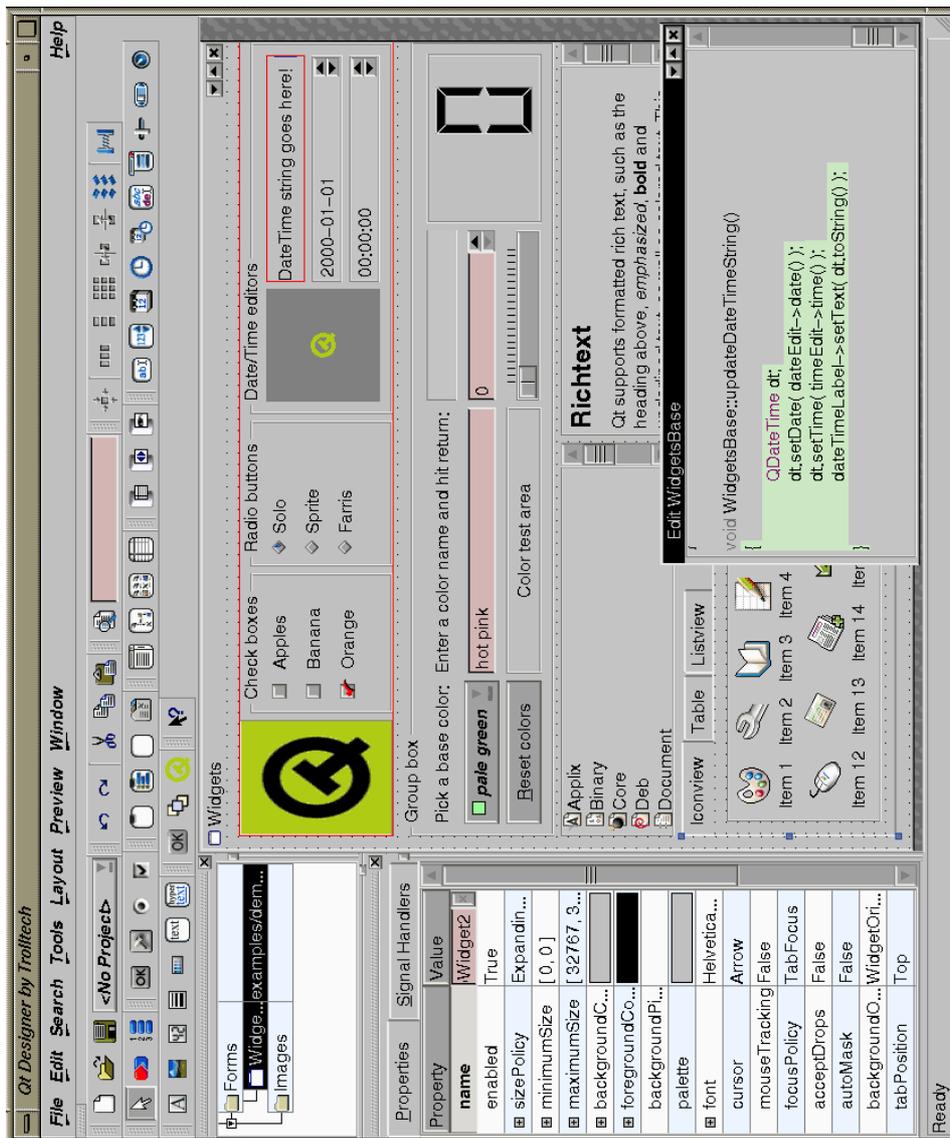


Figura 11: Una sessione d'uso del designer

- facilita l'internazionalizzazione delle applicazioni;
- permette di usare controlli custom;
- permette di includere controlli in altri controlli;
- dispone di undo multilivello, limitato solo dalla memoria del calcolatore;
- permette di modificare tutte le proprietà dei controlli;
- permette di definire acceleratori;
- utilizza formati XML per la memorizzazione dei dialoghi, il codice C++ equivalente viene prodotto da un programma di traduzione esterno (chiamato **uic**, user interface compiler), che riceve in ingresso la descrizione in XML;
- permette di estendere agevolmente la libreria di Qt, tramite il subclassing grafico;
- è parzialmente in grado di simulare il funzionamento dei singoli dialoghi.

Il designer può essere particolarmente utile nelle organizzazioni abbastanza grandi, dove più gruppi di persone lavorano sulla progettazione dei dialoghi e sulla implementazione di una stessa applicazione. Può darsi che gli esperti di HCI che ne fanno parte non conoscano abbastanza bene la programmazione da essere in grado di scrivere autonomamente almeno la parte di applicazione che si occupa di visualizzare e gestire i dialoghi.

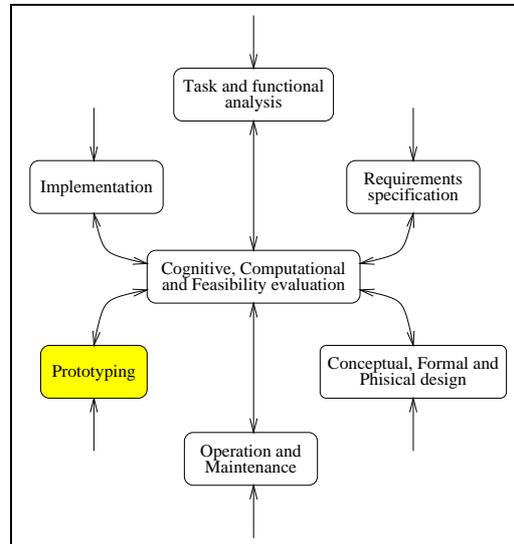
Senza un GUI designer, questi esperti probabilmente progetterebbero i dialoghi disegnandoli sulla carta o utilizzando un programma di grafica non specializzato. Di conseguenza, i programmatori sarebbero costretti ad interpretare² i loro disegni, cercando poi di tradurli in pratica mediante la scrittura del programma.

Questa interpretazione necessaria può introdurre una *distanza semantica* nel dialogo tra esperti di HCI e programmatori, che può fare in modo che ciò che gli esperti desiderano non è quello che i programmatori hanno capito che deve essere fatto. Può anche nascere un problema di realizzabilità pratica, perché non tutto ciò che è scrivibile in un foglio può essere costruito convenientemente.

Conseguenza ultima di questo processo è che l'applicazione può non rispettare il suo disegno. Per contro, utilizzando un GUI designer gli esperti di HCI possono facilmente progettare e realizzare l'interfaccia dell'applicazione.

²interpretare significa spiegare ciò che è oscuro

16 Il prototipo



In questa sezione verrà illustrato il prototipo ideato dagli autori. Si tratta di una descrizione essenzialmente testuale, con alcune immagini per chiarire gli aspetti più importanti. Come spiegato precedentemente, l'interfaccia è stata realizzata col Qt designer. Sono state considerate tutte le fasi che un'applicazione attraversa durante la sua vita, comprese l'installazione, gli aggiornamenti e la rimozione. Per quanto riguarda le funzionalità presenti, ricordiamo che si tratta di un prototipo verticale per la sola categoria degli studenti.

Per rimarcare ulteriormente l'intenzione di produrre un Vis5D migliore, ma che non tagli i legami con quello attuale, si è deciso di chiamare il nuovo sistema **NewVis5D**; inoltre viene utilizzato lo stesso formato binario per memorizzare i data set. Sempre per lo stesso motivo di uniformità con l'applicazione attuale viene mantenuta la lingua inglese e, per quanto possibile, si conservano i nomi dei comandi principali.

16.1 Installazione

A questo compito provvede un programma specializzato in installazioni di applicazioni (detto *installer*), simile a quelli usati sui sistemi Windows. Ad ogni categoria d'utenti gioverà senz'altro disporre di una facility del genere, poiché l'alternativa più comune, usata anche da Vis5D, è decomprimere il pacchetto con oscuri comandi Unix del tipo **tar xfvz NewVis5D.tar.gz**. Recentemente si sono anche diffusi dei *package manager*, programmi d'installazione generica in grado di montare qualsiasi applicazione già compilata. Il lato negativo di questo approccio è che tali programmi continuano ad essere essenzialmente a linea di comando e, disponendo di un ricco set di opzioni, risultano ostici da usare per gli utenti inesperti.

L'installer porrà all'utente una serie di chiare domande circa la directory dove installare il programma e sulle componenti ausiliarie da includere, tra quelle comprese nel pacchetto. Durante la decompressione e la predisposizione dei file, operazioni piuttosto lunghe, verrà visualizzato un dialogo contenente un indicatore di progresso e il tasto **Cancel**, per dare possibilità all'utente di bloccare l'installazione in qualsiasi momento, annullandone gli effetti. L'installer visualizzerà un'icona sul desktop di X Window ed inserirà una voce nell'area dei menu del window manager riservata alle applicazioni.

Aggiornamento

Gli installer delle versioni successive saranno in grado di identificare l'installazione di versioni precedenti e ne terranno conto, collocando il pacchetto nelle medesime directory e cancellando i file non più necessari.

Rimozione

Una parte dell'installer rimarrà memorizzata permanentemente sul sistema assieme agli altri file del pacchetto. Il suo compito è di rimuovere tutte le componenti dal sistema, se l'utente lo deciderà, ripristinando per quanto possibile la situazione precedente alla sua installazione.

16.2 Avvio del programma

Il programma può essere avviato in tre modi: cliccando sull'icona sul desktop, selezionando una voce nel menu di sistema, oppure lanciando direttamente l'eseguibile da una shell di X Window. A differenza di Vis5D, l'applicazione non richiede nessun parametro particolare sulla linea di comando e può partire senza la necessità di caricare alcun data set; in quest'ultimo caso verrà avviata con la sua finestra principale "vuota" e molti controlli ingrigiti, perché non invocabili data la mancanza del data set.

Durante il caricamento del visualizzatore viene mostrato uno *splash screen* al centro della sua finestra, recante il logo del programma, informazioni circa la versione in uso e un indicatore di progresso per dare feedback sul processo di caricamento. La figura 12 mostra il visualizzatore durante questa situazione. Al termine del caricamento lo splash screen scompare, cedendo il posto alla finestra di visualizzazione del data set, se il suo nome era stato passato come parametro sulla linea di comando, oppure allo sfondo grigio dell'applicazione, se il data set non era stato definito.

Il ruolo delle icone

Le funzionalità del programma sono attivabili essenzialmente in due modi: tramite apertura dei menu e selezione di una loro voce, oppure tramite click sulla corrispondente icona. E' tuttavia da precisare che non tutte le funzionalità elencate nei

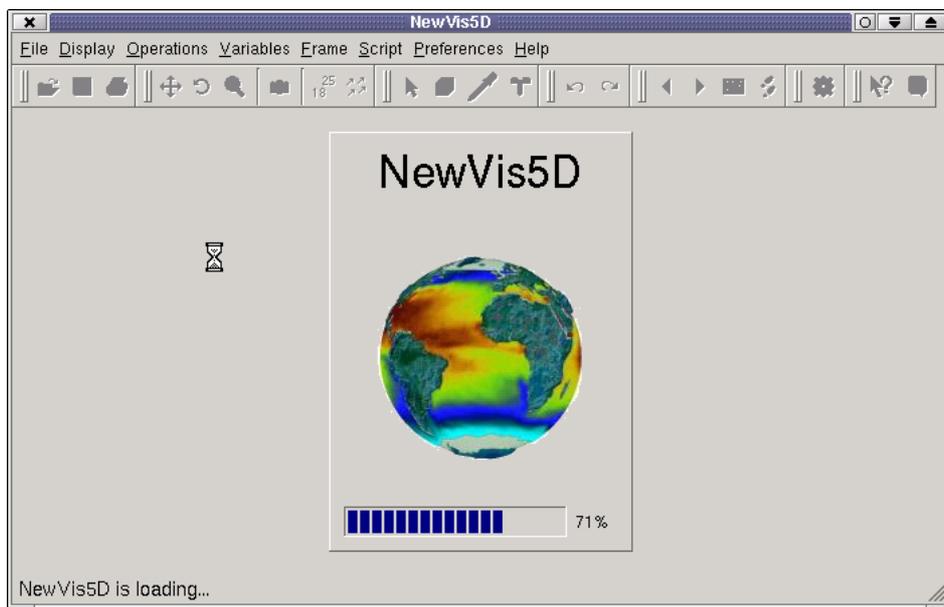


Figura 12: Avvio di NewVis5D

menu sono accessibili anche sotto forma di icone, ma solo le più usate. Questo scelta ha due spiegazioni: la prima è che abbiamo voluto evitare un sovrappollamento della barra delle icone, che potrebbe disorientare l'utente. La seconda è di carattere pratico, legata alla notevole occupazione spaziale delle icone, date le discrete dimensioni che abbiamo scelto per agevolarne l'individuazione.

Le icone sono raccolte su una barra, divisa in tante parti quanti sono i menu dell'applicazione. Ciascuna parte fa riferimento ad un solo menu; ogni icona si trova nella parte riferita al proprio menu. Queste parti sono staccabili: attraverso un'operazione drag and drop l'utente le può spostare in qualsiasi punto dello schermo, nelle posizioni che trova più comode.

Sempre a riguardo delle icone, sono da notare in particolare le ultime due: la seconda visualizza una finestra contenente l'help on line del sistema, incentrato sul controllo che detiene il *focus*. La prima visualizza anch'essa una finestra di help, ma riferita al controllo che si andrà a selezionare quando il puntatore assumerà un aspetto di freccia con accanto un punto di domanda, situazione che si ottiene subito dopo la pressione dell'icona. E' possibile chiedere aiuto su tutti i controlli.

Per terminare sulla descrizione delle icone, si puntualizza che alcune di esse recano le classiche immagini, standard di fatto, che si trovano in molti programmi per sistemi a finestre. Altre le abbiamo dovute disegnare noi; in particolare citiamo quelle per attivare le funzionalità di clipping, sounding, esecuzione dell'animazione, visualizzazione di campo vettoriale e di campo di vento. Le icone potrebbero essere accompagnate da una piccola scritta, per rendere più semplice il loro riconoscimento. Non siamo esperti di design perciò, per un'applicazione reale critica

dal punto di vista iconologico, ci saremmo rivolti ad uno specialista del settore.

Gli acceleratori

Per i comandi più frequenti è stata prevista una terza via da tastiera, che consiste nella pressione di tasti funzione oppure nella combinazione contemporanea di due tasti, di cui uno è una lettera o un tasto funzione e l'altro un modificatore (come **Shift**, **Ctrl** e **Alt**).

La status line

Da notare le informazioni presenti nella *status line*, la linea più in basso contenuta nella finestra dell'applicazione. In questo oggetto vengono visualizzate brevi scritte sensibili al contesto, che variano in funzione del controllo che possiede il focus. Nel caso il programma venga avviato senza data set, la status line informa della sua mancanza e ricorda all'utente come caricarlo.

I tool tip

A proposito di aiuto on line, annunciamo la presenza nel toolkit Qt di un'utile meccanismo in grado di visualizzare brevi messaggi di aiuto sensibili al contesto. Ogni volta che l'utente ferma per qualche secondo il puntatore del mouse su uno degli oggetti che compongono l'interfaccia, il sistema può visualizzare accanto all'oggetto una piccola finestra gialla contenente del testo che riassume brevemente la funzione svolta da quel specifico controllo.

Ad esempio, se l'utente si sofferma sull'icona di apertura file, il sistema può mostrare un tool tip del tipo **Opens a file**. Questa facility è attiva solo nelle modalità studenti e professionisti, ma può essere attivata o disattivata in qualsiasi momento agendo su una voce del menu **H**elp. I tool tip di Qt sono l'analogo virtuale dei post-it³ del mondo reale, che tutti dovrebbero conoscere.

16.2.1 Selezione categoria utente

Procedendo nell'esecuzione, l'applicazione presenta all'utente un dialogo *modale* che chiede di selezionare la categoria di cui fa parte (figura 13), che è stata ride-nominata *profilo utente* per maggiore chiarezza. In base alla scelta effettuata, il programma predispone le sue funzionalità in modo da rendere il suo uso da parte dell'utente il più agevole possibile.

Chi si classifica come studente, per esempio, sarà in grado di attivare i soli comandi relativi a tale categoria, come illustrati nella sezione di definizione dei compiti. I comandi invocati solo dai ricercatori o solo dai professionisti non saranno visualizzati; cambieranno anche l'aiuto ed il supporto offerto on-line. Dal menu **P**references dell'applicazione è comunque possibile cambiare il tipo di categoria

³nome registrato dalla 3M

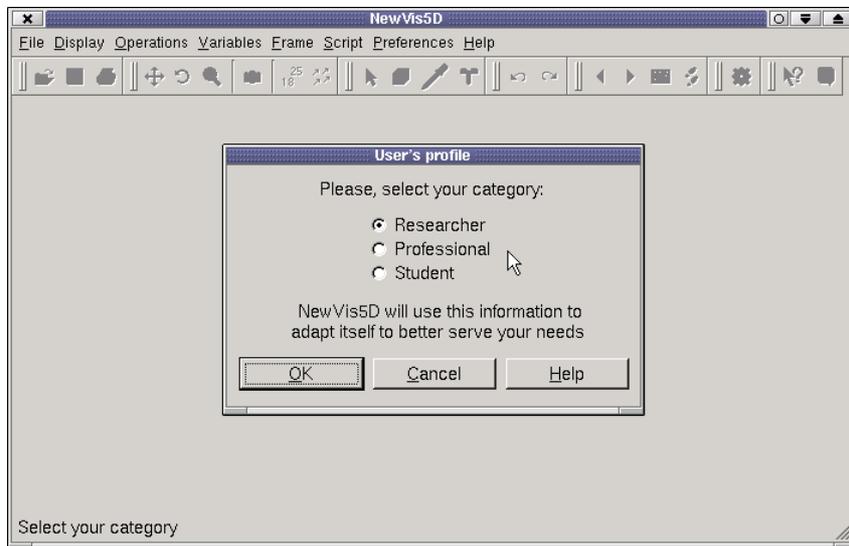


Figura 13: Selezione della categoria d'utente

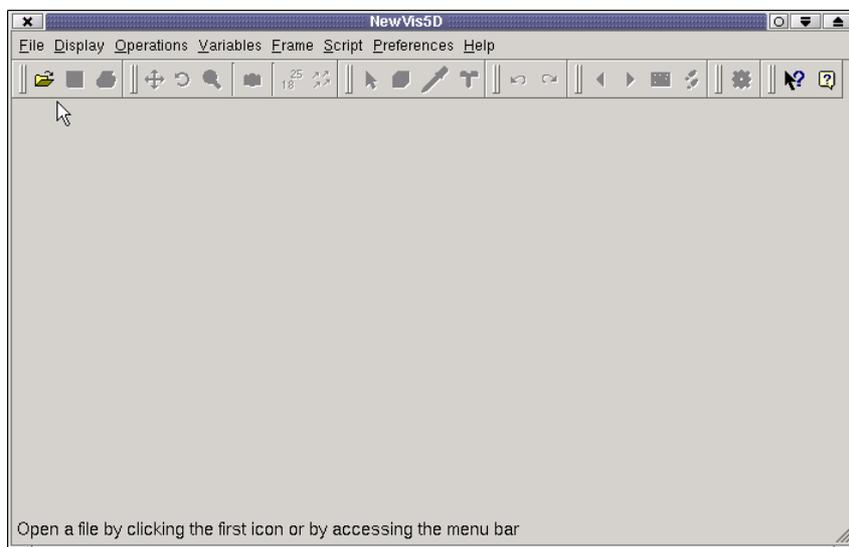


Figura 14: Aspetto di NewVis5D senza data set caricato

di utente, in qualsiasi momento. Per velocizzare la scelta, il dialogo propone la stessa selezione usata durante l'ultima sessione di utilizzo del sistema. La figura 14 mostra l'aspetto del programma una volta scelta l'interfaccia per gli studenti, senza alcun data set caricato perché nessuno era stato passato sulla linea di comando. Come indicano le icone, l'unica funzionalità invocabile in questa situazione è l'apertura di un data set, se si escludono quelle relative agli aiuti.

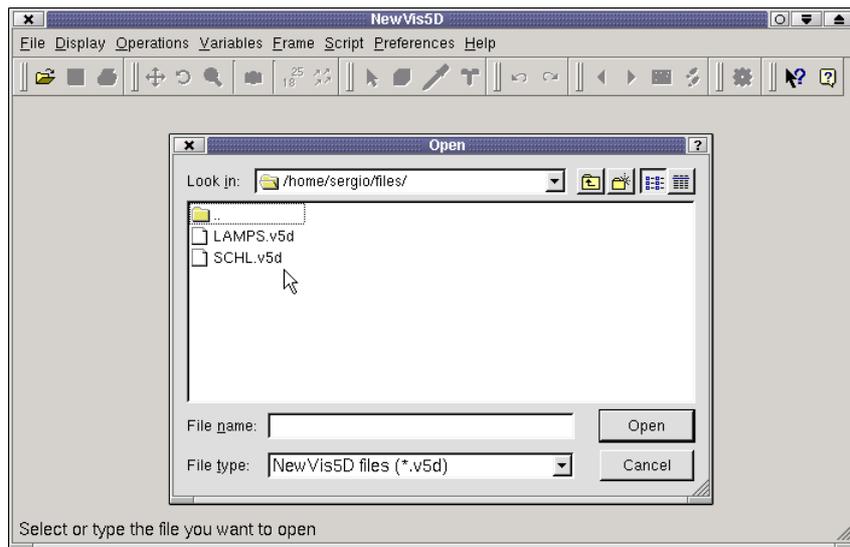


Figura 15: Apertura di un data set

16.2.2 Apertura di un data set

Una volta selezionata l'apertura di un file, indifferentemente tramite voce di menu, icona o acceleratore, verrà visualizzato il familiare dialogo modale della figura 15. Questo dialogo è specializzato nell'esplorazione del file system e rende possibile il reperimento del data set desiderato, in qualsiasi punto si trovi.

Dopo aver selezionato il file di interesse, la finestra principale dell'applicazione cambia come indicato nella figura 16; al suo interno compare il visualizzatore del data set. A questo punto tutte le funzionalità sono attivabili, come indicato dai vividi colori delle icone. In particolare, pan risulta la modalità di manipolazione del visualizzatore selezionata di default, come verrà descritto meglio più avanti.

16.3 La barra dei menu in dettaglio

Approfondiamo ora la particolare barra di menu che abbiamo disegnato, illustrata nella figura 16. Sono state conservate molte delle funzionalità offerte da Vis5D, alcune addirittura mantengono il nome originario.

Le differenze sostanziali sono due: la prima, macroscopica, è che le funzionalità vengono invocate tramite selezioni di voci di menu o icone piuttosto che

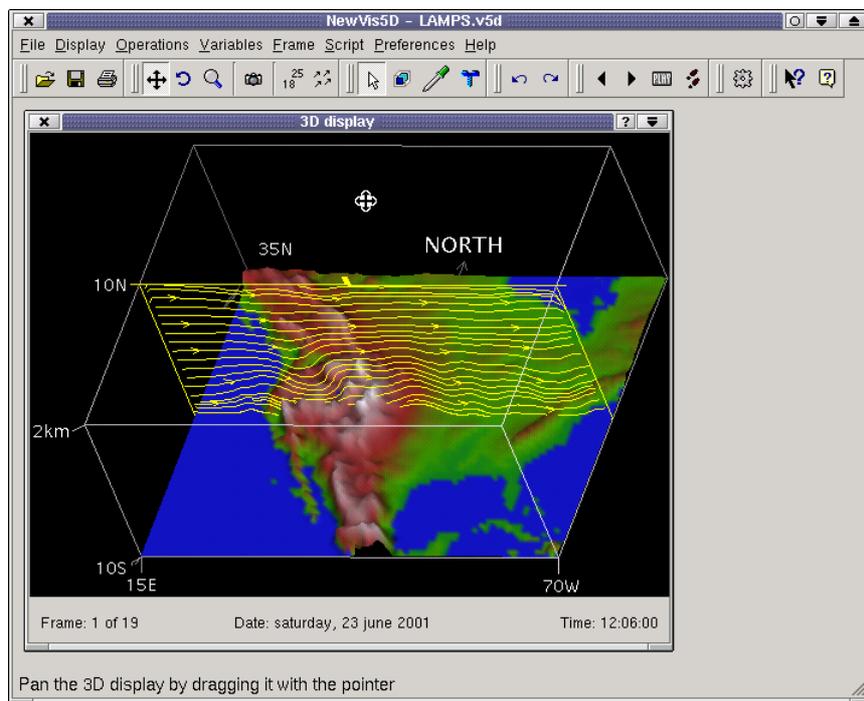


Figura 16: NewVis5D dopo aver caricato un data set

attraverso la pressione di pulsanti. La seconda, meno evidente ma più importante, è che i menu consentono una disposizione logica molto più efficace, in particolare dal punto di vista gerarchico. Questo dovrebbe ridurre la probabilità di disorientamento dell'utente e il tempo di ricerca della funzionalità desiderata.

Altro vantaggio della tecnologia dei menu a tendina è quello di non visualizzare tutte le informazioni che contengono, ma solo quelle previste nel particolare menu che è stato aperto. E' da sottolineare, comunque, che eventuali miglioramenti introdotti nei menu, tra una versione e la successiva del sistema, sono immediatamente visibili all'utente. Prevediamo la possibilità di variare la disposizione e il contenuto di ogni menu in funzione della categoria di utente selezionata; il menu mostrato si riferisce agli studenti.

Si rammenta che ogni volta che l'utente sposta il focus su una voce di menu, sulla status line apparirà un breve messaggio che illustra la funzionalità relativa a quella voce. Seguono descrizioni delle voci contenute in ciascun menu.

16.3.1 Menu File

Menu classico, presente in ogni applicazione visuale che elabora file e voglia mantenere un minimo di consistenza con le altre. E' illustrato nella figura 17; molto brevemente:

- Fornisce le funzionalità base per aprire e chiudere un data set, nonché per

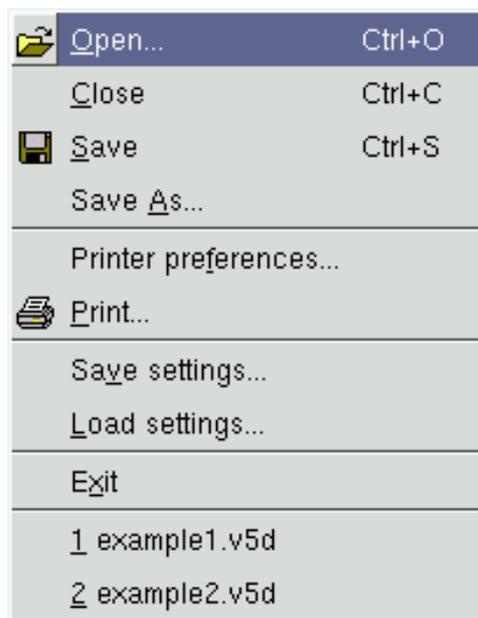


Figura 17: Contenuto del menu **F**ile

salvarlo con un altro nome. Le finestre mostrate da questi comandi sono i classici dialoghi modali che operano sui file, come quello di figura 15 che si riferisce all'apertura.

- Permette di configurare la stampante e di stampare.
- Le voci **S**ave settings... e **L**oad settings... permettono di memorizzare e richiamare le impostazioni del visualizzatore, in modo da rendere più efficiente l'interazione velocizzando il suo uso.
- Le ultime righe in basso propongono l'apertura di alcuni degli ultimi file visualizzati di recente, cioè permettono di richiamare le storie passate.

16.3.2 Menu **D**isplay

Contiene funzionalità particolarmente attinenti alla finestra visualizzatrice del data set, che verrà chiamata **3D display** per uniformità con Vis5D e il cui aspetto è illustrato nella figura 18. Le voci del menu sono mostrate in figura 19.

Ogni volta che si carica un data set viene aperto automaticamente il rispettivo **3D display**; dato che gli studenti non eseguono l'analisi comparata non hanno necessità di mantenere aperti più data set contemporaneamente, perciò verrà consentito loro di aprire al più un solo **3D display**. Ecco le funzionalità che questo menu contiene:

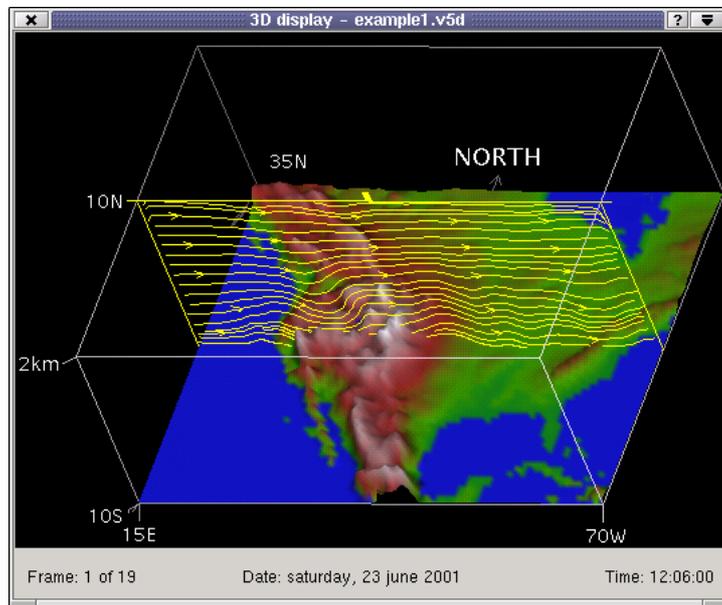


Figura 18: La finestra **3D display**

	<u>P</u> an	Ctrl+P
	<u>R</u> otate	Ctrl+R
	<u>Z</u> oom	Ctrl+M
	Save snapshot...	F10
	Show <u>s</u> calar fields	
	Show <u>w</u> ind fields	
	<u>C</u> lassic views...	
	Select <u>v</u> iew...	Ctrl+V

Figura 19: Contenuto del menu **Display**

- Le prime tre righe del menu (**P**an, **R**otate e **Z**oom) sono flag mutuamente esclusivi che selezionano i tipi di manipolazioni che si possono svolgere sul visualizzatore, tramite interazione diretta. Le stesse selezioni si possono compiere cliccando le icone corrispondenti sulla barra delle icone. Queste attività sono abilitate solo se si è in modalità **N**ormal, come spiegato più avanti nella descrizione del menu **O**perations; in caso contrario sono ingrigite.

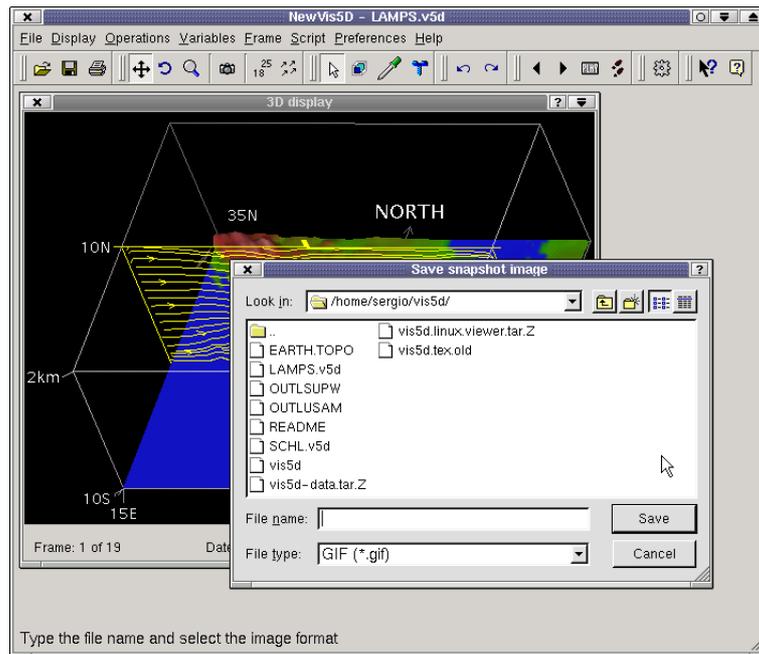


Figura 20: Salvataggio di un'immagine

- La voce **Save snapshot...** scatta una fotografia del contenuto corrente del **3D display**, aprendo successivamente un dialogo modale standard di salvataggio che chiede il nome del file ed il formato da usare per la memorizzazione, come mostrato in figura 20.
- Le voci **Show scalar fields** e **Show wind fields** sono flag, non mutuamente esclusivi, che aprono dialoghi non modali secondari nei quali l'utente può selezionare le variabili da visualizzare ed i grafici corrispondenti.
- Il comando **Classic views...** apre un piccolo menu *pop-up* aggiuntivo con le voci **South**, **Top** e **West**. Esse permettono di impostare il punto di vista sul **3D display**, tra i tre già disponibili in Vis5D.
- La voce **Select view...** apre un dialogo modale per impostare numericamente il punto di vista dell'utente nel **3D display**.

E' interessante notare che noi, mettendo questo menu per secondo, abbiamo rotto la convenzione che invece vorrebbe per secondo un menu del tipo **E**dit, le cui

classiche funzionalità potrebbero malamente essere rappresentate da quelle presenti nel menu **Operations**. I motivi sono due: il primo è che gli utenti passeranno la maggior parte del tempo manipolando il **3D display**, dunque una posizione comoda per il relativo menu può aumentare l'efficienza. Il secondo motivo è che le funzionalità del menu **Operations** sono poco assimilabili a quelle di un classico menu **Edit**, dunque non avrebbe senso considerarle equivalenti.

La finestra 3D display

Spostando il puntatore del mouse sul **3D display**, esso assume la forma del tipo di manipolazione attualmente selezionata, se non è in corso nessun tipo di operazione particolare, come sarebbe invece selezionando le voci **Clipping**, **Probing** o **Sounding** del menu **Operations**.

Dopo aver attivato le voci di questo menu l'interazione col **3D display** si realizza solamente tramite *dragging*, cioè premendo il tasto principale del mouse (il sinistro secondo le impostazioni di default per destri), trascinando il puntatore in una nuova posizione e rilasciando il pulsante. Il grafico si aggiorna continuamente mentre l'utente lo manipola. Come esempio, le figure 21 e 22 illustrano il funzionamento del comando **Rotate**.

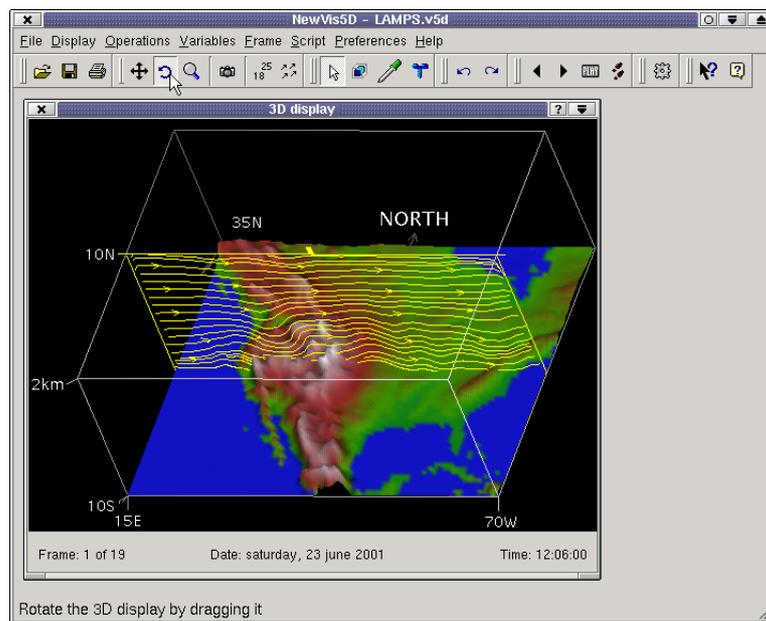


Figura 21: Selezione della manipolazione **Rotate**

Quando si utilizza lo zoom il puntatore assume la forma di una lente d'ingrandimento, con al centro un segno + o -, in funzione dello stato del tasto **Shift**. Questo permette di ingrandire o rimpicciolire il grafico.

Il tasto destro del mouse, se premuto sopra il **3D display**, visualizza un menu

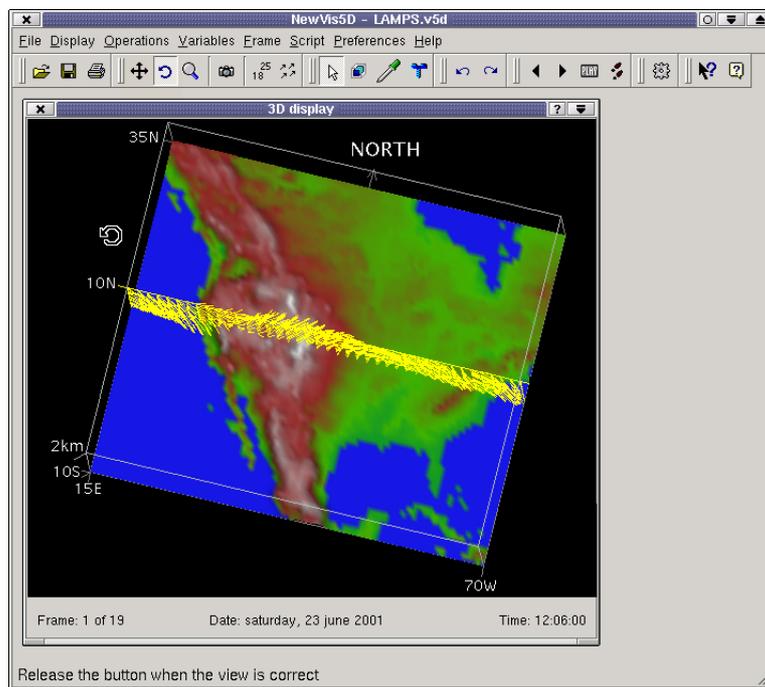


Figura 22: Termine della rotazione del data set

pop-up che contiene le funzionalità secondo noi più usate che hanno a che fare con tale finestra. Abbiamo previsto solo le voci **Pan**, **Rotate**, **Zoom**, **Classic views...**, **Save snapshot...** e **Animate**.

Come mostrato in figura 18, sul fondo del **3D display** possono comparire alcune scritte che riportano il numero del frame corrente all'interno della sequenza, la data e l'ora corrispondenti e le coordinate attuali del cursore. La loro visualizzazione è controllata da alcuni dei flag contenuti nel menu **Preferences**. Per quanto riguarda la data, si è scelto di visualizzarla in due modi diversi, in funzione della categoria d'utente attiva. Se l'utente è studente o professionista viene visualizzata nel formato civile, secondo le impostazioni locali del sistema su cui gira il programma. Se l'utente è un ricercatore la data viene mostrata in formato giuliano.

La voce **Show scalar fields...**

E' un flag che, se attivato, visualizza una finestra non modale contenente la lista di tutti i campi scalari disponibili nel data set, ciascuno dei quali viene accompagnato da un insieme di proprietà che determinano se è visibile e, se sì, sotto quale forma viene rappresentato (vedere figura 23). Questi campi corrispondono ai vari **U**, **V**, **W**, **T** e **P**, già descritti nell'overview di Vis5D. Il dialogo è facilmente ridimensionabile dall'utente tramite dragging del contorno della finestra.

La modifica ad una qualsiasi proprietà si esegue tramite doppio click sulla cella

che si trova all'intersezione tra il campo considerato e la proprietà desiderata. Apparirà un dialogo modale che chiederà l'introduzione del nuovo valore, proponendo quello vecchio come base per le modifiche; se il valore che deve essere introdotto ha un range discreto (come {No, Yes} per un flag), allora verrà data la possibilità di scegliere il valore tra una lista di tutti i valori possibili. Valori fuori range o impossibili non verranno accettati: il sistema mostrerà un messaggio d'errore e ricorderà all'utente il range accettabile per la proprietà che sta cercando di modificare.

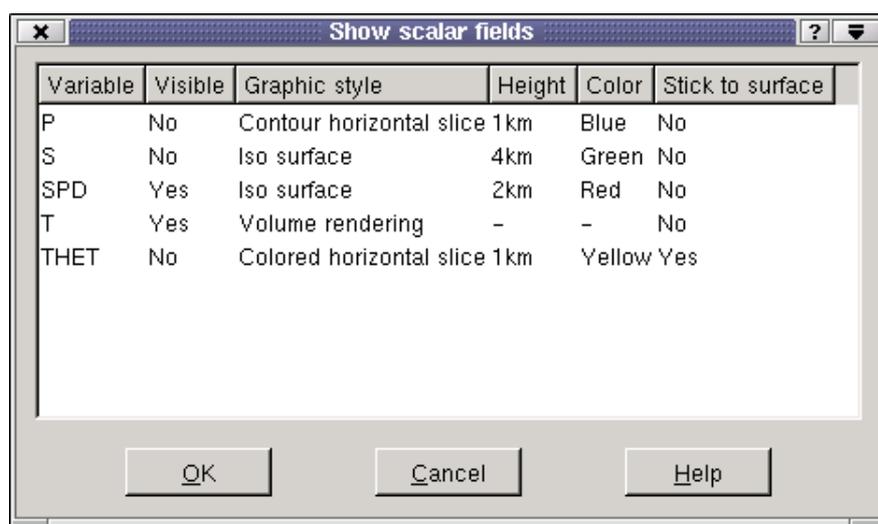


Figura 23: Selezione dei campi scalari da visualizzare

Se viene selezionato un grafico abbastanza oneroso da calcolare in termini di tempo, durante l'elaborazione verrà mostrato un piccolo dialogo modale contenente un indicatore di progresso ed il tasto **Cancel**, tramite il quale si può annullare il processo.

La voce **Show wind fields...**

Si tratta di un flag che, se abilitato, visualizza un dialogo non modale contenente una lista di campi di vento (figura 24). Ognuno di essi è corredato da una serie di proprietà che determinano se è visibile e in che modo viene visualizzato. Attivare uno di questi campi è l'equivalente di premere pulsanti del tipo **Hwind1** sotto Vis5D.

Per modificare una qualsiasi proprietà si procede come nel caso del dialogo precedente; gli errori verranno trattati allo stesso modo. Analogamente verrà visualizzato un dialogo con indicatore di progresso ogni volta che verrà selezionato un grafico che impiega più di alcuni secondi per essere generato.

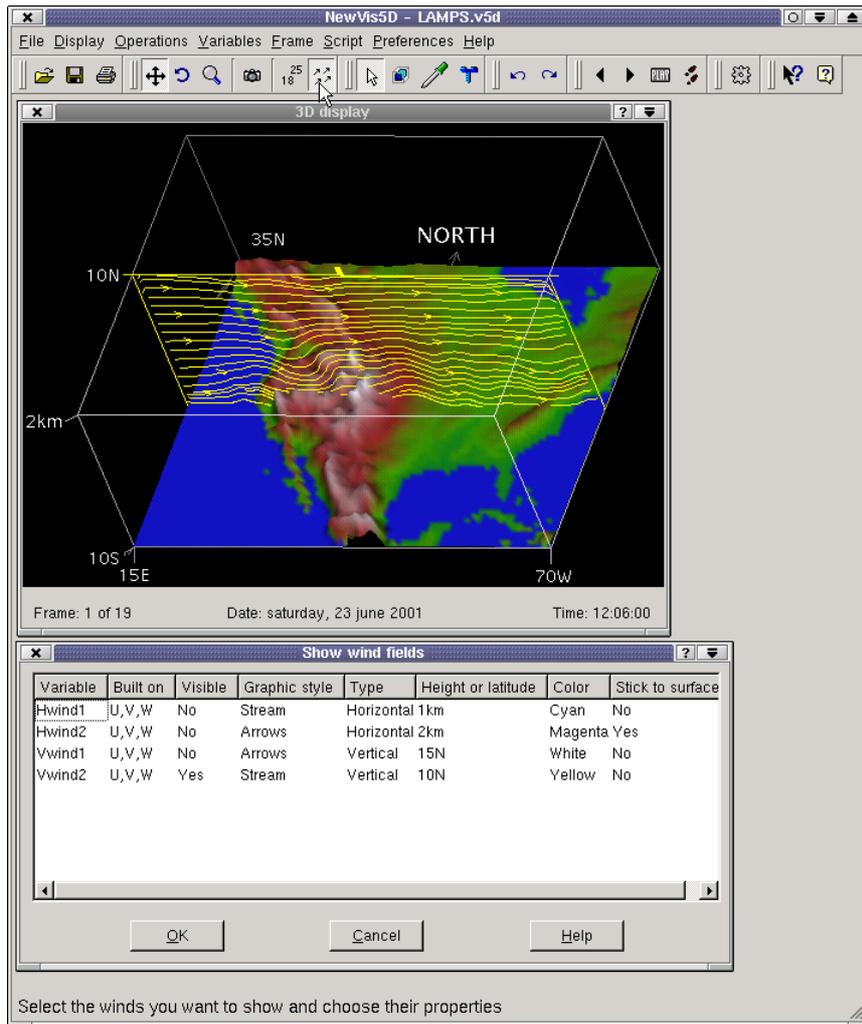


Figura 24: Selezione dei campi di vento da visualizzare

La voce **Select view...**

Aprire un dialogo modale specializzato (figura 25), tramite il quale è possibile impostare numericamente il punto di vista dell'utente e le coordinate del punto del dominio osservato; attraverso alcuni menu a torta è anche possibile apportare modifiche qualitative al punto di vista.

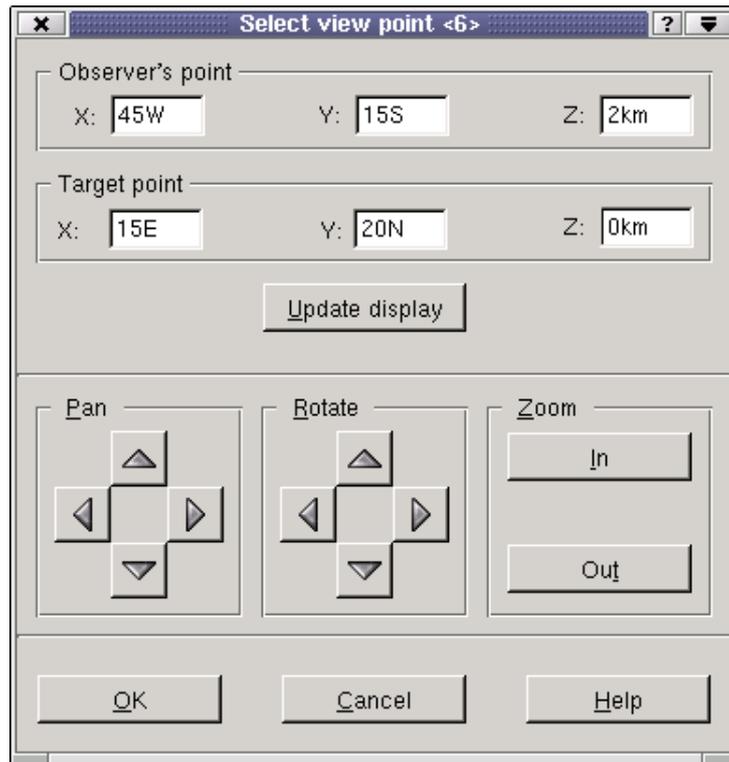


Figura 25: Dialogo di selezione del punto di vista

La parte superiore del dialogo permette l'introduzione delle coordinate in forma numerica; una volta completato il loro inserimento il display si aggiorna premendo il pulsante **Update display**. La parte centrale visualizza alcuni pulsanti che agiscono in modo un poco più qualitativo sulla visualizzazione. Sono stati divisi spazialmente in tre gruppi, a formare tre piccoli menu a torta, in modo che il loro significato sia il più immediato possibile. La peculiare disposizione dei pulsanti migliora l'usabilità, perché la loro posizione è analoga alla direzione lungo la quale agiscono. Una volta premuti, l'aggiornamento del display è immediato.

Per terminare questa descrizione, resta da chiarire il lavoro svolto dai pulsanti **OK** e **Cancel**: il primo rende permanente il punto di vista attualmente selezionato attraverso il dialogo; il secondo annulla i cambiamenti introdotti, riportando il punto di vista alla situazione precedente l'apertura della finestra.

16.3.3 Menu Operations

Contiene quattro opzioni mutualmente esclusive, che specificano il tipo di operazione corrente che si sta compiendo sul data set, come illustrato in figura 26.



Figura 26: Contenuto del menu Operations

La selezione Normal

Questa selezione, identificata dall'icona contenente il disegno del puntatore, è la condizione di default all'avvio. Pone il **3D display** nella modalità di manipolazione: tramite i comandi già visti **P**an, **R**otate e **Z**oom sarà possibile cambiare il punto di vista sul grafico. Nelle altre tre modalità questi comandi sono disabilitati e verranno visualizzati coi colori ingrigiti.

La selezione Clipping

Essenzialmente svolge l'operazione omonima che si trova in Vis5D. Permette di selezionare un sottovolume di dati dal dominio originario; i dati esterni a tale regione non verranno visualizzati.

Una volta posto il programma in questa modalità, nel **3D display** compariranno sei rettangoli, uno per ogni faccia del dominio, i cui vertici andranno posizionati dall'utente in modo da delimitare lo spazio di suo interesse. Prevediamo di marcare i vertici dei rettangoli con simboli evidenti, utilizzando inoltre colori molto vivi, in modo che attirino l'attenzione. Lo spostamento dei vertici si realizza molto intuitivamente tramite il loro drag and drop. Nel caso due o più vertici risultino sovrapposti, il programma dovrà visualizzare i relativi segni di marcatura leggermente distanziati tra loro, in modo che sia sempre possibile scegliere il vertice del rettangolo d'interesse.

L'icona che identifica questa voce sulla barra delle icone e nel menu è stata la più difficile da ideare; alla fine abbiamo scelto il simbolo del solido a sei facce disposto in modo poco regolare, contenuto all'interno di un parallelepipedo tratteggiato. Il solido contenuto è quello selezionato dal clipping, mentre il parallelepipedo tratteggiato è l'intero dominio; dunque l'icona è pensata per permetterne una semplice interpretazione anche dal punto di vista insiemistico. La scelta di un'icona contenente forbici, coltelli o altri arnesi da taglio sarebbe stata infelice

perché, intuitivamente, la funzionalità che descriverebbe sarebbe distruttiva, cosa che non si verifica azionando il clipping.

La selezione Probing

E' analoga all'omonima funzionalità disponibile sotto Vis5D. Una volta attivata, viene mostrato un dialogo non modale che visualizza i valori assunti da tutte le variabili nel punto del **3D display** in cui l'utente clicca (figure 27 e 28), le cui coordinate sono anch'esse visualizzate. Avere a disposizione un dialogo non modale consente di selezionare più punti in successione, senza bisogno di riaprire la finestra. E' altresì possibile inserire i punti che interessano mediante le loro coordinate esatte, scrivendole nelle tre caselle di input presenti nella parte alta del dialogo.

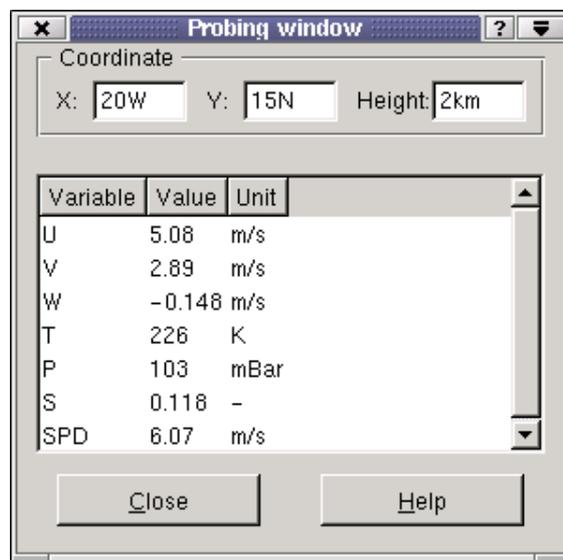


Figura 27: Dialogo di probing

Il dialogo è ridimensionabile, in modo che l'utente possa adattarlo alla lista di variabili da visualizzare e secondo lo spazio che vuole dedicargli sullo schermo. La finestra si chiude cliccando la sua casella di chiusura, premendo il suo tasto **C**lose, oppure selezionando un'altra modalità; nei primi due casi si torna nella modalità **N**ormal. L'icona che rappresenta questa operazione è la pipetta, attrezzo solitamente utilizzato da biologi e chimici per prelevare campioni di fluidi.

La selezione Sounding

Si tratta dell'ultima operazione che si può scegliere; è pressoché identica all'omonima funzionalità offerta da Vis5D. Attivandola viene visualizzato un dialogo non modale (figura 29) che riporta un carotaggio nel dominio, dall'alto verso il basso,

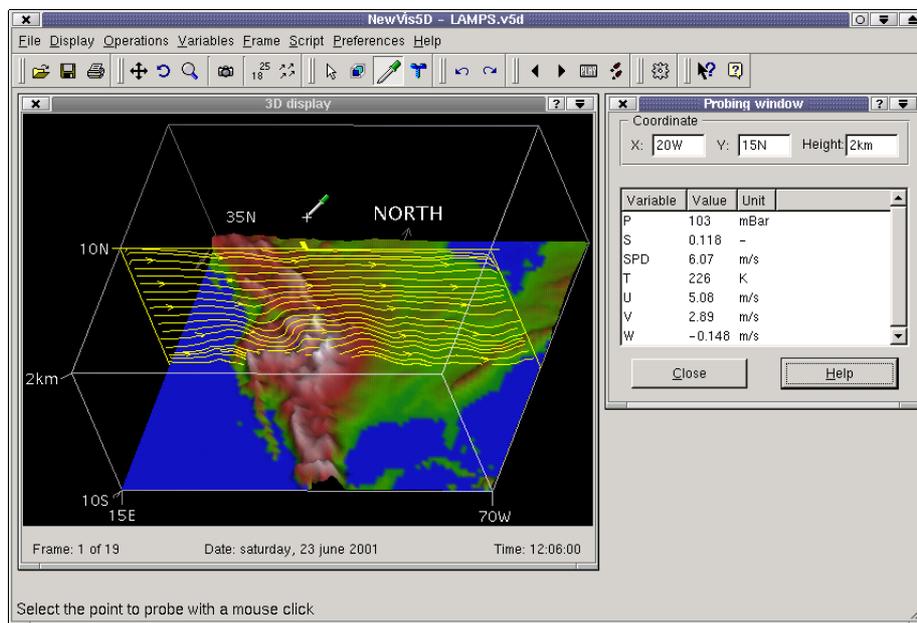


Figura 28: Il probing in azione

nel punto che l'utente seleziona con un click del mouse nel **3D display**. Poiché il dialogo non è modale l'utente può effettuare prelievi successivi in punti diversi, senza alcuna necessità di riaprirlo. Questa finestra è liberamente dimensionabile: il grafico e tutti i controlli che contiene si adatteranno automaticamente alle nuove dimensioni.

Anche se, secondo la definizione dei compiti stilata in precedenza, gli studenti non utilizzano questa funzione, si è preferito incorporarla comunque per dargli almeno modo di provarla. Trattandosi di una funzionalità "avanzata", è difficile implementarla in un modo che risulti contemporaneamente semplice, intuitivo ed efficiente. Noi ci siamo limitati ad emulare il dialogo che compare sotto Vis5D, strutturandolo meglio ed arricchendolo di informazioni ed aiuti. In particolare, la finestra è stata divisa in tre chiare zone: diagramma di visualizzazione del carotaggio nella parte superiore, area riservata ai settaggi al centro e zona dei pulsanti in basso.

Per quanto riguarda il diagramma, sono state aggiunte delle indicazioni testuali (etichette) sui suoi assi e sul grafico di ciascuna delle variabili visualizzate. Alla sinistra sono visibili le bandierine del vento: le loro direzioni e foggie identificano univocamente la direzione e la velocità del vento alle varie quote, nella verticale del punto scelto per il carotaggio. Sulla destra è visibile la scala, espressa in km, delle quote. Al centro è possibile visualizzare fino a tre variabili, selezionate dai controlli contenuti nella cornice **Vertical plot variables**. Ogni variabile è mostrata accompagnata dal suo nome; ad ognuna è comunque possibile assegnare un colore differente, premendo i pulsanti colorati. Sullo sfondo del diagramma si possono

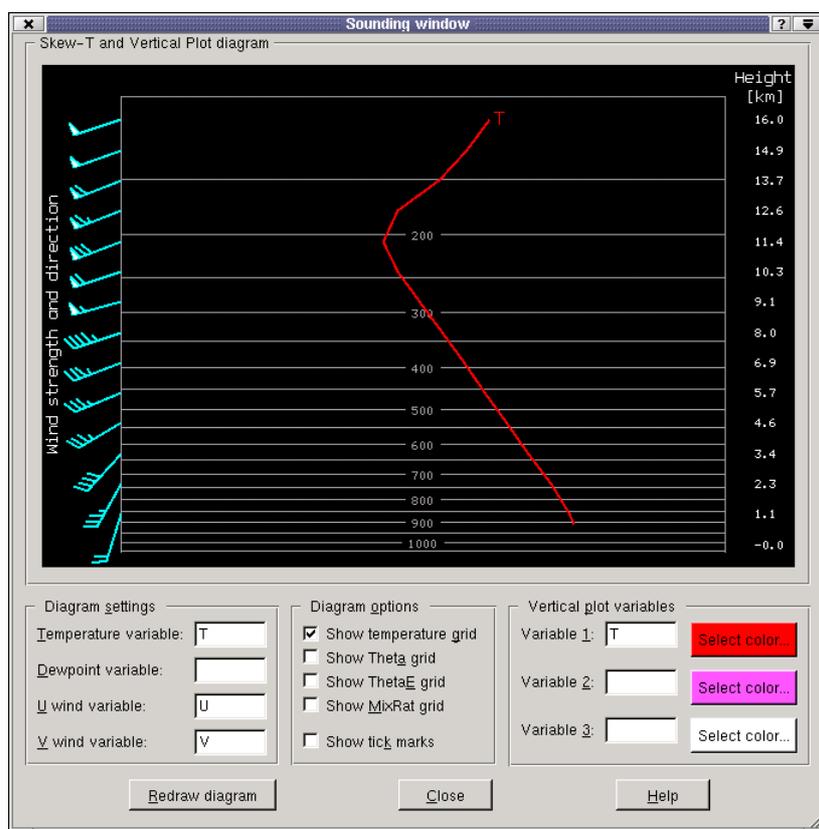


Figura 29: Dialogo di sounding

visualizzare vari tipi di griglie di riferimento, tutte attivabili tramite i check box contenuti nella cornice **Diagram options**. Nella prima cornice a sinistra, chiamata **Diagram settings**, vanno inserite alcune variabili meteorologiche necessarie a costruire il diagramma.

Il pulsante **Redraw diagram** si limita semplicemente ad aggiornare il diagramma secondo le nuove impostazioni; va premuto dopo averle alterate. La finestra si chiude cliccando la sua casella di chiusura, premendo il suo tasto **C**lose, oppure selezionando un'altra modalità; nei primi due casi si torna nella modalità **N**ormal.

Sulla barra delle icone questa funzionalità è rappresentata dall'immagine a forma di cavatappi, perché il suo compito è proprio quello di estrarre un campione dall'alto del data set ed analizzarlo. Un'altra figura interessante, forse migliore di quella scelta, consiste in un occhio che osserva dall'alto un generico data set, rappresentato da un parallelepipedo tratteggiato. Purtroppo non siamo riusciti a disegnarla, poiché le dimensioni delle icone sono di soli 22x22 pixel.

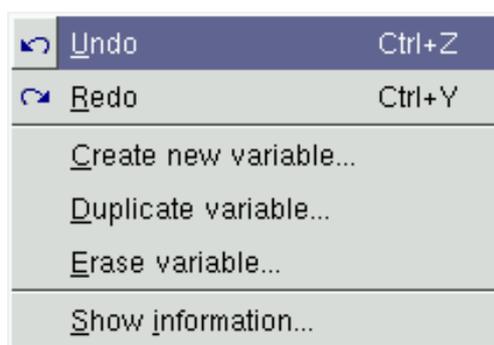


Figura 30: Contenuto del menu **V**ariables

16.3.4 Menu **V**ariables

Contiene tutte le funzionalità che hanno a che fare con la manipolazione delle variabili, esclusa la visualizzazione delle stesse che è completamente gestita dalle voci del menu **D**isplay. Il menu è illustrato in figura 30, segue una breve descrizione delle voci che lo compongono:

- Il comando **C**reate new var... apre un dialogo modale che permette di definire una nuova variabile, applicando una funzione matematica a quelle già esistenti. Il fatto di dover specificare una funzione matematica rende questa funzionalità poco immediata, dunque il suo uso potrebbe richiedere un certo investimento di tempo nella lettura dell'help in linea. Prevediamo di ricevere la funzione attraverso una casella di input, dove l'utente dovrà specificarla in modo testuale.
- La voce **D**uplicate variable... crea una nuova variabile, copiandovi i contenuti provenienti da una variabile già esistente. Avere due variabili con gli

stessi dati è utile se c'è necessità di visualizzare contemporaneamente il medesimo parametro fisico con due rappresentazioni distinte. In realtà questa funzione era ottenibile anche tramite la **C**reate new var..., ma per semplicità abbiamo preferito mettere a disposizione anche questa strada.

Un dialogo modale mostrerà una lista scrollabile contenente tutte le variabili già esistenti; l'utente dovrà selezionare la variabile da clonare e premere il pulsante **C**opy. La pressione di **C**ancel annullerà l'operazione.

- La funzionalità **E**rase variable... rimuove una variabile dal data set. Verrà visualizzato un dialogo modale con una lista scrollabile di variabili; l'utente selezionerà quella da distruggere e premerà **E**rase. Il pulsante **C**ancel annullerà l'operazione.
- La voce **S**how information... apre un dialogo modale che visualizzerà la lista di tutte le variabili presenti nel data set, accompagnandole dalle rispettive proprietà (ad esempio tipo di grafico associato, oppure colore). Inoltre mostrerà alcune statistiche sul data set, come quantità di memoria occupata, estensione spaziale e temporale.
- Il comando **U**ndo annulla l'ultima *modifica* apportata alle variabili del data set. Più precisamente, con *modifica* intendiamo la distruzione di una variabile oppure la creazione di un'altra; entrambe sono azioni che alterano il data set. La prima è un'azione distruttiva, che senza l'undo sarebbe irreversibile; la seconda azione non è ovviamente distruttiva, ma può essere comodo poterla annullare senza passare attraverso la funzionalità **E**rase variable...
Questo comando è multilivello, nel senso che può annullare anche le modifiche precedenti all'ultima; per realizzare questa situazione è sufficiente invocare la funzionalità tante volte quante sono le modifiche da invalidare. La sola limitazione sulla profondità possibile dell'intervento è data dalla quantità di memoria disponibile sul sistema: le modifiche più vecchie non saranno più invertibili perché le situazioni precedenti verranno inesorabilmente dimenticate.
- La voce **R**edo inverte l'azione dell'ultimo comando **U**ndo eseguito, ripristinando la situazione ottenuta come effetto dell'ultima modifica; anche questo comando è multilivello.

16.3.5 Menu **F**rame

La figura 31 illustra le voci contenute nel menu **F**rame; si tratta di tutti i comandi che hanno a che fare con la selezione di un fotogramma della simulazione.

- Le voci **B**ackward one e **F**orward one sono intuitive: fanno indietreggiare oppure avanzare di un fotogramma il filmato. Di conseguenza verranno



Figura 31: Contenuto del menu **F**rame

aggiornate anche le scritte sul fondo del **3D display**, se rese visibili nel menu **P**references. Se si cerca di indietro-reggiare prima del primo fotogramma o di avanzare oltre l'ultimo il nastro "si arrotola", passando rispettivamente all'ultimo o al primo fotogramma.

L'icona che avanza di un fotogramma potrebbe essere scambiata per il tasto **Play** di un registratore; tuttavia pensiamo che la presenza di un altro tasto con lo stesso simbolo speculare e l'esistenza di un terzo tasto con la dicitura **PLAY** esplicita dovrebbero far riflettere l'utente, facendolo convenire sul vero significato dell'icona.

- Il comando **S**elect... permette di selezionare direttamente un particolare fotogramma, tra quelli che compongono il data set. L'icona con il disegno delle impronte di scarpa dovrebbe trasmettere il significato di "go to".

Invocando questo comando verrà visualizzato il dialogo modale mostrato in figura 32. La selezione del fotogramma di interesse può dunque compiersi in due modi: direttamente tramite la casella di input, oppure tramite manipolazione della scroll bar. Nel secondo caso prevediamo di mostrare i fotogrammi che si susseguono nel **3D display**, mentre l'utente muove il cursore.

Premendo il pulsante **D**one si conferma il fotogramma selezionato, mentre un click su **C**ancel invalida il dialogo, facendo ritornare l'applicazione sul fotogramma visualizzato prima di invocarlo.

- Il comando **A**nimate attiva l'animazione, cioè il susseguirsi ordinato di tutti i fotogrammi di cui è composto il data set, mostrati ad intervalli costanti di tempo. Questo comando è mutuamente esclusivo rispetto a tutti gli altri; una volta attivato gli altri saranno inattivati. Agisce come un interruttore: per spegnerlo è necessario cliccarlo di nuovo.

In base alla potenza di calcolo disponibile ed ai tipi di grafici da visualizzare, l'applicazione sceglie tra ricalcolare ogni fotogramma prima di mostrarlo oppure creare un file temporaneo contenente l'animazione completa per

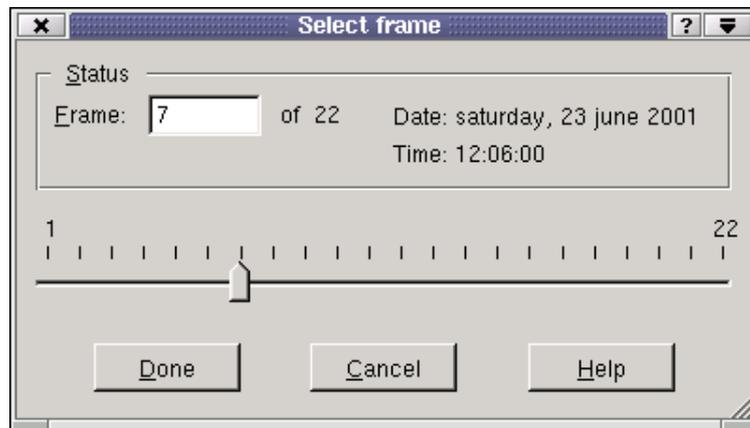


Figura 32: Il dialogo di selezione del fotogramma

poi riprodurlo più velocemente. Nel caso siano necessarie delle pause evidenti dovute al calcolo, verrà mostrato un dialogo modale con indicatore di progresso e tasto **Cancel**, per permettere l'annullamento dell'operazione.

- L'ultima voce **Save movie...** salva l'animazione in un file. Visualizza un classico dialogo modale per il salvataggio, dando all'utente la possibilità di scegliere il nome del file, il percorso nel file system ed il formato che deve essere utilizzato, selezionabile tra alcuni dei più diffusi. Attraverso due caselle di input è anche possibile specificare i numeri del primo e dell'ultimo fotogramma che devono comparire nel file; le caselle sono rispettivamente inizializzate a 1 ed al numero dell'ultimo fotogramma presente nel data set, dunque si salverà l'intera sequenza se non vengono modificate.

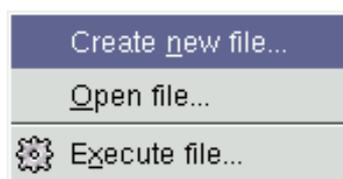


Figura 33: Contenuto del menu **Script**

16.3.6 Menu **Script**

Questo menu, mostrato nella figura 33, raccoglie le funzionalità indispensabili per supportare lo scripting.

La voce **Create new file...** visualizza una finestra non modale di editing, simile ad un normale editor di testo, tramite la quale l'utente può scrivere e salvare i propri programmi in linguaggio Tcl. Il comando **Open file...** apre una finestra di editing,

il cui contenuto è letto da un file specificato tramite un classico dialogo di apertura file. Per concludere, il comando **Execute file...** attiva l'interprete Tcl, dandogli in input un file che l'utente dovrà specificare. Un commento sull'icona usata per quest'ultimo comando: è stata scelta perché rende bene l'idea della computazione, inoltre è abbastanza utilizzata in editor specializzati per linguaggi di programmazione proprio per indicare funzionalità del genere. Osservazione: anche queste tre funzionalità base per lo scripting non sono richieste dalla categoria degli studenti; sono state inserite solo per desiderio di completezza.

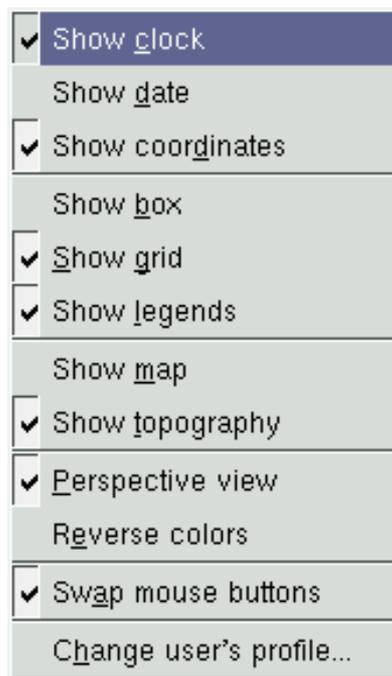


Figura 34: Contenuto del menu **P**references

16.3.7 Menu **P**references

Il contenuto del menu è illustrato nella figura 34. Le voci che contiene sono tutte, esclusa l'ultima, flag booleani che controllano alcune caratteristiche dell'applicazione. Segue una breve descrizione di ogni voce:

- **Show clock** controlla la visualizzazione dell'orologio nel **3D display**;
- **Show date** governa la visualizzazione della data nel **3D display**;
- **Show coordinates** condiziona la visualizzazione delle coordinate geografiche nel **3D display**;
- **Show box** comanda la visualizzazione del parallelepipedo tratteggiato intorno al data set nel **3D display**, già utilizzato dal Vis5D;

- **Show grid** governa la visualizzazione di una griglia di riferimento sulla base del dominio visualizzato nel **3D display**;
- **Show legends** amministra la visualizzazione delle scritte nel visualizzatore all'interno del **3D display**;
- **Show map** controlla la visualizzazione di una mappa orografica (catene montuose) all'interno del **3D display**;
- **Show topography** regola la visualizzazione di una mappa topografica nel **3D display**;
- **Perspective view** pone il **3D display** in modalità prospettica, se attivo;
- **Reverse colors** inverte i colori nel **3D display**, se attivo;
- **Swap mouse buttons** scambia i tasti destro e sinistro del mouse, se attivo;
- Il comando **Change user's profile...** visualizza il dialogo modale della figura 13, per permettere all'utente di cambiare la propria categoria di appartenenza.



Figura 35: Contenuto del menu **H**elp

16.3.8 Menu **H**elp

Come suggerisce il nome, le voci di questo menu (figura 35) hanno a che fare unicamente con il sistema di help in linea. Precisamente:

- Attraverso il comando **What's this?** è possibile chiedere informazioni su uno specifico controllo. Dopo averlo selezionato, il puntatore del mouse assumerà la forma di una freccia con accanto un punto di domanda; successivamente, dopo aver cliccato un qualsiasi controllo, verrà visualizzata una finestra di help relativa a quel controllo e il puntatore ritornerà nella sua forma standard.

- La voce **C**ontents... apre una finestra di help in linea che fornisce informazioni riferite al controllo o alla finestra che detiene il focus; è sempre possibile attivare l'help anche premendo il classico tasto funzione **F1**. La figura 36 mostra una possibile schermata d'aiuto offribile da questo meccanismo.

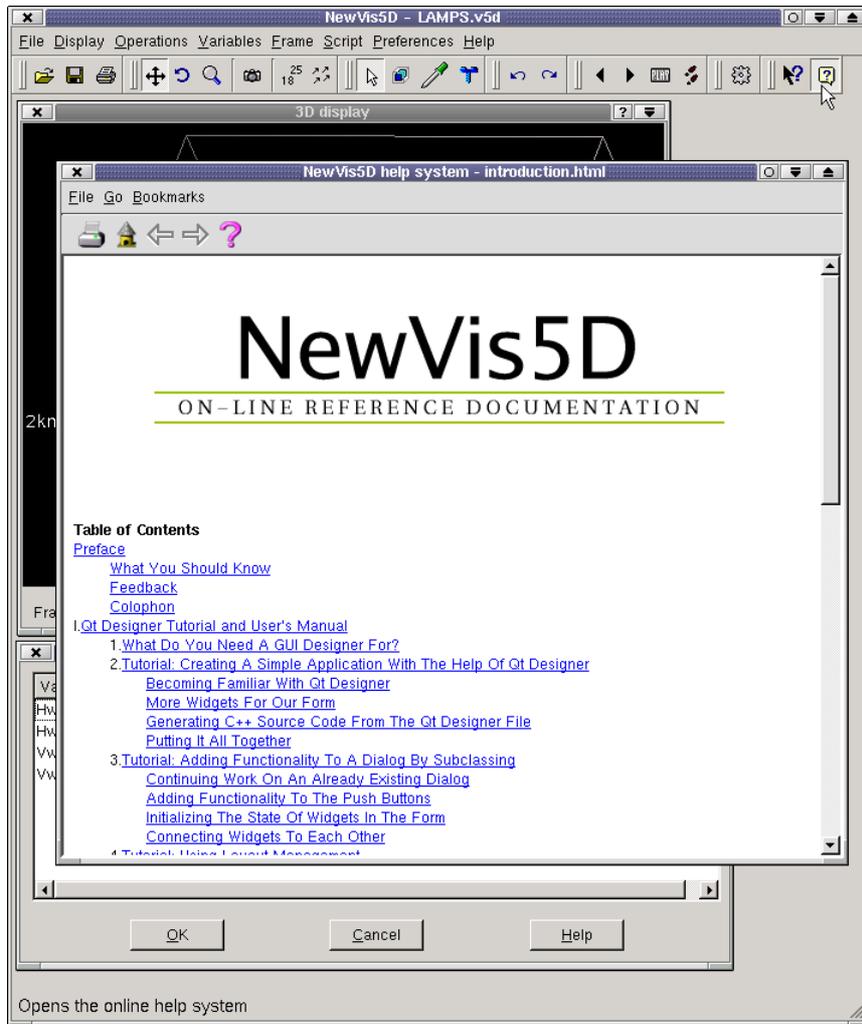


Figura 36: L'aiuto in linea offerto da NewVis5D

- Il comando **I**ndex... visualizza l'indice del sistema di help on line. A questo punto sarà possibile raggiungere ogni sezione del manuale, navigando fra link ipertestuali.
- Il flag **S**how tooltips controlla la visualizzazione dei tool tips, che ricordiamo essere brevi messaggi d'aiuto mostrati in piccole finestre gialle quando l'utente si sofferma con il puntatore del mouse sullo stesso controllo per al-

meno qualche secondo. La figura 37 illustra cosa accadrebbe se l'utente si fermasse sul pulsante di probing.

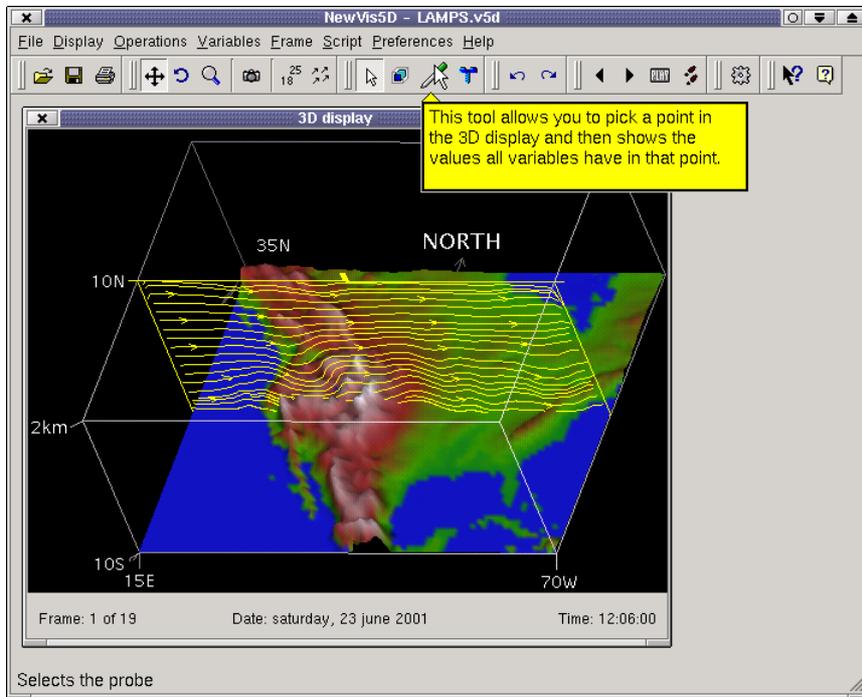
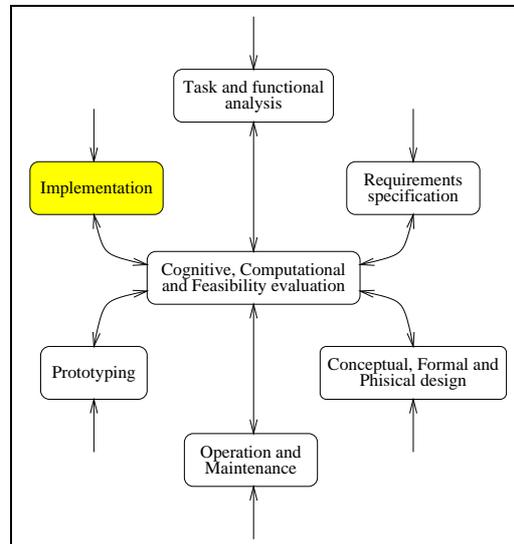


Figura 37: Il tooltip di aiuto per l'icona probing

- Il comando **About...** fornisce informazioni riguardo al prodotto; oltre al logo, visualizza anche la versione dell'applicazione ed i nomi dei progettisti e degli sviluppatori coinvolti nella sua realizzazione.

17 Considerazioni sulla realizzabilità informatica



In generale è necessario che il sistema sia contemporaneamente *efficace* nei confronti degli utenti, cioè ne massimizzi l'usabilità, ed *efficiente*, cioè abbia costi per l'implementazione, tempi di risposta ed uso di risorse sulle macchine comuni accettabili. Dunque esistono due poli opposti, dovuti al tradeoff difficilmente sanabile fra efficacia ed efficienza e tra i quali bisogna necessariamente trovare un punto di equilibrio se si vuole realizzare l'applicazione.

17.1 La nostra scelta

Poiché durante la progettazione è stato seguito un ciclo di vita a stella, incentrato sulle necessità degli utenti, si è naturalmente data precedenza all'usabilità dell'applicazione piuttosto che all'efficienza dei suoi algoritmi.

La preferenza accordata a questa strategia è incoraggiata dal fatto che un'applicazione simile a quella che si intende sviluppare esiste già: è il Vis5D; considerando che i sorgenti sono accessibili, è possibile conoscere perfettamente sia lo spazio occupato che il tempo speso dai suoi algoritmi. Queste informazioni sono particolarmente interessanti perché, si ricorda, l'intenzione è di migliorare Vis5D, perciò molti degli algoritmi di calcolo del nuovo programma possono venire ereditati da Vis5D senza sostanziali modifiche. Dunque è fattibile stimare quanto può divenire pesante la parte di interfaccia che sarà introdotta al posto di quella originale, prima di richiedere l'utilizzo di macchine non comuni.

17.2 Verifica della realizzabilità

Ricordando che i programmi interattivi si possono scomporre in due parti principali, l'*interfaccia* che dialoga con l'utente e l'*implementazione*, la parte di codice che

si occupa del calcolo vero e proprio, analizziamo questi due punti in questo caso. L'obiettivo è verificare se entrambi sono veramente realizzabili.

- **L'interfaccia.** Poiché è stato sfruttato un sistema informatico dedicato per progettartela, il **Qt designer**, che si appoggia ad un toolkit reale, cioè **Qt**, essa è sicuramente realizzabile.

Per quanto riguarda la pesantezza computazionale: considerando che l'interfaccia proposta è ordinaria, nel senso che opera in modo simile alle interfacce di molti altri programmi moderni in circolazione, possiamo affermare che essa non è eccessiva e dovrebbe essere paragonabile a quella dell'interfaccia già integrata in Vis5D.

- **L'implementazione.** Come detto precedentemente, le attività di calcolo del nuovo sistema saranno simili a quelle fornite dal Vis5D, perciò non si attendono da questa direzione sorprese particolari, dovute a scarsa efficienza computazionale o legate a difficoltà realizzative.

Le uniche vere novità rispetto a Vis5D sono due: possibilità di riaprire velocemente i data set visitati nelle sessioni d'uso precedenti del programma (una sorta di richiamo delle storie passate) e capacità di effettuare undo multilivello.

La prima innovazione è molto semplice da gestire: comporta la semplice memorizzazione di alcuni nomi di file. La seconda miglioria è anch'essa realizzabile, ma più faticosa da implementare nonché computazionalmente pesante: è necessario tenere traccia di tutte le modifiche subite dalle variabili che compongono il data set, nei limiti posti dalla memoria del sistema. Questo richiede l'organizzazione e la gestione di strutture dati non banali.

Dunque il prototipo è realizzabile nella realtà e le risorse computazionali necessarie dovrebbero essere simili a quelle richieste attualmente dal Vis5D, sia in termini di memoria che di potenza di calcolo.

17.3 Fusione tra interfaccia e codice

Supponendo di aver terminato la progettazione dell'interfaccia col Qt designer e di voler prendere in prestito, per quanto possibile, gli algoritmi di calcolo di Vis5D, sorge un problema pratico, legato al modo di "collegare" l'interfaccia con l'implementazione. Il Qt designer memorizza i dialoghi creati in un formato XML; da questo è possibile generare dei sorgenti C++ equivalenti tramite l'uso di un traduttore distribuito con il pacchetto chiamato **uic**, user interface compiler.

A questo punto, per realizzare la fusione fisica tra i due gruppi di sorgenti, sarà necessaria una discreta attività di editing da parte di programmatori esperti. Considerata la stretta parentela tra C++ e C, linguaggio in cui Vis5D è scritto, le sue routine si dovrebbero tuttavia "riciclare" facilmente nella nuova applicazione.

17.4 Sulla manutenzione

Poiché il toolkit Qt è realizzato in C++ l'applicazione andrà programmata nel medesimo linguaggio, se si intende mantenere accettabile la durata del processo di implementazione e quindi il suo costo. Come noto, questo linguaggio è orientato agli oggetti e favorisce la riusabilità; si presta dunque meglio del classico C, col quale Vis5D è implementato, a modifiche ed estensioni. Inoltre esso è flessibile, efficiente e portabile, tutte caratteristiche tipiche del linguaggio C da cui deriva.