

La Guida di Inform 7

di Jim Aikin

versione 1.00 (giugno 2, 2009)

*The Inform 7 Handbook
by Jim Aikin*

*Traduzione italiana
di [Leonardo Boselli](#)*

The Inform 7 Handbook Copyright © 2009 Jim Aikin

The Inform 7 Handbook è stato rilasciato ed è soggetto ai termini della *Creative Commons Public License*, qui riportati come riferimento. Vedi l'Appendice B di questo libro e leggi la licenza prima di utilizzare questo materiale. Il suo utilizzo significa che si è in accordo con i termini della licenza.

E' possibile copiare, distribuire e trasmettere quest'opera, è inoltre possibile estrarne delle parti o adattarla, in toto o in parte, secondo le seguenti condizioni:

- 1) L'opera deve essere attribuita mostrando in modo chiaro le parole “*Inform 7 Handbook*, by Jim Aikin”.
- 2) Non si deve indurre a credere che Jim Aikin approvi l'uso che viene fatto dell'*Inform 7 Handbook*.
- 3) Non si deve ricevere denaro o alcun altro compenso per aver copiato, distribuito o trasmesso quest'opera, né per le opere derivate che la incorporino, in toto o in parte.
- 4) Se quest'opera viene alterata, trasformata, o utilizzata per crearne un'altra, ciò che viene ottenuto deve essere distribuito solo alle stesse condizioni o sotto una licenza simile a questa.
- 5) Per ogni riutilizzo o distribuzione, devono essere resi noti a chi riceve l'opera i termini della licenza incorporando il testo della licenza (vedi Appendice B) o includendo un collegamento alla pagina Web in cui la licenza è pubblicata.

Indice

Prefazione alla traduzione.....	7
Come installare le estensioni italiane.....	7
Introduzione.....	8
Ringraziamenti.....	10
L'autore.....	10
Capitolo 1: Come iniziare.....	11
Download e installazione.....	12
Il programma Inform 7.....	13
Le intestazioni di pagina.....	14
Il pulsante Go!.....	16
Il pulsante Release.....	18
Altre funzionalità.....	18
Come utilizzare la documentazione.....	18
Cosa avviene in un gioco.....	19
Inserire comandi.....	19
Giochi da scaricare.....	23
Ad ogni modo, cosa sono questi .z5 e .z8?.....	23
Scrivere il primo gioco.....	24
Dalla cima.....	25
Raccontare una storia.....	27
Gestire i progetti.....	27
Tutto sui Bug.....	28
Utilizzare i comandi di Debug.....	29
Tutto sui Puzzle.....	32
Estensioni per Inform.....	35
Dove trovare ulteriori informazioni.....	36
Capitolo 2: Stanze e scenari.....	38
Creare la prima stanza.....	39
Scenari.....	41
Scenari distanti.....	45
Aggiungere stanze alla mappa.....	47
Scenari in cui si può entrare.....	51
Porte.....	52
Porte chiuse a chiave.....	54
Porte segrete.....	56
Porte pericolose.....	57

Finestre.....	59
“Non puoi andare in quella direzione.”.....	60
Doppie uscite e connessioni serpeggianti.....	62
Cambiare la mappa durante il gioco.....	64
Regioni.....	65
Sfondi.....	67
Rimuovere uno sfondo.....	67
Stanze buie.....	69
Stanze apparentemente identiche.....	70
Capitolo 3: Oggetti.....	71
Creare oggetti.....	71
Oggetti contro generi.....	72
Il nome degli oggetti.....	74
Nomi lunghi.....	76
Aspetto iniziale.....	77
Aggiungere parole al vocabolario.....	78
Contenitori & Supporti.....	79
Guardare sotto e guardare dietro.....	87
Prendere tutto.....	89
Contenitori e supporti accessibili.....	89
Spostare gli oggetti.....	91
L'inventario.....	92
Verificare dove sono gli oggetti.....	93
Plurali e oggetti collettivi.....	94
Oggetti composti di parti.....	96
Leggere argomenti.....	97
Scrivere sugli oggetti.....	98
Meraviglie meccaniche.....	100
Annusare e ascoltare.....	103
Trasformare un oggetto in un altro.....	104
Capitolo 4: Azioni.....	106
Azioni predefinite.....	106
Utilizzare “Instead”.....	109
Trattamento delle azioni.....	109
Liste di regole e interruzioni.....	113
Tre tipi di azioni.....	116
Creare nuove azioni.....	117
[Any Thing].....	121
Azioni con argomenti.....	123
Trattamento delle azioni — Sommario.....	125

Capitolo 5: Creare personaggi.....	128
Panoramica.....	129
Il personaggio del giocatore.....	130
Creare un PNG.....	132
Mr. e Mrs.....	134
Conversazioni, Parte I: Parlare a.....	136
Conversazioni, Parte II: chiedere/dire/dare/mostrare.....	138
Conversazioni, Parte III: Le conoscenze del personaggio.....	142
Conversazioni, Parte IV: Conversazioni basate su menù.....	143
Impartire ordini ai personaggi.....	145
Far muovere i personaggi.....	147
Personaggi che seguono il giocatore.....	151
Personaggi che il giocatore può seguire.....	151
Azioni sceniche.....	153
Azioni del personaggio.....	153
Combattimento.....	155
Stati d'animo.....	156
Parti del corpo.....	157
Capitolo 6: Vincere e perdere.....	158
Un cenno sulla lealtà.....	159
Tenere il punteggio.....	160
Lo scrigno del tesoro.....	162
Capitolo 7: Il tempo e le scene.....	164
Il tempo che passa.....	164
Eventi futuri.....	166
Scene.....	167
Regole per ogni turno.....	171
Capitolo 8: Frasi e punteggiatura.....	174
Tipi di testo sorgente.....	175
Inserimenti nel testo.....	176
Virgolette tra virgolette.....	179
Separare i paragrafi.....	180
Ampi blocchi di testo.....	181
Gli spazi.....	182
Oggetti contro regole.....	184
I nomi degli oggetti.....	186
Punteggiatura.....	187
Affermazioni.....	188
Valori.....	189
Aritmetica.....	192
Eeguire calcoli.....	193

Verificare condizioni.....	195
Creare nuove frasi.....	195
Indentazione.....	197
Intestazioni.....	200
Cicli.....	202
Come ottenere maggiori informazioni.....	202
Capitolo 9: Argomenti avanzati.....	203
Tabelle.....	203
La trama.....	207
Utilizzare l'indice.....	208
Contenuti.....	209
Azioni.....	209
Generi.....	209
Frase.....	209
Regole.....	210
Scene.....	210
Il mondo.....	210
Testo indicizzato.....	210
Aiutare il Parser.....	211
Relazioni.....	211
Aggiungere suggerimenti.....	213
Funzionalità speciali di Glulx.....	213
Cosa ha a che fare Inform 6 con Inform 7?.....	214
Appendice A: Glossario.....	215
Appendice B: Licenza.....	218

Prefazione alla traduzione

Questa è la traduzione in italiano di *The Inform 7 Handbook* di Jim Aikin, una guida a Inform 7, il programma in “linguaggio naturale” che permette di creare avventure testuali.

Come noterai nel seguito, il linguaggio di programmazione è identico ad un sottoinsieme dell'inglese, la documentazione inclusa nel programma è tutta in inglese ed i giochi che vengono prodotti sono in inglese. Che senso ha perciò tradurre in italiano la guida introduttiva, se poi il lettore deve sicuramente conoscere bene la lingua inglese per approfondire gli argomenti, oltre che per giocare?

In primo luogo, sono disponibili estensioni per Inform 7 che permettono di produrre avventure testuali in italiano e questo ne rende l'uso appetibile anche ad autori italiani. In secondo luogo, la guida è pensata per principianti (l'autore l'ha creata per i suoi studenti), magari con poche o nulle conoscenze di programmazione, ed è probabile che in questi casi il possesso dell'inglese sia piuttosto incerto.

Per questi motivi può essere utile farsi un'idea di come funziona Inform 7 leggendo una guida in italiano, per comprendere quali sono i costrutti dell'inglese necessari nella programmazione.

Gli esempi originali in inglese sono stati perciò affiancati dalle “traduzioni” necessarie per ottenere lo stesso effetto con le estensioni in italiano. Nei sorgenti proposti, infatti, si evidenzieranno in corsivo le parole italiane, che saranno lette dal giocatore, per distinguerle dai veri costrutti del linguaggio.

Ovviamente, per un autore italiano, il linguaggio di Inform 7 non può essere considerato in “linguaggio naturale” e quindi l'apprendimento diventa simile a quello di altri linguaggi di programmazione. Questo può essere talvolta un vantaggio, perché chi conosce l'inglese potrebbe tentare frasi che hanno per lui significato, ma che per il programma non hanno senso.

Come avviene già per la guida originale, anche per la traduzione è disponibile il file ODT (modificabile con il Writer di OpenOffice) e perciò chi volesse migliorarla può farlo senza difficoltà.

— Leonardo Boselli
Imperia, Italia
Giugno 2009

Come installare le estensioni italiane

Poiché questa guida è rivolta ad autori italiani, è necessario spiegare subito come installare le estensioni per Inform 7 che rendono possibile la creazione di avventure testuali in italiano.

Qual è il primo passo? Scaricare l'Editor per la propria piattaforma dal sito inform7.com ed installarlo seguendo le istruzioni, come spiegato nel Capitolo 1.

Il secondo passo è costituito dall'installazione delle estensioni vera e propria. Si possono scaricare dal sito [Mille e Una Avventura](http://milleuna.sf.net/)¹ (il link diretto è [ItalianExtensions.zip](http://prdownloads.sourceforge.net/milleuna/ItalianExtensions.zip)²). Per installarle tutte in una volta, basta scompattare il file ZIP e copiare le cartelle che compaiono ("Leonardo Boselli" ed eventuali altre) nel percorso Documenti\Inform\Extensions (questo percorso vale per Windows, mentre per gli altri sistemi operativi di dovrà scegliere quello più adatto). Dopo aver svolto queste operazioni, il programma è pronto per creare avventure testuali in italiano.

1 <http://milleuna.sf.net/>

2 <http://prdownloads.sourceforge.net/milleuna/ItalianExtensions.zip?download>

Introduzione

Una volta qualcuno ha detto che se chiediamo a cinque poeti di definire la poesia, otterremo sette definizioni. Sono sicuro di citare in modo molto approssimativo — la versione originale poteva non avere nulla a che fare con la poesia — ma il significato dovrebbe essere chiaro comunque. Senza dubbio ci sono più modi efficaci per spiegare le sottigliezze del linguaggio di programmazione di Inform 7 di quanti siano coloro che hanno scritto buoni tutorial sull'argomento.

Questo libro non pretende di essere la guida finale a Inform 7. Al contrario: un certo numero di argomenti che autori esperti di Inform vorrebbero approfondire non sono trattati in questa *Guida di Inform 7*. Lo scopo di questo libro è, invece, quello di fornire ai principianti il tipo di informazioni e approfondimenti di cui hanno bisogno per iniziare ad usare Inform 7 senza essere confusi da un numero eccessivo di sottigliezze.

Ho cercato di organizzare il materiale per aiutare gli autori inesperti a reperire velocemente le informazioni utili. Ho fatto del mio meglio per presentare gli argomenti in modo tale che il lettore non abbia bisogno di alcun prerequisito. E' possibile che si trovino varie ripetizioni leggendo la *Guida* dall'inizio alla fine, ma ritengo che un minimo di ridondanza sia meglio che forzare i lettori alla ricerca delle informazioni.

Questo libro nasce dalle mie esperienze di insegnamento a giovani studenti (dai dieci ai quindici anni) su come scrivere *interactive fiction*³ utilizzando Inform 7. Inform 7 (abbreviato "I7") viene distribuito con due ampi e dettagliati tutorial/guide di riferimento, *Writing with Inform*⁴ e *The Inform Recipe Book*⁵ (noti come la Documentazione). Prima di addentrarsi troppo nel processo di scrittura di un gioco, ogni autore di Inform dovrebbe leggere la Documentazione! In effetti dovrebbe farlo più volte. Questo è il modo migliore per acquisire dimestichezza con la potenza di Inform.

Quando gli studenti mi chiedono come realizzare quelle cose che solitamente gli autori principianti vogliono ottenere, ho spesso notato che le informazioni di cui avevano bisogno erano sparse per tutta la Documentazione, rendendo difficile trovare tutti i pezzi per combinarli in un quadro coerente. Comprendere come realizzare alcuni dei più semplici algoritmi di programmazione solo leggendo la Documentazione può richiedere un certo studio. A questo proposito, ho sentito commenti simili anche da parte di principianti adulti.

A volte la Documentazione sembra affermare che una piccola spinta nella giusta direzione è tutto ciò di cui il lettore ha bisogno, ma istruzioni passo-passo e discussioni approfondite su molti dettagli che gli autori dovrebbero avere a portata di mano non sempre sono accessibili facilmente. Inoltre, sebbene la Documentazione sia molto leggibile, è stata scritta da adulti per adulti. Non tutta la terminologia è comprensibile a giovani studenti.

3 Narrativa interattiva: un termine altisonante con cui le avventure testuali sono state etichettate una volta che hanno assunto dignità artistica e letteraria [N.d.T.]

4 Scrivere con Inform [N.d.T.]

5 Il ricettario di Inform [N.d.T.]

La *Guida di Inform 7* utilizza un approccio differente. Le informazioni di base orientate a determinati obiettivi sono organizzate in capitoli. Ad esempio, è disponibile un capitolo per realizzare stanze, un capitolo su come creare personaggi e un capitolo su come gestire i comandi che il giocatore potrebbe scrivere. Nessuno dei capitoli spiegherà ogni aspetto di un certo argomento e quindi, dopo aver letto un paragrafo della *Guida*, si vorrà spesso rileggere il paragrafo corrispondente della Documentazione. Per facilitare il lettore, ho quindi incluso in molti punti i riferimenti incrociati alle pagine da consultare.

Inform 7 non è l'unico linguaggio disponibile per scrivere *interactive fiction* (abbreviata con “IF”). Il suo concorrente principale (se ha senso dire che il *software* libero può fare concorrenza) è probabilmente TADS 3. La versione 3 di TADS (*The Adventure Development System*) è, in qualche modo, perfino più sofisticato di Inform 7. Però la sua interfaccia utente — il modo in cui si presenta all'autore — è differente da I7 come il giorno dalla notte. TADS 3 rispecchia molto fedelmente un classico linguaggio di programmazione come il C (ma se non sai cosa significa, non preoccuparti).

L'interfaccia di programmazione in “linguaggio naturale”⁶ di I7 rende I7 molto attraente per coloro che non hanno competenze specifiche nella programmazione dei computer. Inoltre, mentre sto scrivendo, l'ambiente di sviluppo di TADS 3, chiamato Workbench, è un'applicazione che può essere eseguita solo sotto Windows. I giochi TADS possono essere giocati sul Mac o su un computer Linux, ma gli strumenti di sviluppo sono specifici per Windows. Per l'utilizzo in classe, Inform 7 è una scelta migliore non solo perché il “linguaggio naturale” può rendere più semplice per un principiante la comprensione degli aspetti del sistema, ma anche perché non fa alcuna differenza se lo studente utilizza un Macintosh, Windows o un computer Linux.

Se la complessità senza compromessi di TADS 3 o l'approccio amichevole ma talvolta imprevedibile di Inform 7 non ti soddisfano, puoi anche valutare Inform 6, che è un linguaggio completamente differente, sebbene abbia in comune il nome “Inform”, o un semplice sistema come ADRIFT o ALAN. ALAN è multi-piattaforma (Mac, Windows, Linux), mentre ADRIFT è solo per Windows. La versione completa di ADRIFT non è gratuita, ma ha un prezzo accessibile. Tutti questi sistemi, come anche altri, sono disponibili per il download via Web. La migliore recensione che io conosca che mette a confronto Inform 7 con TADS 3 è stata scritta da Eric Eve; è disponibile sul sito *Brass Lantern*⁷. Alcuni confronti con altri sistemi sono disponibili all'indirizzo www.firthworks.com/roger/cloak/, ma nel momento in cui scrivo (maggio 2009) quest'ultima pagina è vecchia di quasi dieci anni. Stephen Granade fornisce alcuni suggerimenti ed elementi di confronto in un saggio sempre su *Brass Lantern*⁸.

Come lo stesso Inform, La *Guida di Inform 7* è gratuita. Spero che sia utile e spero anche di ricevere messaggi email (in inglese [N.d.T.]) sulle parti che possono essere migliorate o approfondite (mentre scrivo, buoni indirizzi email sono midiguru23@sbcglobal.net e editor@musicwords.net). Versioni aggiornate della *Guida* saranno pubblicate di tanto in tanto.

— Jim Aikin
Livermore, California
Maggio 2009

6 Almeno per quanto riguarda un autore di lingua inglese [N.d.T.]

7 <http://www.brasslantern.org/writers/iftheory/tads3andi7.html>

8 <http://www.brasslantern.org/writers/howto/chooselang.html>

Ringraziamenti

I ringraziamenti vanno prima di tutto e soprattutto a Graham Nelson per il progetto monumentale di creare e mantenere Inform. Emily Short è instancabile nell'assistere e supportare gli autori di Inform e ha fornito numerosi commenti significativi sulla versione quasi definitiva della *Guida*. David Kinder si occupa della programmazione dell'IDE (*Integrated Development Environment*) di Inform 7 su Windows e Andrew Hunter dell'IDE su MacOS. Il sistema Glulx, programmato da Andrew Plotkin, permette di creare giochi con Inform di maggiori dimensioni e con funzionalità aggiuntive; senza Glulx, Inform avrebbe molto meno attrattiva tra i designer di giochi.

Come lo stesso Inform, la *Guida di Inform 7* è, per gran parte, il prodotto dell'impegno di una comunità. Versioni preliminari sono state lette da Michael Callaghan, Eric Eve, Ron Newcomb, Emily Short, Mike Tarbert e Michael Neal Tenuis, ognuno dei quali ha proposto aggiunte e chiarimenti significativi. Ho anche sfruttato materiali sviluppati da Mark Engelberg per altri corsi. Durante la stesura, vari esperti sul gruppo di discussione rec.arts.int-fiction hanno liberamente condiviso le loro conoscenze con me. Ho anche utilizzato alcune domande poste su rec.arts.int-fiction da programmatori principianti ed ho sfruttato liberamente le domande poste dagli studenti nei miei corsi sull'IF.

La *Guida di Inform 7* è stata scritta con OpenOffice Writer, che gestisce molto bene, nel PDF che genera, i riferimenti incrociati ipertestuali nell'indice dei contenuti.

Le immagini usate in questo file (eccetto quelle presi dall'IDE di Inform) sono stati raccolti da vari siti Web; sarebbe veramente difficile dopo tanto tempo individuare chi deve essere ringraziato per averle fornite. I detentori del Copyright che potrebbero obiettare sull'uso senza fini di lucro del loro lavoro dovrebbero contattarmi: sarò felice di rimuovere le loro immagini dalla *Guida*.

Grazie a tutti!

L'autore

Jim Aikin ha pubblicato vari giochi di *interactive fiction*, inclusi "Not Just an Ordinary Ballerina" (scritto in Inform 6), "Lydia's Heart" (in TADS 3), "April in Paris" (TADS 3), "Mrs. Pepper's Nasty Secret" (scritto a due mani con Eric Eve in TADS 3) e "A Flustered Duck" (in Inform 7). E' anche autore di due racconti di fantascienza, *Walk the Moons Road* (Del Rey, 1985) and *The Wall at the Edge of the World* (Ace, 1992). I suoi racconti brevi sono apparsi su *Fantasy & Science Fiction*, *Asimov's Science Fiction* e altre riviste. I suoi libri di saggi includono *A Player's Guide to Chords & Harmony* e *Power Tools for Synthesizer Programming* (entrambi pubblicati da Hal Leonard). Il suo sito personale è www.musicwords.net.

Capitolo 1: Come iniziare



Così... vorresti scrivere *interactive fiction*. Sembra divertente, però non sai da dove iniziare. Può darsi che tu abbia già iniziato e ora sei confuso, o non sai come ottenere i risultati che desideri.

Questo libro ti aiuterà a dare un senso a tutto ciò. Ogni capitolo si occupa di una parte specifica del processo di scrittura, così non è necessario che tu legga il libro dall'inizio alla fine. Sei libero di saltare da una parte all'altra, affrontando qualunque capitolo sembri contenere le informazioni che ti servono. Per esempio, se vuoi scrivere una storia con molti personaggi, potresti saltare direttamente al Capitolo 5. Se pensi di avere problemi nello scrivere le frasi del linguaggio in modo che Inform le capisca, puoi

consultare il Capitolo 8. Comunque alcuni degli argomenti in quest'ultimo capitolo sono piuttosto avanzati e quindi potrebbe non essere il punto migliore da cui iniziare per imparare (ed ecco perché è il Capitolo 8 e non il Capitolo 1).

Perfino in questo Capitolo 1 ci sono argomenti di cui non hai ancora bisogno di preoccuparti quando sei solo all'inizio. Se incontri qualcosa che sembra non avere senso, sentiti libero di ignorarlo per il momento e di riprenderlo più avanti.

Sia che tu legga questo libro, o qualche altro manuale, o che ti tuffi e provi a scrivere senza preoccuparti per niente di leggere un manuale, imparerai presto che scrivere *interactive fiction* (abbreviato con "IF") comporta più competenze della semplice scrittura creativa. La creatività ed il saper raccontare fanno sicuramente parte del processo, ma dovrai anche avere a che fare con la programmazione dei computer. "Programmazione" significa che dovrai fornire al computer le informazioni su cosa eseguire ed i computer sono particolarmente puntigliosi quando devono capire le istruzioni che gli sono impartite! Appena inizierai a scrivere IF, scoprirai che una singola parola scritta in modo errato o un punto mancante al termine di una frase possono interrompere l'esecuzione del programma. Più avanti nel capitolo, ti fornirò una lista di alcuni errori che vedo spesso commettere dagli studenti.

Questo libro si occupa dell'uso del popolare linguaggio di programmazione Inform 7 (abbreviato in "I7"). Ci sono altri linguaggi con cui si può scrivere *interactive fiction*, come ho già detto nell'introduzione. Se ne usi già uno, in questo libro troverai alcuni suggerimenti generali che puoi sfruttare. Ma il libro è principalmente sull'uso di I7. Ritengo che I7 sia una scelta particolarmente adatta a coloro che non sono esperti nella scrittura di IF e nuovi alla programmazione dei computer. I7 è pensato per rendere la fase iniziale di apprendimento il più facile possibile. Allo stesso tempo, I7 include molte funzionalità per gli utenti avanzati, così non sarai limitato in alcun modo se, in un secondo momento, vorrai includere nella tua storia elementi più complessi.

Inform 7 è il successore di Inform 6, che è un linguaggio per scrivere IF altrettanto popolare. I due sono completamente differenti, anche se sono stati creati dalla stessa persona, Graham Nelson, e sono chiamati entrambi “Inform”. In questo libro, mi riferirò spesso a “Inform” senza la preoccupazione di aggiungere il 7, ma tutti questi riferimenti riguardano Inform 7 e non Inform 6.

E' importante sottolineare che Inform 7 è, nel momento in cui scrivo (maggio 2009), ancora in fase di sviluppo. In questi tre anni, molti lo hanno utilizzato con successo per scrivere *interactive fiction*, ma non è ancora ufficialmente terminato. La versione attuale è la 5Z71. Se alcuni dettagli di questo libro non sembrano corretti, è possibile che tu stia usando una versione più recente di Inform, in cui certe caratteristiche sono cambiate. I fondamenti del linguaggio di I7 sono piuttosto ben sviluppati e probabilmente non cambieranno molto in futuro, ma molto probabilmente nuove funzionalità saranno aggiunte e alcuni dettagli potrebbero essere modificati.

Inoltre, è molto probabile che cambino la numerazione delle pagine e degli esempi della documentazione di Inform. Nota che la numerazione qui utilizzata si riferisce alla versione 5Z71.

Download e installazione

Per ottenere la versione più recente di Inform, visita il suo sito Web e naviga fino alla pagina dei Download (<http://inform7.com/download/>). Seleziona l'icona che mostra il sistema operativo del tuo computer (Macintosh, Windows o qualche varietà di Linux) e scarica il programma. Sulla stessa pagina puoi leggere le istruzioni su come installarlo su ciascuno dei sistemi operativi proposti.

Lavorando con Inform, creerai dei *progetti*. Ogni gioco creato è un progetto separato. E' una buona idea creare una cartella chiamata “Inform Projects” all'interno della cartella dei Documenti. Questo potrebbe essere fatto automaticamente durante l'installazione. Ogni volta che crei un nuovo progetto, salvalo in quella cartella. Quando avrai salvato il tuo primo progetto in quella cartella, Inform si ricorderà di quella collocazione per te.

E' un gioco o una storia?

Nella *Guida di Inform 7* ci riferiremo ai progetti Inform sia come “storia” che come “gioco”, senza fare molta distinzione tra i due termini. Ci sono però alcune reali differenze: alcune *interactive fiction* hanno una trama importante e molto poco del divertimento di un gioco. Altre invece sono principalmente giochi e la storia è così debole che potrebbe non esistere per nulla.

Sta a te decidere come combinare gli elementi della storia con quelli ludici. Per gli scopi di questo libro, il gioco è una storia e la storia è il gioco.

Il programma Inform 7

La prima volta che esegui Inform 7, una finestra di dialogo ti chiederà se vuoi aprire un progetto esistente o crearne uno nuovo. Poiché non hai ancora un progetto da aprire, seleziona “Start a new project...” (inizia un nuovo progetto). Si aprirà una finestra in cui puoi attribuire un nome al tuo progetto e specificare a Inform dove vuoi che sia salvato (utilizza, ad esempio, la cartella Inform Projects in Documenti).

Quando chiuderai questa finestra, Inform continuerà l'esecuzione e vedrai la finestra principale suddivisa in due pagine — una a destra e una a sinistra. La pagina di sinistra sarà vuota eccetto per il nome del tuo nuovo gioco ed il tuo stesso nome come autore. Invece, sulla pagina destra vedrai l'indice della documentazione di Inform.

Ciò che vedi non è veramente Inform (anche se puoi pensarlo, se vuoi). In realtà vedi un programma che è una sorta di contenitore per Inform. Questo tipo di programma è noto come IDE (Integrated Development Environment). L'IDE di Inform ha numerose funzionalità. Prima di iniziare a scrivere la tua storia, dai un'occhiata all'IDE. Le sue funzionalità più importanti sono discusse nei prossimi paragrafi di questo capitolo, mentre altre sono menzionate nel Capitolo 9.

Solitamente terrai il testo della storia che stai scrivendo nella pagina a sinistra, mentre in quella di destra vedrai pannelli differenti in momenti diversi — la Documentazione, il gioco stesso, i messaggi di errore e così via. Talvolta l'IDE aprirà nuovi pannelli automaticamente nella pagina di destra, ma in effetti puoi visualizzare quello che desideri in entrambe le pagine. Per esempio, se stai studiando due diverse pagine della documentazione, puoi aprirne una a sinistra e l'altra a destra nello stesso momento.

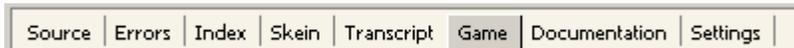
Per scegliere ciò che viene mostrato in ciascuna pagina, seleziona una delle opzioni nella riga delle intestazioni di pagina (è mostrata in testa alle pagine nell'IDE su Windows, mentre è disposta verticalmente sul bordo destro nell'IDE per il Macintosh). Ogni linguetta apre un pannello differente.

Fino a che non hai scritto qualcosa nel pannello del sorgente e utilizzato il pulsante Go!, alcuni degli altri pannelli saranno vuoti. E' normale.

Prova a cliccare sui titoli dei capitoli nella documentazione per leggere alcune pagine. Se vedrai cose che non puoi capire per il momento, ma non devi preoccuparti. Perfino nel Capitolo 1 della Documentazione alcune informazioni sono piuttosto avanzate e non sono all'inizio. Dai solo un'occhiata e prendi dimestichezza su come la Documentazione è organizzata.

Nella *Guida di Inform 7*, i riferimenti alla “Documentazione” sono sempre relativi a *Writing with Inform*. Al fondo dell'indice di *Writing with Inform* vedrai un collegamento a *The Inform Recipe Book*. Questi sono manuali completamente separati. Meritano entrambi di essere letti. Condividono lo stesso insieme di esempi, che vale la pena di studiare attentamente.

Le intestazioni di pagina



Lungo la parte superiore di ogni pagina, se stai usando l'IDE per Windows, o verticalmente lungo un bordo in quella per il Macintosh, vedrai la serie di linguette dei pannelli, o intestazioni: Source, Errors, Index, Skein, Transcript, Game, Documentation e Settings. Selezionando uno di questi si apre un nuovo pannello. Tratteremo i dettagli di come usare i pannelli Skein, Transcript e Index nel Capitolo 9. In questo capitolo forniremo solo una breve introduzione.

Il pannello **Source** (sorgente) è quello in cui scriverai il gioco. In testa al pannello vedrai due pulsanti: Source (sorgente) e Contents (contenuti). Quando inizi a creare il tuo primo gioco, il pannello Contents non è utile, infatti è vuoto. Invece, in progetti di grandi o anche di medie dimensioni, è molto utile aggiungere titoli e sottotitoli (vedi “Intestazioni” nel Capitolo 8) al sorgente. “Source” nella terminologia dei computer è ciò che stai scrivendo. Quando avrai aggiunto alcune intestazioni, il pannello Contents fornirà un metodo rapido per spostarti all'interno del sorgente. Puoi saltare velocemente da un paragrafo all'altro, magari distante dal primo centinaia di linee, senza dover utilizzare le barre di scorrimento ai lati del pannello ed andare a caccia del paragrafo che stai cercando.

Il pannello Source include un'utile ed importante funzionalità chiamata **colorazione della sintassi**. Appena inizierai a lavorare con Inform, vedrai che tipi differenti di testo nel sorgente saranno evidenziati con diversi colori. Ad esempio, il testo tra virgolette sarà sempre blu. La colorazione della sintassi è utilizzata in molte forme di programmazione dei computer. E' una funzionalità che rende più semplice leggere ciò che si sta scrivendo, ma i colori non hanno alcun effetto sul gioco stesso.

Sorgente: codice o testo?

I programmatori di computer chiamano ciò che scrivono *codice sorgente*. Il termine “codice” risale agli albori della programmazione dei computer e probabilmente rispecchia il fatto che molti linguaggi di programmazione sembrano astratti e difficili da comprendere come messaggi scritti in un codice segreto. I creatori di Inform preferiscono il termine *testo sorgente*, perché ritengono che “testo” sia un termine più efficace per descrivere ciò che scriverai. La maggior parte delle volte, la *Guida di Inform 7* chiamerà ciò che scriverai “codice sorgente” o semplicemente “codice”, perché “testo” in Inform si riferisce anche alle frasi che si scrivono tra virgolette, che costituiscono l'output mostrato durante il gioco.

Il pannello **Errors** (errori) si apre automaticamente nella pagina di destra quando si clicca sul pulsante Go! (vedi più sotto) se Inform non capisce quello che hai scritto. Tranne quando si ha a che fare con qualche problema, il pannello Errors si può ignorare.

Il pannello **Index** (indice) è estremamente utile. Comunque, finché non è stato compilato il primo gioco (vedi “Il pulsante Go!” più sotto), l'indice sarà vuoto. Nel processo di convertire il sorgente in un gioco,

Inform crea praticamente liste di tutto ciò che c'è nel gioco. Consultando le pagine dell'indice, è possibile imparare molto di ciò che riguarda il funzionamento di Inform ed anche individuare i problemi potenziali del gioco, o semplicemente essere gratificati nel vedere che il gioco sta crescendo.

Il pannello **Index** è diviso in sette pagine: Contents (contenuti), Actions (azioni), Kinds (generi), Phrases (frasi), Rules (regole), Scenes (scene) e World (mondo). Ne parleremo più diffusamente nel Capitolo 9.

Il pannello **Skein** (trama) è utilizzato principalmente per verificare il funzionamento del gioco mentre questo viene scritto. Il pannello Skein è un utile strumento nel momento in cui il gioco comincia ad aumentare di dimensioni, ma quando si è appena agli inizi, si può ignorarlo senza problemi. In sintesi, il pannello Skein mantiene una traccia di tutte le sessioni giocate mentre si verifica come procede il lavoro. Uno dei modi per verificare il lavoro svolto dopo aver apportato qualche modifica è quello di aprire il pannello Skein, cliccare con il tasto destro (Mac: cliccare con il pulsante CTRL premuto) su uno dei *nodi* a forma di losanga e scegliere “Play to Here” (gioca fino a qui) dal menù contestuale. Questa azione ripeterà i comandi che sono stati impartiti in quella particolare sequenza. Per maggiori dettagli sul pannello Skein, è possibile consultare le **pagg. 1.7, 1.8 e 1.9** (“The Skein”, “A short Skein tutorial” e “Summary of the Skein and Transcript”) nella Documentazione.

Il pannello **Transcript** (trascrizione) è utile quando è necessaria una programmazione di tipo avanzato. Ad essere onesto, nello scrivere il mio primo gioco con Inform 7, non l'ho mai utilizzato. Per avere maggiori informazioni su come utilizzare i pannelli Skein e Transcript, vedi il Capitolo 9.

Il pannello **Game** (gioco) è quello in cui apparirà il gioco, permettendo di provarlo mentre lo si sviluppa. Non è possibile eseguire altri giochi nel pannello Game, solo quello che si sta scrivendo. L'aspetto del gioco nel pannello Game sarà simile, ma probabilmente non identico, a come apparirà in un programma interprete. I giocatori eseguiranno il gioco in un interprete (vedi “Scaricare ed eseguire giochi” più avanti in questo stesso capitolo) che è un programma differente. Il testo del gioco dovrebbe essere sempre lo stesso, non importa su quale interprete sarà giocato, ma l'aspetto visivo ed il tipo di carattere potrebbero cambiare.

Come regola generale, basarsi sull'aspetto visivo del gioco per fornire ai giocatori informazioni importanti è una pessima idea. Non solo l'aspetto varia da un interprete all'altro, ma l'*interactive fiction* è popolare anche tra gli utenti non vedenti, perché possono usufruire di questo tipo di giochi. Di norma, tutto ciò che il giocatore deve sapere per giocare dovrebbe essere presentato come testo ordinario.

Il pannello **Documentation** (documentazione) contiene due ampi tutorial/testi di riferimento: *Writing with Inform* e *The Inform Recipe Book*. Entrambi sono risorse di grande valore ed è bene conoscerli. Anche nel pannello Documentation è presente un indice di oltre 400 esempi che mostra come usare Inform. Nella *Guida di Inform 7*, ci riferiamo a *Writing with Inform* come alla “Documentazione”.

Il pannello **Settings** spesso non è necessario. Ad un certo punto, comunque, quando il gioco sarà diventato abbastanza grande, apparirà un messaggio di errore in fase di compilazione. A questo punto, è necessario aprire il pannello Settings e cambiare la versione del formato Z-code da 5 alla versione 8, o dal formato Z-code a Glux. Se il gioco contiene immagini o suoni, occorre selezionare anche la casella

“Create a Blorb archive for release” (crea un archivio Blorb per la pubblicazione), prima di premere il pulsante Release nella barra degli strumenti principale. L'abilità di Inform di gestire immagini e suoni può essere utile, ma è anche abbastanza limitata. Ne parleremo molto poco in questo libro.

Il pulsante Go!



In cucina ho una magnifica invenzione chiamata la macchina del pane Zojirushi. Produce pane fatto in casa. Tutto quello che devo fare è misurare gli ingredienti grezzi, versarli nella teglia, chiudere il coperchio e premere il pulsante di avvio. Piccole palette all'interno della macchina mescolano l'impasto. Poi la macchina attende un'ora, affinché l'impasto lieviti, prima di scaldarlo e di iniziare a cuocerlo. Tre ore dopo aver versato gli ingredienti, ottengo una pagnotta fumante.

Il pulsante Go! (vai!) di Inform è molto simile al pulsante di avvio della mia macchina per il pane. Il testo scritto nel pannello Source corrisponde agli ingredienti grezzi — la farina, l'acqua, lo zucchero, il lievito e così via. Quando viene premuto il pulsante Go!, Inform impasta ciò che è stato scritto e lo converte in un gioco pronto per l'uso. Questa funzione è spiegata nel **par. 1.4** (“The Go! button”) della Documentazione.

Questo processo sembra molto semplice — basta premere un pulsante. In realtà Inform esegue un'enorme quantità di lavoro per tradurre il testo sorgente in un gioco. Questo lavoro è portato a termine da una “macchina” software chiamata **compilatore**. Non occorre preoccuparsi nei dettagli di ciò che fa il compilatore, ma poiché ci riferiremo ad esso qui e là in questo libro, occorre sapere che esiste e ciò che fa. Il compilatore converte il testo sorgente in un gioco.

Se metto troppa acqua nella mia macchina per il pane, la macchina non lo può sapere. Semplicemente produrrà una “pagnotta” che sarà un pasticcio bagnato. Se dimentico di aggiungere il lievito, la macchina andrà avanti e farà il suo lavoro, ma produrrà una duro blocco spezza-denti. La macchina per il pane non controlla ciò che metto nella casseruola — porta semplicemente avanti il suo processo in modo automatico.

Un compilatore è molto più intelligente di una macchina per il pane. Per fare il suo lavoro, il compilatore di Inform deve considerare ogni singola linea di ciò che è stato scritto e capire che cosa farne. Se capisce ciò che è stato scritto, dopo pochi secondi aver premuto il pulsante Go!, il gioco apparirà nella pagina di destra, pronto per essere eseguito. Più della metà delle volte, però, il compilatore incontrerà dei problemi. Incontrerà una parola, una frase o un paragrafo che non riesce a capire. Invece di veder apparire il gioco nella pagina di destra, comparirà una pagina che fornirà un rapporto sui problemi riscontrati.

Non è nulla di cui preoccuparsi — è normale. L'unica cosa che si deve fare è trovare l'errore, correggerlo e premere Go! di nuovo. A seconda dei problemi specifici, occorrerà ripetere questa sequenza cinque o sei volte prima che il sorgente si compili con successo.

Accanto alla spiegazione di ogni errore nella pagina di destra, vedrai una piccola freccia arancione. Quando la clicchi, il pannello Source salterà immediatamente al punto incriminato, che sarà anche

evidenziato. Ora sai cosa devi correggere (talvolta devo ricordare agli studenti di leggere prima il messaggio e poi cliccare sulla freccia).

Ogni messaggio fornisce un certo numero di informazioni sul tipo di problema riscontrato. A volte queste informazioni saranno subito comprensibili. Altre volte, possono confondere invece di essere utili. Ad esempio, se premi un punto e virgola (;) dove invece dovevi premere un due punti (:), Inform non dirà che il problema è quello. Invece, leggerai un paragrafo più o meno di questo tipo, “The phrase or rule definition ... is written using the 'colon and indentation' syntax for its 'if's, 'repeat's and 'while's, where blocks of phrases grouped together are indented one tab step inward from the 'if ...:' or similar phrase to which they belong. But the tabs here seem to be misaligned, and I can't determine the structure.” (“La frase o la definizione della regola ... è scritta utilizzando la sintassi 'due punti ed indentazione' per i suoi 'if's, 'repeat' e 'while', dove i blocchi delle frasi sono raggruppati assieme ed indentati una tabulazione oltre 'if ...:' o frasi simili a cui appartengono. Le tabulazioni però sembrano allineate male e non riesco a determinarne la struttura”). In questo caso, Inform specificherà anche la linea esatta dove ha incontrato il problema — ma la linea che identifica è la linea *dopo* l'errore di battitura.

Dopo un po' inizierai a comprendere quale genere di errori commetti più comunemente ed i messaggi d'errore che generano. Per maggiori dettagli, vedi “Tutto sui Bug” più avanti in questo capitolo. Se tutto va bene, però, il tuo gioco apparirà nella pagina di destra, nel pannello Game, pronto per essere provato.

Cinque problemi comuni

Osservando i principianti che iniziano a studiare Inform, vedo determinati tipi di errori che si ripresentano continuamente. Senza seguire alcun ordine particolare, qui ci sono alcune cose che devi controllare nel tuo codice:

Dimenticare di dire “say”. Vuoi che il gioco produca un certo output in una determinata situazione, così la scrivi semplicemente, circondandola come al solito con le virgolette. In un paio di casi, come sono le descrizioni delle stanze, questo è corretto, ma se stai scrivendo una regola in cui vuoi che Inform dica qualcosa, devi dire “say” (di) prima di iniziare il testo virgolettato.

Punto mancante alla fine di una frase. Termina sempre le frasi con un punto. Ci sono speciali regole per inserire punti tra virgolette⁹, ma se la frase non ha un punto terminale, Inform si confonde. Se la frase è nel mezzo di un paragrafo, Inform quasi certamente si confonderà.

Parole scritte erroneamente. E' un problema in cui si incorre spesso ed è difficile da scoprire. Devi scrivere molte volte la parola “description” e su uno schermo ad alta risoluzione puoi non accorgerti di averla scritta “descripton” o “descrpition”.

9 Ovvie per un lettore di lingua inglese, ma assolutamente assurde per un italiano [N.d.T.]

Virgola invece del punto e virgola o vice-versa. Questi due segni si somigliano molto sullo schermo, ma sono completamente differenti. Un punto e virgola è come un segnale di stop sulla strada; dice a Inform, “Ok, fermati qui per un attimo, poi continua, perché non abbiamo ancora finito”. Una virgola è come una freccia che punta avanti — dice a Inform: “Dopo fai questo”.

Errate indentature. Vedi pagina 197 nel Capitolo 8 della *Guida* per una spiegazione completa di come Inform tratta l'indentazione.

Una cosa devi sapere sul pulsante Go!. La prima cosa che Inform fa, quando lo premi, è quella di salvare il sorgente sul disco. Questo nuovo file sovrascrive tutto ciò che c'era in precedenza nello stesso file. Così se stai provando varie modifiche del tuo gioco, è preferibile creare delle versioni alternative — una per sperimentare e una come copia di sicurezza. Per fare una copia con cui sperimentare, usa il comando “Save As...” (salva come...) del menù File.

Il pulsante Release



Quando il gioco è finito — o forse non proprio finito, ma abbastanza avanzato da poter essere distribuito ad altre persone che così possono testarlo, trovare problemi e proporre suggerimenti — puoi premere il pulsante Release (pubblicazione). Questo produrrà un gioco in formato .z5, .z8 o Glulx a seconda dei Settings che hai scelto. Puoi dare questo file ad altre persone e inserirlo come attachment in un'email ed inviarglielo. Potranno caricarlo in un interprete e giocarlo.

Alcune delle funzionalità utilizzabili nell'IDE in fase di test (vedi “Usare i comandi di Debug” più avanti in questo capitolo) non sono disponibili nella versione di rilascio.

Altre funzionalità

Il pulsante Stop (arresta) non è usato spesso, ma non è possibile fare certe modifiche nel pannello di Skein mentre il gioco è in esecuzione (sarebbe come indossare un nuovo paio di lame mentre si pattina su una pista ghiacciata). Il pulsante Stop terminerà la sessione di gioco corrente, rendendo possibili le modifiche allo Skein.

Come utilizzare la documentazione

Molti utilizzano la documentazione di Inform come un tutorial combinato ad una guida di riferimento. All'inizio la userai più come un tutorial. Per fare questo, leggi i capitoli dal 2 al 7 cercando gli esempi. Gli esempi possono essere facilmente copiati in un gioco di prova così puoi provarli e sperimentare attraverso di essi le varie tecniche.

Per prima cosa, inizia un nuovo progetto o apri un progetto che puoi cancellare. Io ho un progetto chiamato Test Game sul mio disco rigido proprio per questo scopo (in effetti, ne ho dozzine). E' meglio iniziare con un sorgente vuoto quando si provano gli esempi, perché il loro inserimento in un gioco esistente renderebbe il tutto un pasticcio.

Apri l'esempio cliccando sul pulsante blu numerato a forma di losanga. Iniziando con un pannello Source vuoto nel tuo Test Game, clicca sul pulsante quadrato nella prima linea dell'esempio. Il pulsante copierà l'intero esempio nel pannello Source. Ora devi semplicemente cliccare Go! e l'esempio diverrà un gioco. Dopo aver giocato per vedere cosa fa, prova a cambiare il sorgente e provalo di nuovo. Questo è un ottimo modo per capire come funziona Inform.

Quando avrai letto gran parte della Documentazione (forse varie volte), diventerà più facile trovare le informazioni di cui hai bisogno. La Documentazione non ha indice, ma l'IDE include un campo di ricerca. Puoi scrivere ciò che vuoi in questo campo e Inform leggerà tutta la Documentazione alla ricerca di corrispondenze. Potresti non trovare alcuna corrispondenza oppure dozzine. Più preciso sei su ciò che vuoi cercare, meglio lavorerà il motore di ricerca. Frasi come "end the game" (fine del gioco) e "omit contents in listing" (ometti i contenuti nelle liste) funzionano meglio di "scenery" (scenario), che produce troppi risultati.

Attualmente (nella versione 5Z71), il motore di ricerca non trova testo nel *Recipe Book*, solo in *Writing with Inform* e negli esempi.

Quando hai compilato il tuo sorgente con successo, il pannello Indice conterrà molti collegamenti (i punti interrogativi blu) alla Documentazione. Ad un certo punto, vorrai esplorare l'indice, perché elenca collegamenti a molte informazioni importanti.

Cosa avviene in un gioco

Se hai giocato alcune avventure testuali, non hai bisogno di sapere come funziona l'IF. Ma a beneficio di coloro che hanno appena iniziato, qui tratterò degli aspetti di base. In un gioco, reciti la parte di un personaggio della storia. La storia può essere basata su un'idea molto semplice o può essere complessa, coinvolgendo molti personaggi, luoghi ed eventi. In una storia semplice, tu (il personaggio) potresti vagare in una grotta, collezionando tesori. In una storia complessa probabilmente incontrerai altri personaggi. Potrai combatterli o diventargli amico. La storia può avere diversi finali, alcuni di essi a lieto fine ed altri invece negativi. Uno dei tuoi scopi è quello di scegliere le azioni che portano ad un *happy end*, ma finché non cominci ad agire, non saprai quali sono le azioni corrette.

Inserire comandi

Per giocare, scrivi *comandi* (istruzioni al computer) quando vedi la richiesta dei comandi. Solitamente ha questo aspetto:

>

A differenza di un videogame, dove hai bisogno di riflessi pronti per gestire personaggi animati che si muovono nell'ambiente circostante, in un'avventura testuale non avviene nulla finché non scrivi un comando e premi il tasto Invio del computer. Dopo aver premuto Invio, il gioco legge il tuo ultimo comando e stampa una risposta.

Alcune risposte sono messaggi d'errore: forse parole scritte male, ad esempio, perché il gioco non ha compreso quello che volevi dire. Altre risposte mostrano ciò che sta accadendo mentre la storia si sviluppa. Il comando che hai inserito può causare un cambiamento nel mondo virtuale della storia, o ti può fornire semplicemente più dettagli su ciò che sta accadendo. Se sei solo un giocatore non hai bisogno di fermarti per riflettere su come il gioco interpreta i tuoi comandi e risponde ad essi. Ma se vuoi essere un autore di avventure testuali, devi conoscere anche questi aspetti. Il congegno software che legge ed interpreta ciò che il giocatore scrive è chiamato **parser**. Ogni avventura testuale ha un parser. Il parser che è incluso in Inform 7 è molto sofisticato. E' capace di interpretare un'ampia varietà di comandi inseriti dall'utente. (Puoi anche cambiare il suo comportamento, se ne hai bisogno. Alcune tecniche sono spiegate in questa *Guida*).

Comunque nessun parser è capace di comprendere tutti i tipi di frasi articolate che ci scambiamo nella vita di ogni giorno. Il parser è programmato per comprendere semplici frasi, come ad esempio OPEN THE DOOR (apri la porta) e PICK UP THE BALL (prendi la palla). Molti comandi si presentano nella forma <VERBO> <NOME> <FORSE UNA PREPOSIZIONE> <FORSE UN ALTRO NOME>.

Alcuni verbi come WAIT, SLEEP e JUMP (aspetta, dormi e salta) non sono seguiti da alcun nome. Altri verbi invece ne hanno bisogno — per esempio, OPEN THE DOOR e PICK UP THE BALL. Alcuni verbi sono seguiti da due nomi. Esempi possono essere PUT THE BOX ON THE TABLE (metti la scatola sul tavolo) e HIT THE OGRE WITH THE STICK (colpisci l'orco con il bastone).

Quando vengono inseriti dei comandi durante il gioco, l'articolo THE (o gli articoli italiani) può sempre essere evitato, infatti molti giocatori esperti non lo usano mai. Scriverebbero semplicemente HIT OGRE WITH STICK (colpisci orco con bastone) e l'azione sarebbe comunque eseguita.

Se un oggetto ha un nome che include uno o due aggettivi, è probabilmente possibile riferirsi ad esso semplicemente con un aggettivo. Se l'oggetto nel gioco è chiamato “the tarnished gold crown” (la corona d'oro intarsiata), scrivendo PICK UP GOLD (prendi oro) dovrebbe funzionare (o almeno dovrebbe funzionare se l'autore ha svolto un lavoro accettabile nello scrivere il gioco).

Riguarda te!

In molte *interactive fiction*, tu (il giocatore) reciti la parte del personaggio “you” (in seconda persona). Quando il gioco stampa un messaggio, come “You can't see any such thing” (non puoi vedere nulla del genere), sta parlando del personaggio chiamato “you”. Il pronome “you” è grammaticalmente in seconda persona, così possiamo dire che questi giochi sono scritti in *seconda persona*. Solo pochi giochi sono scritti in modo che il personaggio sia “I” (in prima persona) o “he”/“she” (in terza persona).

In molti giochi, l'azione è descritta come se stesse accadendo in questo preciso momento. Per esempio, “You walk up the stairs” (sali le scale). Grammaticalmente, questo viene chiamato *tempo presente*. Solo pochi giochi sono scritti al passato, come in “You walked up the stairs” (hai salito le scale). Il racconto in seconda persona e al tempo presente non è molto usato in racconti o romanzi, ma è quasi la regola nell'*interactive fiction*. E' possibile scrivere una storia con qualsiasi combinazione di persone e tempi, ma le tecniche per farlo vanno oltre gli scopi di questa *Guida*.

Ogni tanto il parser chiederà più informazioni. Per esempio, se sei in una stanza con un corona d'oro, un anello d'oro e un globo d'oro, quando scriverai PICK UP GOLD (prendi l'oro) il parser risponderà, “Which do you mean, the gold crown, the gold ring, or the gold orb?” (“Cosa intendi, la corona d'oro, l'anello d'oro o il globo d'oro?”). A questo punto il parser cercherà di interpretare l'input come una risposta. Se scriverai semplicemente ORB (globo), il parser comprenderà che vuoi dire PICK UP GOLD ORB (prendi il globo d'oro) ed il gioco procederà eseguendo quell'azione.

Se sei in una stanza con un altro personaggio, puoi tentare di impartirgli ordini, come questo: BOB, PICK UP THE STICK (Bob, prendi il bastone). Bob potrebbe o meno eseguire l'ordine — in questo momento, stiamo solo considerando come vengono inseriti i comandi. In questo caso, scrivi il nome del personaggio, poi una virgola ed infine il comando.

Molte storie interattive hanno luogo in quello che si può chiamare il mondo virtuale (*model world*). Questo è un luogo che è stato creato dall'autore della storia. Esiste solo all'interno del computer — in effetti, solo all'interno dell'interprete software del computer. Può trattarsi di un antico castello, l'interno di una stazione spaziale o un'affollata città moderna. Comunque il mondo virtuale è sempre costituito da *stanze*. Le stanze sono i luoghi dove avviene l'azione. Una “stanza” può essere un ampio spazio aperto o potrebbe essere l'interno di una cabina telefonica. La parola “stanza” è usata nella scrittura di IF per riferirsi ad una qualsiasi locazione del gioco, non importa se è letteralmente una stanza o un luogo differente.

Se vuoi vedere cosa c'è nella stanza in cui ti trovi, puoi usare il comando LOOK che può essere abbreviato con L (GUARDA abbreviato con G). Quando usi LOOK, leggerai la descrizione della stanza e ciò che contiene. In genere, puoi solo vedere ciò che c'è nella stanza in cui ti trovi; non puoi vedere all'interno di un'altra stanza. Alcuni giochi hanno finestre attraverso le quali si può vedere l'esterno, ma solitamente vengono realizzate sfruttando qualche trucco di programmazione.

Se ci sono più stanze nel gioco, sarà possibile passare da una all'altra. Questo viene effettuato solitamente impartendo dei comandi basati sulle direzioni della bussola — per esempio, GO NORTH (VAI A NORD). Questo tipo di comando è così comune che può essere abbreviato. Puoi scrivere N per andare a nord, NE per nordest, E per est, SE per sudest, e così via. A seconda del tipo di mondo in cui la storia si svolge, le direzioni possono includere anche UP, abbreviato con U (SU), oppure DOWN, abbreviato con D (GIU'), IN (DENTRO) e OUT (FUORI).

Viaggiando attraverso il mondo virtuale, incontrerai vari tipi di oggetti. Alcuni di essi possono essere trasportati: potrai raccoglierti, portarli con te e lasciarli in altri luoghi. Altri oggetti fanno parte dello scenario e non possono essere trasportati.

La prima cosa che potresti fare, quando entri in una stanza, è usare in comando EXAMINE (ESAMINA) su tutti gli oggetti che sono menzionati nella descrizione della stanza stessa. Molti giochi moderni interpretano X come un'abbreviazione di EXAMINE, ma alcuni giochi della vecchia scuola non lo fanno. Leggi attentamente la descrizione della stanza e poi usa X su tutto ciò che è menzionato. Con questa operazione puoi scoprire dettagli interessanti.

Se un oggetto si può trasportare, potrai raccogliertelo. Supponiamo che l'oggetto sia una palla da bowling. I comandi PICK UP BALL, TAKE BALL e GET BALL significano tutti la stessa cosa (prendi la palla). (I programmatori di Inform chiamano *taking* questo tipo di azione). Per leggere la lista degli oggetti che stai trasportando, usa il comando INVENTORY (INVENTARIO). Può essere abbreviato con INV o semplicemente I.

Di tanto in tanto troverai dei contenitori. Un contenitore può essere sempre aperto, come un cestino, o si può aprire (OPEN) e chiudere (CLOSE), come una valigia. Se un contenitore è aperto, puoi provare a inserirgli dentro degli oggetti con comandi come PUT BOWLING BALL IN THIMBLE (metti la palla da bowling nel ditale). Alcuni oggetti, che possono essere aperti o chiusi, possono anche essere bloccati/chiusi a chiave (LOCK) o sbloccati/aperti con una chiave (UNLOCK).

[*N.d.T.*] A questo proposito è bene segnalare una problema che si riscontra usando la lingua italiana. Il comando APRI LA PORTA, viene interpretato come OPEN THE DOOR, e quindi se la porta è chiusa a chiave, non si apre anche se il giocatore possiede la chiave. In inglese può a volte bastare UNLOCK THE DOOR, mentre in italiano si deve essere più precisi: APRI LA PORTA CON LA CHIAVE. Lo stesso problema si presenta con CHIUDI che viene usato sia per CLOSE che per LOCK. Per distinguere i due usi occorre specificare, nel secondo caso, la chiave che si vuole utilizzare.

Molti giochi includono una serie di puzzle (vedi "Tutto sui puzzle" più avanti in questo capitolo). Alcuni puzzle sono facili, mentre altri possono essere particolarmente difficili. Al giorno d'oggi, molti giochi contengono dei suggerimenti che ti possono aiutare se non sai come risolvere un puzzle, ma in altri, puoi capire come procedere solo con la tua intelligenza.

Alcuni giochi non sono aggressivi verso il giocatore: puoi rimanere bloccato per un po' prima di capire che cosa devi fare, ma male che vada il personaggio che interpreti può vagare senza meta (o rimanere bloccato in una stanza chiusa a chiave) senza sapere cosa fare. Altri giochi invece sono particolarmente crudeli. In un gioco di quest'ultimo tipo, se fai la mossa sbagliata, il tuo personaggio può morire, forse in un modo molto spiacevole. Fortunatamente, il comando UNDO (DISFA) solitamente ti tira fuori dai guai, ma non sempre purtroppo. Alcuni giochi sono così crudeli che non permettono di usare UNDO se il tuo personaggio è morto. Per questa ragione, è una buona idea utilizzare il comando SAVE (SALVA) per memorizzare di tanto in tanto lo stato del gioco, specialmente prima di tentare qualcosa che potrebbe essere pericoloso. Se rimani ucciso o perdi il gioco in qualche altro modo, potrai usare RESTORE (RIPRISTINA). Il comando RESTORE apre una finestra di dialogo dove è possibile scegliere il salvataggio da ricaricare. Selezionando quello più recente, puoi tornare al punto in cui eri giunto prima di finire nei guai (non sarebbe bello se la vita reale funzionasse in questo modo?).

Giochi da scaricare

Uno dei modi migliori per imparare a creare un gioco è quello di scaricarlo e giocarlo alcuni. Un buon posto per iniziare a cercare è l'Interactive Fiction Database (<http://ifdb.tads.org/>). Su questo sito puoi leggere la recensione dei giochi, cercare per autore e scaricare i giochi stessi.

Altre risorse per cercare giochi includono Baf's Guide (<http://www.wurb.com/if/>) e una rivista online chiamata SPAG (Society for the Promotion of Adventure Gaming, <http://www.sparkynet.com/spag/>). Alcuni giochi sono disponibili sui siti Web degli autori, ma in numero ancora maggiore possono essere trovati nell'Interactive Fiction Archive (<http://www.ifarchive.org/>).

Per giocare avventure testuali, hai bisogno del gioco stesso e di un programma separato chiamato **interprete**. Dopo aver scaricato ed installato un interprete, dovrai caricare il gioco nell'interprete (alcuni giochi scritti con TADS ed altri strumenti di sviluppo sono programmi a sé stanti, ma solitamente un interprete è necessario).

Attualmente, su Mac OS X il miglior interprete è Zoom¹⁰. Zoom può eseguire tutti i giochi Z-code (Inform) e Glulx, così come i giochi TADS e quelli Hugo.

Su Windows, può essere necessario usare due differenti interpreti, a seconda del tipo di gioco che si vuole eseguire. I giochi scritti con Inform e compilati nel formato Glulx (il nome del file termina con estensione .blb o .ulx) possono essere eseguiti da un interprete Glulx. I giochi scritti con Inform e compilati nel formato .z5 o .z8 necessitano di un interprete Z-code di cui sono disponibili vari tipi.

Per avere maggiori informazioni su quali interpreti usare e dove scaricarli, un buon sito è il *Brass Lantern* di Stephen Granade (<http://brasslantern.org/beginners/tadownload.html>).

Ad ogni modo, cosa sono questi .z5 e .z8?

All'inizio c'era Zork. Beh, non era proprio l'inizio. All'inizio c'era Adventure. Adventure fu la prima vera avventura testuale. Era copiata e liberamente distribuita tra i possessori di computer negli anni '70 e non è mai diventato un prodotto commerciale. Ma intorno al 1980, alcuni capirono che potevano fare soldi con le avventure testuali. Fondarono una società chiamata Infocom. Il primo gioco prodotto da Infocom si chiamava Zork.

Zork ebbe molto successo e generò una serie di seguiti, che ebbero altrettanto successo. Ma poi, nella metà degli anni '80, i computer divennero sufficientemente veloci per gestire la grafica. I giochi grafici erano molto più attraenti di quelli basati sul testo. I giochi testuali persero di interesse commerciale.

Ma a quella data, comunque, c'erano centinaia di migliaia di computer che avevano installato Zork o uno degli altri giochi Infocom. Questi erano disponibili per molti sistemi operativi diversi (e a quel

¹⁰ <http://www.logicalshift.co.uk/unix/zoom/>

tempo c'erano molti più sistemi operativi di oggi). I dati del gioco erano gli stessi, non importava quale fosse il computer posseduto, perché Infocom aveva creato una *macchina virtuale* per ciascun sistema operativo. Una macchina virtuale... beh, non preoccuparti dei dettagli tecnici. Il punto è che, se possiedi una macchina virtuale prodotta da Infocom per il tuo computer, puoi giocare qualsiasi gioco Infocom.

La macchina virtuale sviluppata da Infocom è chiamata Z-machine (come vedi, prendi il nome da Zork). Perfino dopo la chiusura di Infocom, la Z-machine era ancora largamente diffusa.

Così quando Graham Nelson iniziò a lavorare sulla prima versione di Inform nei primi anni '90, ebbe una felice intuizione: non avrebbe cercato di sviluppare da zero il suo sistema di gioco. Invece, avrebbe creato un linguaggio di programmazione il cui compilatore poteva produrre file in un formato adatto per essere caricato dalla Z-machine. I giochi così compilati possono ancora essere eseguiti da chiunque possieda una Z-machine sul disco rigido. Questo è stato uno dei fattori che hanno assicurato il successo di Inform.

I computer a quei tempi avevano molta meno memoria in confronto a quelli odierni, così la Z-machine doveva essere piccola ed efficiente. Poteva caricare ed eseguire diversi tipi di file, ma i più comuni avevano nomi che terminavano con .z5 e .z8. Un file .z5 può occupare 256Kb (sì, sono kilobytes, non megabytes), mentre il più esigente formato .z8 può essere usato per giochi che necessitano di 512Kb.

Al giorno d'oggi, l'era della Z-machine di Infocom è storia antica. Se possiedi ancora un computer Atari o un Kaypro funzionante e un interprete per la Z-machine di quel computer, puoi eseguire un gioco scritto oggi con Inform, se è stato compilato come file .z5 o .z8. Ma chi possiede più un Kaypro? Oggi, sono stati scritti altri nuovi motori compatibili con la Z-machine. Hanno nomi come Frotz e Nitfol (entrambe erano parole magiche in uno dei seguiti di Zork).

Per fare in modo che gli autori Inform possano scrivere giochi di dimensioni ancora maggiori, Andrew Plotkin ha creato il formato Glulx. I giochi Glulx possono essere molto grandi e includere suoni, grafica e sotto-finestre multiple all'interno della stessa finestra di gioco. Inoltre possono essere giocati su qualunque computer moderno, poiché esistono interpreti Glulx per Macintosh, Windows e Linux. Infine, sono solitamente troppo grandi per essere giocati su telefoni cellulari e computer palmari, mentre sono invece disponibili interpreti Z-machine per alcuni palmari.

Scrivere il primo gioco

Non ci sono regole assolute che possano stabilire cosa deve comparire in un'avventura testuale. Una delle cose migliori riguardo all'*interactive fiction* è che si è liberi di scrivere qualunque cosa si voglia ed inserirci qualunque elemento che possa essere divertente.

Ma se speri di creare un gioco che possa essere distribuito ad altre persone, devi pensare attentamente a ciò che la gente potrà vedere o fare quando esegue il tuo gioco. Qui seguono alcuni suggerimenti su come produrre un gioco efficace. Se questi suggerimenti non sono comprensibili ora, torna a leggere la lista tra qualche settimana — può avere più senso dopo un po' di pratica di scrittura.

- 1) I giocatori non ti leggono nel pensiero. Se vuoi fargli sapere qualcosa (come le direzioni in cui possono muoversi quando si trovano in una certa stanza, o il fatto che una bomba a tempo sta effettuando il suo conto alla rovescia), devi scrivere un testo di output che li informi.
- 2) Quando crei degli oggetti, aggiungi sempre tutte le parole del vocabolario che il giocatore potrebbe usare per riferirsi a quegli oggetti.
- 3) Quando crei nuove azioni per agire sugli oggetti, aggiungi sempre i sinonimi che ti vengono in mente. Forzare il giocatore ad indovinare il termine che hai in mente è considerato un indice di scarsa qualità.
- 4) Dopo aver scritto alcuni paragrafi, premi il pulsante Go! per compilare il gioco e verificarne il funzionamento. Prova tutti i comandi stupidi che ti vengono in mente ed osserva come quei comandi vengono gestiti. Poi riscrivi e verifica ancora.
- 5) Scrivi un'introduzione al gioco, utilizzando una regola “When Play Begins” (quando il gioco inizia) — vedi il Capitolo 7, che fornisce al giocatore una chiara idea su un paio di cose: dove si svolge la storia, che personaggio interpreta e cosa, quel personaggio, sta cercando di fare.

Inform contiene un gran numero di regole che permettono al gioco di eseguire una serie di azioni automaticamente. Il gioco può, ad esempio, costruire frasi che non hai mai programmato, per descrivere situazioni che si possono verificare durante lo svolgimento della storia. Le regole che sono incluse in I7 sono chiamate la *libreria standard* e le *regole standard*. Puoi usarle senza sapere nulla del loro funzionamento. Infatti, le userai. Ogni regola della libreria può essere individualmente disabilitata se produce risultati che non desideri. Disabilitare porzioni della libreria è, comunque, un argomento avanzato; non ne parleremo molto in questo libro.

Dalla cima

Per iniziare un gioco, devi creare almeno una stanza, dove la storia comincerà. Il Capitolo 2 di questo libro tratta di come creare le stanze. Tuttavia, prima di scrivere la stanza iniziale, devi impostare alcune cose preliminari. Raccomando di iniziare il gioco come descritto qui nel seguito. Prima, occorre assegnare al gioco un titolo e specificare il nome dell'autore (questi aspetti sono gestiti automaticamente da Inform all'inizio di un nuovo progetto, ma possono essere cambiati in ogni momento con la modifica della prima riga nella pagina del sorgente). Dopo il titolo, occorre saltare una linea e scrivere una frase “Use”, come questa:

[Use American dialect, full-length room descriptions, and the serial comma.](#)

[\[In italiano: Include Italian by Leonardo Boselli. Use full-length room descriptions.\]](#)

[*N.d.T.*] Il primo “include” nella versione italiana richiama l'estensione che gestisce l'interfaccia utente in italiano. Per installare l'estensione consulta la “Prefazione alla traduzione”. Degli altri “Use”, l'unico significativo per un autore italiano è il “full-length room descriptions”, spiegato più avanti.

Inform è stato scritto da un Inglese, così usa i termini formulati in inglese britannico (invece che americano) a meno che non sia specificato diversamente. “Use American dialect” cambia lo spelling da britannico ad americano.

Quando si utilizza la descrizione delle stanze a piena lunghezza (full-length), il giocatore leggerà la descrizione completa di ogni stanza ogni volta che entrerà in una di esse. Se non si usa l'opzione full-length, la descrizione della stanza sarà stampata completamente solo al primo accesso; dopodiché, solo il nome della stanza sarà stampato (in realtà, ogni oggetto che è stato spostato o eventuali persone nella stanza saranno comunque menzionati). Il giocatore può impostare le descrizioni complete o abbreviate utilizzando i comandi VERBOSE (prolisso) e BRIEF (breve), che sono già definiti in Inform. In questo modo, specificare a Inform l'utilizzare delle descrizioni full-length riguarda solo l'impostazione iniziale e non pregiudica eventuali cambiamenti. Fai attenzione: se la soluzione di un puzzle è possibile solo quando il gioco è nella modalità prolissa, ma il giocatore ha impostato quella breve, hai scritto un puzzle sleale!

La *serial comma* che compare nell'”Use” è la virgola finale nelle elencazioni. Se è impostata, il tuo gioco scriverà:

```
You can see an apple, a pear, and a banana here.
```

(Qui puoi vedere una mela, una pera e una banana)

Se non è utilizzata, l'output è differente:

```
You can see an apple, a pear and a banana here.
```

(è inutile far notare che la modalità standard è quella necessaria in italiano [N.d.T.]

Molti giochi necessitano anche di un'introduzione — testo che “imposterà la situazione” della storia. Puoi creare una introduzione scrivendo una regola “When Play Begins” (quando il gioco inizia), forse qualcosa del genere:

```
When play begins, say "Lord Triffid has invited you to spend your week's vacation in his castle. Upon arriving, though, you begin to feel a bit uneasy. Perhaps it's the bats flying in and out of the attic window that put a damper on your mood, or perhaps it's the sound of barking, snarling guard dogs...."
```

[In italiano: When play begins, say “Lord Triffid ti ha invitato a passare un weekend nel suo castello. All'arrivo, comunque, ti senti un po['] a disagio. Forse ti senti depresso a causa dei pipistrelli che volano avanti e indietro dalla finestra dell'attico, oppure sei infastidito dal chiasso dei cani da guardia che abbaiano e ringhiano...”] (Nota l'apostrofo tra quadre: altrimenti diventerebbe una virgoletta [N.d.T.]

Cos'è un WIP?

Se leggi le discussioni *sull'interactive fiction* sui forum o i newsgroup, sentirai spesso parlare di “WIP”. Questa è un'abbreviazione per “work in progress” (lavoro in corso). Giochi complessi possono essere “in corso” per mesi o anche per più tempo. Finché stai lavorando su un gioco, o persino solo pensandoci di tanto in tanto, è un WIP.

Raccontare una storia

Forse il modo più semplice per scrivere una storia — qualsiasi storia, sia interattiva che scritta sulla carta o recitata — è quella di pensare che la storia riguarda una persona che ha un *problema*. La ragione per impostare le storie in questo modo è semplice: se il personaggio principale non ha un problema, la storia sarà certamente noiosa. Quando leggiamo una storia, vogliamo goderci la suspense di chiederci se il personaggio principale risolverà il problema ed alla fine vogliamo la soddisfazione di vedere come il problema è stato risolto. Nell'*interactive fiction* il giocatore è il personaggio principale e ciò aumenta la suspense.

Il problema della storia può essere banale come ritrovare un gattino perduto, o complicato come salvare la terra da una invasione aliena. Finché il problema è emozionalmente coinvolgente per il personaggio principale, funzionerà egregiamente nella storia.

Se il problema è troppo semplice, il lettore (o il giocatore) penserà: “com'è stupido”. Così l'autore deve anche preoccuparsi che il problema non sia troppo semplice da risolvere per mantenere vivo l'interesse dei lettori.

Nell'*interactive fiction* sviluppare la storia principale significa spesso risolvere una serie di puzzle. Più avanti, nel paragrafo “Tutto sui puzzle” di questo stesso capitolo, spiegheremo nel dettaglio come si creano i puzzle.

Gestire i progetti

E' una buona idea salvare il progetto sempre in una cartella specifica del disco. In Windows, per esempio, questa è probabilmente la cartella: Documenti > Inform > Projects. Se hai cura del tuo lavoro (e dovresti!), è davvero una buona idea creare un *backup* del tuo progetto in una locazione separata ogni volta che lo modifichi. Le memorie USB sono poco costose e comode da utilizzare. Usa sempre una locazione separata *fisicamente* per il backup, non solo una differente partizione dello stesso disco rigido. E' opportuno avere una copia di backup per proteggersi contro la possibilità di un malfunzionamento del disco rigido.

Ogni tanto, salvo il progetto in un nuovo file numerato. Dopo aver lavorato sulla versione 05 per alcuni giorni, sono solito utilizzare il comando “Save As...” (salva come) per memorizzare il progetto come versione 06 e così via (la release finale del mio progetto “A Flustered Duck” è la 21). Ecco perché si tratta di una buona idea: se ti penti di un cambiamento nel design effettuato ad un certo punto, o se cancelli inavvertitamente qualcosa, puoi aprire una vecchia versione del progetto e copiare una porzione di codice sorgente dalla vecchia versione alla nuova.

Questo accorgimento mi ha salvato da seri problemi anche mentre stavo scrivendo questa *Guida*. Lo raccomando caldamente.

Tutto sui Bug



Quando un programma per computer non fa ciò che dovrebbe fare, cioè fa la cosa sbagliata o, peggio, non funziona proprio, siamo di fronte ad un *bug*. Secondo una leggenda, un giorno dei programmatori che stavano cercando di usare uno dei primi computer cominciarono ad ottenere errori molto strani. Dopo ore frustranti, qualcuno pensò di aprire la scatola dei circuiti per ispezionarla. Una falena morta era stata risucchiata in uno dei tubi a vuoto. Dopo quell'evento, ogni comportamento strano di un programmi viene detto “bug”.

Trovare e correggere i bug impegna gran parte del tempo nella programmazione dei computer. Fortunatamente, molti dei bug che incontrerai non avranno a che fare con insetti morti! (sebbene questo possa costituire un interessante puzzle). Molti bug odierni sono causati da errori nelle istruzioni del codice sorgente che il computer sta cercando di eseguire.

Tutti scriviamo codice contenente bug, così devi cercare di abituartici. Molti bug si trovano e si correggono facilmente. Alcuni di essi ti possono far impazzire per ore. Fa parte della programmazione.

I bug di Inform si possono presentare in varie forme.

Quando premi Go! per ordinare a Inform di compilare il codice sorgente, il compilatore può trovare vari errori. Puoi aver scritto cose che Inform non capisce. Invece di produrre un gioco, Inform stamperà un messaggio di errore, o anche un'intera serie. Solitamente questi messaggi ti daranno un'idea precisa sul tipo di correzione — ma talvolta il compilatore non riesce ad indovinare quale sia il problema.

Nel secondo tipo di bug, quando premi il pulsante Go!, il tuo gioco si compilerà con successo ed appare nella finestra di destra, pronto per essere provato. Il gioco però può non comportarsi nel modo che ti saresti aspettato. Ad esempio, un oggetto che volevi mettere bene in vista in una stanza potrebbe non comparire in nessun luogo del gioco. Inform non può trovare questi bug. L'unico modo per individuarli è quello di verificare ripetutamente il gioco mentre è in fase di scrittura. Prova un'intera serie di comandi, inclusi quelli stupidi, per vedere quello che accade (questo si dice “cercare di romperlo”).

Il terzo tipo è chiamato “errore di run-time”. In un errore di questo tipo, il gioco cerca di fare qualcosa che non può fare. Invece di produrre l'output normale, il gioco genererà un messaggio di errore. Gli errori di run-time possono avvenire, per esempio, quando il tuo codice cerca di riferirsi ad un oggetto che non esiste. Quando incorri in un errore di run-time, leggi attentamente il messaggio di errore nel pannello del gioco e poi il codice che viene usato dal comando che ha prodotto l'errore.

Gli errori di run-time più gravi sono quelli che causano l'interruzione o il blocco del programma. Fortunatamente, questi ultimi sono piuttosto rari. Però non devi sorprenderti se avvengono — semplicemente rileggi il codice che è stato invocato dopo il tuo ultimo comando. Nove volte su dieci, il problema sarà proprio in quel punto.

Verificare il gioco

La verifica di un gioco, o di qualsiasi altro programma, avviene in due fasi. Nella prima fase (alpha), verifichi il gioco mentre lo sviluppi. Ma anche se lo verifichi nel modo più attento possibile, ti sfuggiranno dozzine di terribili bug. E' un fatto.

Nella seconda fase (beta), hai bisogno dell'aiuto di alcuni volontari, che eseguono il *beta-test* del tuo gioco prima che sia rilasciato e ti inviano un rapporto su ogni bug che incontrano. Un buon beta-test non trasforma un gioco terribile in uno buono, ma può mutare un gioco terribile in uno accettabile, o trasformare un gioco decente con grossi problemi in un grande gioco.

Ringrazia sempre i tuoi beta-tester per averti inviato i loro rapporti sui bug!

Se hai problemi nell'individuare un bug, un'utile tecnica è quella di commentare (“comment out”) una sezione del tuo codice sorgente che pensi possa causare il problema. In Inform, i commenti sono racchiusi tra parentesi quadre [come queste]. Il testo racchiuso tra di esse sarà ignorato dal compilatore — a meno che non sia racchiuso tra virgolette. Tra virgolette, le parentesi quadre hanno uno scopo differente che sarà trattato in altri capitoli.

Commentando blocchi di codice, puoi verificare il gioco con o senza determinate funzionalità. Tutto ciò aiuta ad individuare il punto in cui si presenta il problema.

Il quarto tipo di bug può sembrare simile altri tre quando viene segnalato. Poiché lo stesso Inform è ancora in fase di sviluppo, il compilatore ha indubbiamente alcuni bug in se stesso. Se incontri qualcosa di strano quando ti sembra che tutto sia corretto, potrebbe trattarsi di un bug del compilatore. Il 90% delle volte non lo sarà — sarà invece il tuo codice. Ma i bug del compilatore esistono. Se incontrerai quello che apparirà come un bug del compilatore, il primo passo è quello di inviare un messaggio al newsgroup rec.arts.int-fiction (vedi pag. 37) per chiedere una conferma di ciò che hai individuato. Se è veramente un bug del compilatore, manderai un rapporto utilizzando la maschera fornita sul sito di Inform (vai all'indirizzo <http://inform7.com/contribute/report/>).

Utilizzare i comandi di Debug

Inform possiede un ottimo insieme di comandi che possono velocizzare il processo di verifica e di individuazione dei bug nel tuo gioco. Questi comandi sono disponibili solo durante lo sviluppo; quando pubblichi il gioco finito, gli altri giocatori non saranno in grado di utilizzarli. Questi comandi non individuano realmente i bug; quello che fanno è aiutarti a vedere ciò che sta avvenendo nel tuo gioco mentre viene eseguito e/o a prendere scorciatoie che rendono più rapido il processo di verifica.

I comandi che puoi utilizzare mentre il tuo gioco è eseguito in Inform includono ACTIONS, RULES, PURLOIN, GONEAR e SHOWME (azioni, regole, ruba, vai verso e mostrami). Inoltre, per verificare velocemente una serie di comandi, puoi scrivere uno script TEST ME (verificami). Il pulsante Replay può essere utilizzato per eseguire nuovamente la sequenza di comandi più recente, che è molto utile per verificare se sei riuscito a risolvere un problema che si era presentato nell'ultima esecuzione. La finestra Skein contiene tutte le tue ultime serie di comandi e puoi sfruttarla per eseguire nuovamente ogni serie che hai utilizzato in precedenza.

Con il comando PURLOIN (ruba), puoi “afferrare” istantaneamente qualsiasi oggetto che si trova nel mondo virtuale. Oggetti in stanze lontane o in contenitori chiusi possono essere rubati — e possono essere rubati oggetti che non si possono prendere ordinariamente, come persone o parti di scenario. Non c'è alcuna motivo che giustifichi il “furto” di persone o parti di scenario, se non per divertimento. Però, “sgraffignando” la chiave di ferro arrugginito che è nascosta nel barile di quercia all'interno della cantina si può verificare velocemente se la chiave funziona per aprire la porta della stanza della principessa. Per farlo, non hai bisogno di scendere in cantina e rompere il barile; semplicemente usa il comando PURLOIN con la chiave.

Il comando GONEAR (vai verso) permette di essere trasportati istantaneamente in ogni stanza del gioco. Con un gioco molto ampio, possono esserci dozzine di stanze, e GONEAR risparmierà ore di lavoro. Semplicemente scegli una parte dello scenario della stanza in cui vuoi essere trasportato e usa GONEAR con quell'oggetto.

Il comando ACTIONS (azioni) fa in modo che Inform stampi ciò che viene eseguito quando scrivi quel comando. Può essere utile se non ottieni da un comando ciò che ti aspetti.

Il comando SHOWME (vedi il **par. 2.7** della Documentazione, “The SHOWME command”) stampa lo stato corrente di ogni oggetto nel mondo virtuale. SHOWME SHOES (mostrami le scarpe), per esempio, stamperà tutti i dati associati all'oggetto scarpe. E' utile quando hai a che fare con oggetti che possono trovarsi in una serie di stati differenti (per esempio, una coppa di vino che può essere avvelenata o sicura), e puoi usare il comando SHOWME GOBLET (mostrami la coppa) dopo aver gettato le pillole nella coppa per vedere se quest'ultima ha veramente cambiato il suo stato come si suppone debba fare.

Se stai utilizzando le “scene” nel tuo gioco (vedi Capitolo 7), il comando SCENES (scene) può essere usato per mostrare quale scena è attualmente in corso.

Il comando TEST ME (vedi il **par. 2.8** della Documentazione, “The TEST command”) fornisce un modo rapido per eseguire una lunga serie di comandi. Per iniziare, definisci la serie di comandi che vuoi utilizzare nel gioco, come segue:

[Test me with "open suitcase / unlock suitcase with rusty key / open suitcase / search suitcase / take dynamite / light match / light dynamite with match / z / z / z".](#)

Dopo aver compilato il gioco, quando scrivi TEST ME il gioco eseguirà l'intera sequenza di comandi, in ordine. Se verifichi più volte puzzle complessi, questo risparmierà molto lavoro di scrittura. La

parola “me”, qui usata spesso, non è necessaria. Puoi procedere anche in questo modo, scrivendo test separati e poi, se vuoi, scrivere un comando TEST ME che comprenda gli altri, come viene mostrato da queste righe:

Test suitcase with "open suitcase / unlock suitcase with rusty key / open suitcase / search suitcase".
Test dynamite with "take dynamite / light match / light dynamite with match / z / z / z".
Test me with "test suitcase / test dynamite".

Ora i comandi TEST SUITCASE (verifica la valigia) e TEST DYNAMITE (verifica la dinamite) possono essere usati indipendentemente, o possono essere combinati mediante TEST ME. In ogni caso il risultato è lo stesso

Il comando RULES (regole) può essere usato per verificare quali regole sta usando Inform per eseguire altri comandi. Questa è una tecnica piuttosto avanzata — le prime volte che la provi potresti non capire quello che vedi — ma talvolta può essere molto utile. Se il gioco produce un output che non desideri, puoi riuscire a capire quale regola sta producendo quell'output utilizzando RULES e poi il comando che non è eseguito correttamente. Però il meccanismo di RULES non è perfetto: qualche volta il parser rifiuterà un input prima di considerare il libro di regole principale e quindi non si riuscirà a stabilire immediatamente quali sono le modifiche necessarie. Qui c'è un esempio:

```
>rules
Rules tracing now switched on. Type "rules off" to switch it off again, or
"rules all" to include even rules which do not apply.
```

(Il tracciamento delle regole è ora attivo. Scrivi “rules off” per disattivarle, o “rules all” per includere anche le regole che non vengono applicate)

```
>steve, take the crown
Steve has better things to do.
```

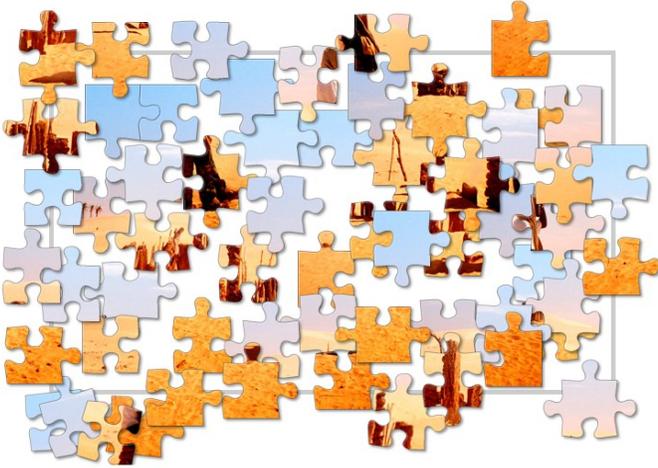
(Steve, prendi la corona. Steve ha cose migliori da fare.)

In questo caso, RULES non ti dice che regola causa il rifiuto del comando scritto del giocatore. Per risolvere il problema, ho cercato (usando il campo di ricerca) la frase “has better things to do” (ha migliori cose da fare) nella Documentazione. Questa ricerca mi ha condotto immediatamente al **par. 12.4**, “Persuasion” (persuasione), dove ho trovato la risposta.

Qui faccio un passo avanti e ti fornisco la risposta, anche se non la discuteremo fino al Capitolo 5. Ciò di cui abbiamo bisogno, se vogliamo cambiare ciò che avviene in un gioco, è una regola di persuasione (Persuasion Rule):

Persuasion rule for asking Steve to try taking the crown:
persuasion succeeds.

Tutto sui Puzzle



Siamo franchi: questo paragrafo non ti dirà tutto sui puzzle. Per un semplice motivo, gli autori sognano sempre nuove possibilità! Però i puzzle fanno così parte dell'*interactive fiction* che un libro su come scrivere IF utilizzando Inform non può ignorarli del tutto.

Sono stati scritti veramente pochi lavori di IF senza puzzle. In un lavoro di questo tipo, il giocatore vaga, cercando oggetti e/o conversando con personaggi, ma non ci sono ostacoli al movimento o all'azione. E' accessibile l'intera storia e non sono necessarie capacità speciali per risolvere i problemi. C'è

persino una competizione occasionale, l'IF Art Show, dedicata a questo tipo di giochi.

Anche se la storia non ha veri puzzle, il giocatore deve comunque operare delle scelte. Queste scelte possono influenzare l'esito della storia: l'autore può scrivere cinque o sei linee di diramazione della storia e il giocatore potrebbe dover giocare un certo numero di volte per essere sicuro di non perdere nulla di importante, o per capire cosa l'autore aveva in mente. Ma se la storia non ha puzzle, ci aspettiamo di trovare facilmente ognuna delle diverse scelte. Dovrebbe essere facile seguire ciascuna delle diramazioni della storia.

In molta IF, tuttavia, il giocatore deve spremere la sua materia grigia per trovare le azioni corrette che possano far procedere nella trama. Il giocatore che non intuisce cosa deve fare è *bloccato*. Un giocatore bloccato può vagare in tondo nel mondo virtuale della storia per un'ora, provando azioni che non producono alcun risultato e diventando sempre più frustrato.

Quando il giocatore capisce come risolvere un puzzle, c'è solitamente un premio di qualche genere. Un nuovo oggetto potrebbe diventare disponibile (qualcosa di cui il giocatore ha bisogno per risolvere un altro puzzle), un nuovo tesoro viene scoperto e si apre una nuova mappa o regione del mondo. In giochi che hanno un punteggio (vedi il Capitolo 6), il giocatore dovrebbe guadagnare punti dopo la risoluzione di ogni puzzle.

Probabilmente la forma base dei puzzle nell'IF è la porta chiusa. Il giocatore non possiede ancora la chiave che apre la porta e deve andare alla sua ricerca. Aprire una porta chiusa dà al giocatore l'accesso ad una nuova stanza, o forse ad un'intera regione del mondo virtuale, contenente dozzine di stanze.

Nel modo in cui io la vedo, ogni puzzle di ogni gioco, non importa che forma può prendere, non è essenzialmente differente da una porta chiusa. Non puoi vedere ciò che c'è dall'altra parte finché non trovi la chiave, qualunque cosa sia la chiave. La "chiave" può essere una mela che devi dare alla strega malvagia affinché si addormenti — in un certo senso, è ancora una chiave.

Le porte e le scatole chiuse a chiave si trovano in centinaia di giochi. Piuttosto spesso ce ne saranno diverse ed anche diverse chiavi. Questo può non essere molto divertente, perché il giocatore di solito ha la chiave sbagliata. Trovare un buon posto dove nascondere le chiavi può non essere facile. I giocatori cercheranno di usare i comandi LOOK BEHIND (guarda dietro) e LOOK UNDER (guarda sotto) con ogni oggetto (come un letto, un dipinto o un tappeto) che sembri possa nascondere un oggetto. Puoi anche aspettarti che i giocatori usino il comando SEARCH con ogni oggetto (come ad esempio una poltrona con cuscini) che sembri il posto ideale dove un oggetto possa essere nascosto. Per maggiori dettagli su come impostare la cosa, vedi il Capitolo 3, “Oggetti”.

Un giocatore d'esperienza saprà di dover usare il comando EXAMINE con ogni oggetto menzionato nella descrizione di una stanza. Così è corretto lasciare qualche indizio relativo ai vari puzzle nelle descrizioni. La descrizione di un pannello nel muro, ad esempio, può suggerire al giocatore che quel pannello è una porta segreta. O magari il pannello può sembrare completamente insignificante quando viene esaminato, ma se il giocatore esamina il tappeto scoprirà che un segno sul tappeto indica il pannello sul muro (se pensi di farlo, sii certo di menzionare il tappeto nella descrizione della stanza).

Le stanze oscure sono un tipo comune di puzzle (vedi il paragrafo sulle “Stanze oscure” nel Capitolo 2, pagina 69). Per risolvere il puzzle su una stanza oscura, il giocatore deve trovare e portare con sé un oggetto che faccia luce.

I congegni meccanici che devono essere accesi, o che hanno bisogno di energia per funzionare, sono altrettanto comuni. Un congegno meccanico può avere numerosi pulsanti e leve e può non essere evidente quali pulsanti o leve devono essere azionati. Può sembrare che non abbiano effetto nella stanza dove si trova il giocatore, ma potrebbero avere importanti conseguenze in una stanza differente. Oppure l'effetto dovuto alla pressione di un bottone o all'azionamento di una leva può essere ritardato di qualche turno.

Alcuni oggetti possono essere fuori portata: il giocatore può vederli, ma non riesce a raggiungerli. La soluzione può essere un lungo palo, o una sedia su cui salire, ad esempio.

Persone e animali possono svolgere la funzione di puzzle in vari modi. Guardie e cani da guardia, per esempio, possono essere usati per non permettere al giocatore di attraversare una porta invitante. La porta non è chiusa, ma è come se la fosse. Il giocatore deve trovare un modo per distrarre o diventare amico della guardia. Quasi ogni personaggio può comportarsi come una guardia — una solerte bibliotecaria, ad esempio, che non lascia entrare il giocatore nella sala del computer finché non le viene mostrata la tessera della biblioteca.

Al contrario, un personaggio può possedere un oggetto di cui il giocatore ha bisogno e il puzzle potrebbe consistere nel farglielo cedere. Alcuni personaggi possono eseguire azioni che il giocatore non può fare, in questo caso il puzzle consiste nell'ottenere dal personaggio l'esecuzione di un qualche tipo di azione. Il giocatore potrebbe non essere in grado di prendere un oggetto al fondo di una piscina, ad esempio, non sapendo nuotare — ma un amichevole delfino potrebbe voler obbedire agli ordini (come un delfino possa prendere qualcosa... lo lascio come esercizio al lettore).

Alcuni puzzle sono *temporizzati*. In un puzzle temporizzato, il giocatore deve impartire il comando giusto, o anche un'intera serie di comandi, in un numero fissato di turni. Un esempio estremo potrebbe essere un candelotto di dinamite la cui miccia è stata accesa: dopo cinque turni, la dinamite esplode ed il personaggio del giocatore morirà.

Agli inizi dell'IF, i labirinti erano comuni. Il primo vero gioco di IF, *Adventure*, aveva due labirinti. Un labirinto è un insieme di stanze (solitamente tra dieci e quindici) in cui è difficile spostarsi per qualche motivo. Tipicamente, tutte le stanze in un labirinto hanno descrizioni simili, rendendo difficile o impossibile dire in quale ci si ritrovi. L'idea alla base di un labirinto è questa: per qualche ragione il giocatore non può capire in quale direzione deve andare per uscire dall'altra parte. Le connessioni tra le stanze di un labirinto possono essere "intrecciate", il che significa che se vai a nord della stanza A ed arrivi nella stanza B, andare a sud della B molto probabilmente non ti porterà indietro alla stanza A!

Questo tipo di sistema di navigazione è semplice da realizzare in Inform (vedi "Connessioni serpeggianti" nel Capitolo 2). Di questi tempi, però, molti giocatori considerano i labirinti poco interessanti, se non noiosi.

Gli autori di IF hanno scritto molti saggi e suggerimenti sui puzzle. Ad esempio, Stephen Granade ha scritto un ottimo saggio breve sul suo sito Brass Lantern¹¹. Inoltre, il manuale di Graham Nelson su come programmare Inform 6, il *Designer Manual 4* (solitamente chiamato *DM4*), ha una sezione sulla progettazione di puzzle che si può trovare su inform-fiction.org¹².

Nello scrivere il tuo gioco, è vitale che tu pensi attentamente a come realizzare i puzzle.

Per prima cosa, hai fornito al giocatore sufficienti informazioni per permettergli di risolvere il puzzle? Ricorda: ciò che è ovvio per te (l'autore) potrebbe essere assolutamente oscuro per chi non riesce a leggerti nel pensiero.

In secondo luogo, hai considerato e programmato nel gioco tutti i comandi che il giocatore potrebbe usare nel risolvere il puzzle? Un giocatore che capisce il puzzle ma non capisce qual è la frase esatta che deve scrivere per risolverlo recenserà il gioco in modo molto negativo. Questo problema è chiamato *guess-the-verb* puzzle (indovina-il-verbo). Tutti odiano i puzzle del tipo indovina-il-verbo. Da evitare anche le risposte del parser che confondono. Qui segue un esempio:

```
>hit guard
Violence isn't the answer to this one.
(Colpisci la guardia. La violenza non è la risposta corretta.)
```

```
>hit guard with stick
You smack the guard in the head with the stick, and he goes down like a
sack of potatoes. Congratulations! Now you can steal the jewels from the
vault!
(Colpisci la guardia con il bastone. Colpisci la guardia sulla testa con il bastone e cade come un sacco...)
```

11 <http://brasslantern.org/writers/iftheory/betterpuzzles-a.html>

12 <http://www.inform-fiction.org/manual/html/s50.htm>

La risposta di base al comando HIT GUARD (colpisci la guardia) è completamente errata, perché la violenza è la risposta corretta in questo caso.

Mi piace includere alcuni puzzle semplici nei miei giochi ed alcuni difficili. Come giocatore, preferisco quelli facili, perché ho scarse capacità di problem-solving. Quando gioco le avventure degli altri, spesso rimango bloccato. Molti autori pensano che sia una buona idea aiutare i giocatori includendo qualche forma di sistema di suggerimenti nel gioco. Troverai alcune idee su come impostare un sistema di suggerimenti nel Capitolo 9. Altri autori pensano che farlo banalizzi il gioco o semplicemente che sia troppo difficile realizzarlo.

Estensioni per Inform

Molte persone della comunità di Inform 7 (che esiste solo su Internet) hanno scritto estensioni (*Extensions*) per Inform. Un'estensione è un file contenente codice che può essere usato per fare qualcosa di specifico — qualcosa che non è incluso nella versione standard di Inform.

Ci sono estensioni per creare particolari tipi di oggetti, come liquidi o porte segrete; per produrre conversazioni più realistiche ed interessanti tra il personaggio del giocatore (PG o PC in inglese) ed i personaggi non giocanti (PNG o NPC in inglese), per costruire menù di suggerimenti che i tuoi giocatori possono consultare e molti altri scopi.

Le estensioni sono scritte usando il linguaggio di programmazione di Inform — così tecnicamente, un'estensione non può fare ciò che non potresti fare tu stesso. Ma molte di esse sono state scritte da esperti, sono state verificate e private dei bug, e sono accompagnate da una documentazione che spiega come usarle. Quindi perché sprecare settimane per programmare qualcosa di complicato quando puoi scaricare un'estensione utilizzabile in un quarto d'ora?

Le estensioni possono essere scaricate dalla pagina delle estensioni del sito di Inform 7¹³. Prima di scaricare la tua prima estensione, dovresti creare una cartella chiamata Downloaded Extensions nella tua cartella Inform. Salva tutte le estensioni scaricate in quella cartella.

Importante: dopo aver scaricato l'estensione, devi installarla prima di poterla usare. Installarla fa capire al programma Inform che esiste. Per farlo, apri il menù File e seleziona il comando “Install Extension...”. Nella finestra di dialogo che si apre, seleziona l'estensione che hai scaricato e premi il pulsante Install. In questo modo l'estensione viene copiata in una cartella speciale e ce ne saranno due copie sul tuo disco rigido — una nella cartella dei download e una nella cartella delle estensioni. Se hai già installato altre estensioni dello stesso autore, tutte condivideranno la stessa sotto-cartella all'interno della cartella delle estensioni.

Se l'installazione è andata a buon fine, la sua documentazione sarà inclusa nella sezione documentazione del programma Inform. Per verificarlo, sfoglia l'indice della Documentazione, scorri la

¹³ <http://inform7.com/write/extensions/>

pagina verso il basso, se ce n'è bisogno, e clicca su “Installed Extensions” (estensioni installate). Vedrai che sono elencate alfabeticamente secondo il primo nome dell'autore.

Quando l'estensione è installata, usarla nel gioco è semplice. All'inizio del codice sorgente, proprio sotto il titolo, inserisci una riga come questa:

[Include Plurality by Emily Short.](#)

Questo è il formato richiesto: devi specificare il nome dell'autore e non devi mettere una virgola dopo il nome dell'estensione.

Se stai usando alcune estensioni, è bene visitare periodicamente il sito di Inform per verificare se sono state aggiornate. Gli aggiornamenti aggiungono nuove funzionalità e correggono eventuali bug.

Nota: dopo aver aperto un'estensione in una nuova finestra dell'IDE, puoi modificarne il codice proprio come faresti con qualunque altro codice Inform. *Ti raccomando caldamente* di non farlo. Fino a che non diventi un utente avanzato di Inform, è troppo facile creare problemi insolubili modificando un'estensione. Se vuoi provare a modificare un'estensione — per esempio, perché pensi che contenga un bug — fanne prima una copia di backup copiando il file e incollandolo in una nuova cartella del tuo disco rigido. In questo modo, se non ottieni i risultati che ti aspettavi, puoi sempre ripristinare la versione originale senza bisogno di scaricarla nuovamente.

Dove trovare ulteriori informazioni

Una volta che hai iniziato a scrivere con Inform, troverai la sua documentazione veramente utile. Alcune pagine che non avevano senso all'inizio diverranno, dopo un mese o due, veramente chiare e facili da capire. In particolare, i 400 e passa esempi sono una grande risorsa — ma per ricavarne il massimo, dovrai studiare il codice linea per linea. Le spiegazioni che sono incluse negli esempi sono spesso brevi e non sottolineano le parti più interessanti. Dopo aver letto varie sezioni della Documentazione, troverai il campo di ricerca dell'IDE sempre più utile. Scrivendo una frase ricordata a metà (some “something new”), puoi individuare velocemente la pagina che contiene le informazioni di cui hai bisogno.

Se — no, meglio dire “quando” — ti verranno domande su come usare Inform, troverai il newsgroup [rec.arts.int-fiction](#) una risorsa utilissima (il newsgroup è spesso chiamato dai frequentatori assidui *r.a.i-f*). Se chiederai qualcosa, probabilmente potrai ricevere una risposta da un esperto di Inform in poche ore — o male che vada, il mattino seguente.

Per ottenere il massimo dal newsgroup, alcune semplici regole possono aiutare:

- 1) Metti “I7” all'inizio della linea dell'oggetto del tuo messaggio. Inform 7 non è l'unico sistema di programmazione trattato nel newsgroup e vuoi attirare le persone giuste verso il tuo messaggio rendendo evidente quale sistema tu stia usando.

- 2) Descrivi il problema nel modo più chiaro possibile. Le domande prolisse e confuse produrranno meno risposte utili di quelle che sono chiare e concise.
- 3) Prima di inviare il messaggio, cerca di creare un semplice esempio del gioco (di non più di 20 righe) che esemplifichi il tuo problema. Questo renderà più semplice agli esperti vedere che cosa stai sbagliando — e nel processo di creare un gioco più semplice, puoi addirittura risolvere il problema da solo! (Spesso succede così, nella mia esperienza).

Il forum di IF (<http://www.intfiction.org/forum/>) ha meno traffico del newsgroup, ma gli esperti di Inform sembrano visitarlo regolarmente, perciò è un altro buon posto dove porre domande. Per avere informazioni generali su come scrivere IF, ifwiki (<http://www.ifwiki.org>) è un ottimo punto di partenza per iniziare la ricerca. Molte informazioni sono disponibili su Internet, alcune di esse nascoste in luoghi insospettabili, ma il wiki ha collegamenti per molti di essi. Anche le pagine sull'*interactive fiction* di Emily Short (<http://emshort.wordpress.com>) contengono molte utili risorse.

Che cos'è un newsgroup?

Agli inizi di Internet (perfino prima del World Wide Web), la gente era solita inviare messaggi in un'area di Internet chiamata Usenet. Usenet è ancora disponibile. Contiene decine di migliaia di newsgroup. Un newsgroup è una sorta di forum basato sul Web: tutti possono inviare messaggi, che sono letti da tutti gli altri.

Il tuo programma che gestisce le email può avere una pagina con cui è possibile accedere ai newsgroup. Comunque, puoi dover pagare un extra per poter accedere al server dei newsgroup. Alcuni ISP (Internet Service Provider) includono l'accesso ai newsgroup nel servizio di base, ma molti non lo fanno. Se vuoi iniziare ad usare i newsgroup, puoi consultare l'articolo su questo argomento su Wikipedia.

Puoi anche leggere i newsgroup per mezzo di Google Groups, utilizzando il tuo browser Web. L'accesso ai Google Groups è gratuito, ma l'interfaccia utente francamente non è molto buona.

Capitolo 2: Stanze e scenari



Ho già spiegato nel Capitolo 1 (vedi “Ciò che avviene in un gioco”) che il mondo virtuale in cui si svolge la storia è suddiviso nettamente in stanze. Per gli scopi dell'IF, una “stanza” potrebbe non essere una cucina o un soggiorno; può essere un campo da calcio o una cabina telefonica*. In entrambi i casi si usa lo stesso termine.

Mentre i personaggi (compreso il giocatore) viaggiano da stanza a stanza, per molti scopi possiamo pensarle come un contenitori sigillati. Quando il giocatore si trova in una stanza, potrà vedere, toccare e usare gli oggetti che si

trovano in quel luogo, mentre tutto ciò che è collocato in altre stanze sarà invisibile e fuori portata. Infatti, se il giocatore si riferisce a qualcosa che non si trova nell'attuale stanza, il parser (il software che interpreta ciò che viene scritto dal giocatore) affermerà che l'oggetto citato non esiste.

Questo è il comportamento che di solito è più realistico in un gioco, ma talvolta considerare una stanza come un contenitore sigillato non dà i risultati sperati. In questo caso, ci sono un paio di metodi per aggirare il comportamento di base. Se il giocatore si trova nella parte a nord di un campo aperto, ad esempio, si richiede che possa vedere (ma non toccare) gli oggetti che si trovano a sud di quello stesso campo. Più avanti in questo capitolo suggeriremo alcuni metodi per ottenere questo tipo di effetto. Ci sono anche alcune eccezioni che entrano in gioco quando si progetta un sistema di conversazione per permettere al giocatore di parlare ad altri personaggi: puoi facilmente consentire al comando ASK BOB ABOUT THE JEWELS (chiedi a Bob dei gioielli) di funzionare come ci si aspetterebbe anche se i gioielli non sono nella stanza. In molti casi, tuttavia, un oggetto si deve trovare nella stessa stanza del giocatore per fare in modo che possa essere utilizzato o perfino visto.

Questo concetto è basilare nell'*interactive fiction*. E' chiamato *scope* (contesto). Un oggetto è nel contesto se il personaggio del giocatore può vederlo; altrimenti non appartiene al contesto. Il parser si occupa di verificare se un oggetto è visibile o meno. Se il parser afferma di non sapere nulla di un determinato oggetto quando il giocatore si riferisce ad esso, significa che l'oggetto non è “in scope”. Questo accade, molto probabilmente, perché l'oggetto è in una stanza differente. Altri casi in cui può accadere sono: quando l'oggetto si trova in un contenitore chiuso nella stessa stanza, o quando non c'è abbastanza luce per vedere qualcosa.

* Una cabina telefonica può anche essere creata come un *enterable container* (un contenitore in cui si può entrare) all'interno della stanza principale — vedi il paragrafo su “Contenitori e supporti in cui si può entrare” nel Capitolo 3.

Creare la prima stanza

Ogni gioco Inform comprende almeno una stanza. Crearne una è facile; puoi fare così:

The Forest Path is a room. *[In italiano: The sentiero nella foresta is a room.]*

Questa semplice frase produce, in effetti, un intero gioco Inform (sebbene non molto interessante). Se crei un nuovo gioco, scrivi questa frase nel pannello Source e compila il gioco premendo il pulsante Go!. Ti troverai in una stanza (senza funzionalità) chiamata Forest Path (sentiero nella foresta).

Di solito è una buona idea iniziare le parole del nome della stanza con una maiuscola. Non è richiesto, ma è una buona abitudine perché produce un gioco dall'aspetto più professionale. Se vuoi, puoi aggiungere “The” prima del nome della stanza. Inform normalmente elimina il “The” prima di stampare il nome della stanza nell'output del gioco, così una stanza chiamata The Forest Path apparirà nel gioco come Forest Path. Puoi cambiare questo comportamento, comunque, se usi la proprietà **printed name** (nome stampato), in questo modo:

The printed name of the Forest Path is “The Forest Path”.

[In italiano: The printed name of the sentiero nella foresta is “Il sentiero nella foresta”.]

Nota la punteggiatura: il punto è *fuori* dalle virgolette (com'è ovvio per un italiano [*N.d.T.*]), non all'interno. Ciò è importante. Quando Inform vede un punto prima di una virgoletta chiusa, aggiunge una riga vuota extra nell'output del gioco ogni volta che scrive quel testo. Non vogliamo che accada con il nome della stanza (e specialmente non vogliamo che accada con alcuni nomi di stanze e non con altri, poiché renderebbe l'output un vero pasticcio). Mettere il punto fuori dalle virgolette è la cosa corretta da fare.

Quasi sempre vorrai fornire una **description** (descrizione) alla stanza, per fornire al giocatore qualche informazione su dove si trova. La descrizione consiste in un testo che deve essere stampato, così devi scriverlo tra virgolette, come in questo caso:

Forest Path is a room. “Tall old trees surround you.”

[In italiano: The sentiero nella foresta is a room. “Vecchi alberi ti circondano.”]

Come non descrivere una stanza

Scrivendo la descrizione di una stanza, ci sono due trappole che devi evitare. Prima di tutto, evita di iniziare con “Sei in...” o “Ti trovi in...”. Diventa subito molto noioso. Semplicemente descrivi ciò che si vede. In secondo luogo, evita di descrivere la stanza come se il giocatore fosse arrivato da o fosse rivolto in una particolare direzione. Iniziando a scrivere, puoi pensare che il giocatore sia arrivato nella sala da pranzo dall'ingresso — ma più avanti, potrebbe ritornare dalla cucina. Una buona descrizione deve dare la stessa impressione non importa come si sia arrivati nella stanza.

La descrizione di una stanza può avere la lunghezza che preferisci. Deve sempre terminare con un punto (o con un punto di domanda o un punto esclamativo, sebbene questi ultimi non siano molto comuni). In questo caso il punto deve essere all'*interno* delle virgolette, perché vogliamo che Inform inserisca una riga vuota dopo la descrizione e prima di qualsiasi testo sia stampato successivamente.

Nella descrizione, elencherai quasi sempre gli oggetti che possono essere visti in quella stanza. Potrebbero essere importanti per la storia o, semplicemente, aggiungere colore ed atmosfera ad essa. Gli oggetti che sono citati nella descrizione della stanza sono considerati *scenery* (scenari).

In generale, è una buona idea aggiungere un oggetto di tipo scenario (vedi il prossimo paragrafo) per ogni oggetto che è citato nella descrizione della stanza. All'inizio i giocatori non sanno ciò che è importante per la storia o ciò che è irrilevante, così inizieranno ad esaminare tutti gli oggetti che sono menzionati nell'output del gioco — o almeno cercheranno di farlo. Se viene esaminato qualcosa che non è presente nella stanza (*in scope*), il parser risponderà: “You can't see any such thing” (non puoi vedere nulla di simile). Se quell'oggetto è stato appena citato, quel messaggio è frustrante, e i giocatori decideranno presto che non vale la pena di continuare. Immagina una situazione del tipo:

Forest Path

```
Tall old trees surround you.
```

(Sentiero nella foresta. Ti circondano vecchi alberi.)

```
>x trees
```

```
You can't see any such thing.
```

(x alberi. Non puoi vedere nulla di simile.)

Nei primi tempi dell'*interactive fiction*, questo tipo di risposta era considerata normale, ma a quel tempo i computer avevano poca memoria. Gli autori di giochi dovevano fare del loro meglio con testi brevi ed un numero di oggetti ridotto, così non se ne poteva sprecare per descrivere nel dettaglio uno scenario irrilevante ai fini della trama. Oggi, non definire lo scenario della stanza è considerato indice di bassa qualità. Nel prossimo paragrafo, verrà spiegato come definire lo scenario.

Nelle descrizioni è importante fornire un'idea chiara delle direzioni in cui ci si può muovere. Cioè, se il giocatore può spostarsi ad est, nordest, o a ovest per raggiungere altre stanze, la descrizione della stanza deve citarle o almeno provvedere una vaga idea di ciò che si trova in quelle direzioni:

Forest Path

```
Tall old trees surround you. The path runs roughly north and south from here, and a little side path runs off through the bushes to the northeast.
```

(Sentiero nella foresta. Ti circondano vecchi alberi. Da qui, il sentiero si estende verso nord e verso sud, mentre una piccola strada laterale si dirama a nordest attraverso i cespugli.)

Agli albori, la mancata elencazione delle uscite nelle descrizioni era usato come puzzle, ma al giorno d'oggi questo genere di cose è considerato a livello troppo elementare e grezzo. Fortunatamente, ci sono estensioni (come “Exit Lister” di Eric Eve — in italiano, occorre usare “Exit Lister IT” di Leonardo Boselli [*N.d.T.*]) che inseriscono la lista delle uscite nella barra di stato in cima alla finestra di

gioco. L'utilizzo di “Exit Lister” o qualcosa di simile è un gentilezza bene accetta nei confronti del giocatore — ma è sempre bene elencare chiaramente le uscite anche nella descrizioni della stanza (a meno che una di esse non sia nascosta apposta perché costituisce la soluzione di un puzzle). “Exit Lister” fornisce anche un utile comando ai giocatori, EXITS (uscite), che elencherà le uscite disponibili per rinfrescare la memoria del giocatore.

Scenari

Inform permette di creare molti tipi (**kinds**) di oggetti. La libreria standard comprende contenitori, dispositivi meccanici, porte, persone e così via (vedi il **Capitolo 4** della Documentazione, “Kinds”, per saperne di più). La parola “kind” è un termine tecnico di Inform: è usato per definire nuovi tipi di oggetto quando il gioco ne contiene molti e vogliamo che abbiano un comportamento simile. Per esempio, se un puzzle richiede di comporre parole con cubi su cui sono riportate le lettere dell'alfabeto (*alphabet block*), possiamo usare il codice:

An alphabet block is a kind of thing. [In italiano: A cubo alfabetico is a kind of thing.]

Dopo questo, possiamo scrivere regole per tutti i cubi alfabetici e Inform saprà di cosa stiamo parlando.

Il primo tipo di oggetto che incontreremo in questo libro è del genere scenario (*scenery*). Per rendere l'argomento più chiaro, comunque, la parola “scenery” in Inform non si riferisce a un genere (*kind*) di oggetto (*thing*); è una proprietà che può essere applicata agli oggetti. Un oggetto, un contenitore (*container*) o un supporto (*supporter*) possono essere scenari, una porta (*door*) può essere uno scenario, un dispositivo (*device*) può essere uno scenario e così via.

Se non dichiariamo qualcosa di più specifico nel creare uno scenario, Inform assume che l'oggetto creato è ordinario, non un dispositivo, una porta o un contenitore. Assume anche che lo scenario non possa essere spostato: cioè che il giocatore non possa prenderlo e portarlo a spasso.

Quando il gioco costruisce la descrizione da stampare, citerà ogni oggetto ordinario che si trova nella stanza, ma non menzionerà gli oggetti dello scenario (a meno che l'oggetto non sia un supporto che regge qualcosa; vedi pag. 79 per maggiori informazioni sui supporti). Inform assume che gli oggetti dello scenario siano già stati citati nella descrizione della stanza stessa quando è stata scritta, così è bene assicurarsi di averlo fatto.

Possiamo aggiungere uno scenario al nostro sentiero nella foresta in questo modo:

*The Forest Path is a room. “Tall old trees surround you.”
[In italiano: The sentiero nella foresta is a room. “Vecchi alberi ti circondano.”]*

The tall old trees are scenery in the Forest Path. The description is “Ancient oaks stretch out their heavy branches overhead, blocking the sun.”

[In italiano: The vecchi alberi are scenery in the sentiero nella foresta. The description is “Antiche querce distendono i loro pesanti rami, oscurando il sole.”]

Ora il giocatore che scrive X TREES (x alberi) leggerà una descrizione che aggiunge alcuni dettagli alla scena. Da notare che la frase inizia con:

The description is [...]

La frase “The description is” (la descrizione è) è opzionale con scenari e stanze, ma è *richiesta* con oggetti che possono essere presi e spostati (come è descritto nel Capitolo 3). Se non viene usato “The description is” con uno scenario, devi assicurarti di mettere sempre la descrizione nella frase che segue la creazione dell'oggetto. Questo codice funziona esattamente come quello mostrato sopra — ma solo con gli scenari, non con gli oggetti ordinari.

The tall old trees are scenery in the Forest Path. “Ancient oaks stretch out their heavy branches overhead, blocking the sun.”

[*In italiano:* The vecchi alberi are scenery in the sentiero nella foresta. “Antiche querce distendono i loro rami, oscurando il sole.”]

Se provi questo codice, presto scoprirai due problemi. In primo luogo, puoi usare i comandi X TREES, X TALL TREES o X OLD TREES (x alberi o x vecchi alberi), ma non puoi usare X OAKS o X ANCIENT OAKS (x querce o x antiche querce). Il parser risponderà: “You can’t see any such thing” (non puoi vedere nulla del genere) che è piuttosto stupido. Avviene perché Inform non legge mai all'interno del testo tra virgolette per verificare quali sinonimi sono stati usati. In secondo luogo, se provi a scrivere TAKE TREES (prendi gli alberi), il parser si lamenterà rispondendo: “That’s hardly portable” (non può essere trasportato). La parola “trees” (alberi) è plurale, così l'output corretto dovrebbe essere “Those are hardly portable” (non possono essere trasportati). Il parser non ha idea del fatto che l'oggetto “alberi” sia plurale. Dobbiamo aiutarlo in qualche modo.

Per risolvere il primo problema, aggiungiamo una regola “Understand” (comprendi). Scoprirai presto che molti degli oggetti nei tuoi giochi avranno bisogno di regole di questo tipo. Lo scopo principale è quello di aggiungere parole extra che il giocatore può utilizzare per riferirsi agli oggetti.

Per risolvere il secondo problema, dobbiamo fare in modo che il parser capisca che l'oggetto “vecchi alberi” è plurale (**plural-named**). Ci sono due modi per farlo. Esplicitamente, aggiungendo al nostro sorgente la frase: “The tall old trees are plural-named” (*in italiano:* the vecchi alberi are plural-named). O implicitamente, cambiando la frase con cui creiamo gli alberi in modo che dica: “Some tall old trees” (*in italiano:* Some vecchi alberi). Il secondo metodo è più semplice. Inform sa che quando gli oggetti sono creati con la parola “some” (alcuni), essi sono plural-named. Mettendo tutto assieme, gli alberi appariranno nel codice sorgente in questo modo:

Some tall old trees are scenery in the Forest Path. “Ancient oaks stretch out their heavy branches overhead, blocking the sun.” Understand “ancient”, “oaks”, “heavy”, “branches”, “branch”, and “tree” as the tall old trees.

[*In italiano:* Some vecchi alberi are scenery in the sentiero nella foresta. “Antiche querce distendono i loro rami, oscurando il sole.” Understand “antiche”, “querce”, “rami”, “ramo”, and “albero” as the vecchi alberi.]

Da notare che quando aggiungiamo parole al vocabolario, dobbiamo porre le virgole che separano i termini al di fuori delle virgolette (come è ovvio per un italiano [N.d.T.]).

L'uso di “some”¹⁴ per creare un oggetto plural-named è comodo, ma c'è un problema potenziale di cui occorre essere al corrente. In inglese, alcuni nomi sono “collettivi”. Esempi di questo tipo sono “sand” (sabbia) e “water” (acqua). Se scriviamo “Some sand is scenery in the Beach”, Inform sarà confuso e penserà che “sand” è un oggetto plural-named. Se dobbiamo costruire una frase che riguardi l'oggetto “sand”, deve iniziare con “The sand are...”. Per evitare il problema, dobbiamo creare “sand” usando “the”, non “some” — e poi dobbiamo dire a Inform di riferirsi a “sand” come “some sand” quando deve costruire una frase, non come “a sand”. Ecco come farlo:

[The sand is scenery in the Beach. The indefinite article of the sand is "some".](#)

Il termine “indefinite article” (articolo indefinito) si riferisce, in inglese, alle parole “a” e “an”.

Dopo aver creato i “vecchi alberi” come scenario, possiamo riferirci a quell'oggetto sia come “alberi vecchi” o semplicemente “alberi”. Il compilatore comprenderà entrambe le forme. E' bene, però, avere l'abitudine di usare sempre il nome completo. Se usiamo forme abbreviate, Inform di solito capirà cosa intendiamo, ma è possibile ottenere codice che contiene bug difficili da trovare.

La ragione è questa: puoi avere diversi oggetti il cui nome contiene “alberi” — i vecchi alberi nella foresta, gli alberi di pere nel frutteto e i vasi degli alberi bonsai nel giardino. Se scrivi nel sorgente solo “alberi”, il compilatore cercherà di capire a quale oggetto ti riferisci e solitamente indovinerà. Una volta ogni tanto, però, sarà confuso. Il risultato potrebbe essere disastroso: il giocatore potrebbe tentare di prendere i vasi degli alberi bonsai e finire per trasportare per sbaglio l'intera foresta o il frutteto.

Per la stessa ragione, è buona norma dare un nome ad ogni oggetto che contenga almeno un aggettivo. Accadono cose spiacevoli se un oggetto è chiamato pallone da spiaggia e un altro semplicemente pallone.

Se te lo chiedi — no, non è possibile riferirsi ai vecchi alberi come “the *antiche querce*”. Le parole nelle regole “Understand” sono fornite per la comodità del giocatore e non dell'autore.

All'inizio, i giocatori non sanno cosa è importante oppure no. Così tentano di fare tutto ciò che sembra possibile. Con i vecchi alberi, ci possiamo aspettare che venga usato il comando CLIMB TREES (cioè “sali sugli alberi”, o più probabilmente, CLIMB TREE, ovvero “scala l'albero”, visto che non è facile salire più alberi contemporaneamente). Normalmente l'output è questo:

```
>climb tree
I don't think much is to be achieved by that.
```

(Scala l'albero. Non credo che si ottenga molto da quest'azione.)

¹⁴ La discussione che segue, sui nomi collettivi, ha senso per un inglese che deve districarsi tra il proprio linguaggio e il “linguaggio naturale” di Inform (a causa dell'uso particolare di “some”). Per un italiano il problema non sussiste. [N.d.T.]

Non c'è nulla di sbagliato in quella frase, eccetto il fatto che è noiosa. Puoi ottenere qualcosa di più interessante utilizzando una regola “Instead”:

Instead of climbing the tall old trees:

say “Vicious flocks of sparrows and wrens dart down at you and peck at you with their sharp little beaks, driving you back.”

[*In italiano:*

Instead of climbing the *vecchi alberi*:

say “*Stormi di passeri ti assalgono, facendoti desistere.*”]

Il risultato è lo stesso: non è accaduto nulla. Il giocatore si trova ancora nella stanza chiamata “sentiero nella foresta” (e se il giocatore usa il comando X SPARROWS, non ce ne saranno nella stanza, visto che non li abbiamo creati), però il gioco è più interessante e divertente di prima. Scrivendo messaggi come questi, ti puoi persino chiedere che cosa quei passeri stiano tentando disperatamente di proteggere. Questo può suggerire un puzzle che può essere aggiunto al gioco, come gettare del beccime per distrarre gli uccelli. Dopo aver gettato il beccime, si potrebbe permettere al giocatore di salire sugli alberi, ma questo tipo di modifica del sorgente deve aspettare il prossimo capitolo.

Nota importante!

Leggendo questa *Guida ad Inform7*, troverai dozzine di esempi di codice che puoi copiare e incollare direttamente nella pagina Source del programma Inform per provare le funzionalità qui illustrate. Questi esempi sono tutti in **blu**. Sfortunatamente, se gli esempi contengono linee indentate, copiarle semplicemente dalla *Guida* nel gioco **non funziona**. Puoi selezionare il testo degli esempi in molti modi e copiarlo, ma quando incolli il testo, l'indentazione è scomparsa (così come gli spazi bianchi tra i blocchi di codice). Senza indentazione, Inform non può capire il tuo codice.

Per fare in modo che gli esempi funzionino, devi leggerli attentamente, capire dove sono le indentazioni (e di quante tabulazioni sono composte) ed inserirle tu stesso, una riga alla volta (vedi il Capitolo 8 della *Guida* per più dettagli sull'indentazione).

Puoi ripristinare le righe bianche aggiungendo dei ritorni a capo, oppure no — a te la scelta. La loro presenza non influisce sul funzionamento del codice, ma l'aggiunta di righe bianche rende il codice più leggibile.

Inoltre, se copi del testo a cavallo della fine di una pagina della *Guida*, il numero di pagina sarà copiato con il codice. In Inform dovrai cancellarlo.

Scenari distanti

Spesso accade, quando scrivi la descrizione di una “stanza” esterna, di voler menzionare cose che sono lontane — visibili, ma non accessibili. Inform non ha alcun meccanismo interno per gestire la situazione, ma possiamo facilmente crearne uno. Prima cerchiamo di capire quale sia il problema e poi come sia possibile risolverlo.

The Forest Path is a room. "Tall old trees surround you. The path continues north and south. In the distance to the west, off among the trees, you can see a crumbling stone wall."

[*In italiano: The sentiero nella foresta is a room. "Vecchi alberi ti circondano. Il sentiero prosegue a nord e a sud. Lontano verso ovest, tra gli alberi, puoi vedere un muro di pietra diroccato."*]

The crumbling stone wall is scenery in the Forest Path. "The wall is ancient and moss-covered." Understand "ancient" and "moss-covered" as the stone wall.

[*In italiano: The muro di pietra diroccato is scenery in the sentiero nella foresta. "Il muro è antico e coperto di muschio."* Understand “antico” e “coperto di muschio” as the muro di pietra.]

Questo codice funziona fino ad un certo punto. Il giocatore può esaminare il muro, ma Inform non sa che è molto lontano. Il risultato, quando il giocatore cerca di interagire con esso, è pessimo:

```
>x wall
The wall is ancient and moss-covered.
(x muro. Il muro è antico e coperto di muschio.)
```

```
>touch wall
You feel nothing unexpected.
(tocca il muro. Non senti nulla di insolito.)
```

```
>take wall
That's hardly portable.
(prendi il muro. Non si può trasportare.)
```

```
>push wall
It is fixed in place.
(spingi il muro. E' fisso dove si trova.)
```

Ovviamente, il personaggio del giocatore non può toccare o spingere il muro, perché è stato descritto (nella descrizione della stanza) come molto lontano. Per risolvere questo problema, creiamo una nuova proprietà (**property**). Gli oggetti in Inform hanno molte proprietà (puoi leggere le proprietà di ogni oggetto nel pannello del gioco scrivendo SHOWME seguito dal nome dell'oggetto). Ad esempio, il nome stampato (printed name) e la descrizione (description) sono proprietà. E' facile creare una nuova proprietà. Creiamo una proprietà che può assumere o uno o l'altro (ma non entrambi) i valori distant/near (distante/vicino). Ogni oggetto nel mondo virtuale sarà o distante o vicino, ma ciascuno di essi sarà vicino a meno che non si specifichi diversamente:

A thing can be distant or near. A thing is usually near.

Instead of doing anything other than examining to a distant thing:

```
say "[The noun] [if the noun is plural-named]are[otherwise]is[end if] too far away."
```

[In italiano:

Instead of doing anything other than examining to a distant thing:

```
say "[The noun] [if the noun is plural-named]sono troppo distanti[otherwise]è troppo distante[end if]."]
```

The crumbling stone wall is scenery in the Forest Path. "The wall is ancient and moss-covered." Understand "ancient" and "moss-covered" as the stone wall. The stone wall is distant.

[In italiano:

The *muro di pietra diroccato* is scenery in the *sentiero nella foresta*. "*Il muro è antico e ricoperto di muschio.*" Understand "*antico*" as the *muro di pietra*. The *muro di pietra* is distant.]

Dopo aver creato "distant" come proprietà, possiamo rendere il muro di pietra (o qualunque altra cosa) "distant" (lontano), semplicemente dichiarandolo. Ora l'output sarà coerente:

```
>touch wall
The crumbling stone wall is too far away.
(tocca il muro. Il muro di pietra diroccato è troppo distante.)
```

Il codice sopra riportato assume che un oggetto distante è sempre il primo oggetto a cui il giocatore si riferisce in un input. Cioè, è l'*oggetto diretto* del verbo. Nel codice sorgente di Inform, l'oggetto diretto è indicato come **noun**. Se è presente un secondo nome nel comando (solitamente preceduto da una preposizione), puoi riferirti ad esso come **second noun**.

Per metterci al sicuro, possiamo intercettare anche messaggi del tipo PUT VASE ON STONE WALL (metti il vaso sul muro di pietra). In questo caso, il nome di pietra è il secondo nome. Per gestirle questa situazione puoi scrivere:

Instead of doing anything when the second noun is a distant thing:

```
say "[The second noun] [if the second noun is plural-named]are[otherwise]is[end if] too far away."
```

[In italiano:

Instead of doing anything when the second noun is a distant thing:

```
say "[The second noun] [if the second noun is plural-named]sono troppo distanti[otherwise]è troppo distante[end if]."]
```

Questo codice non funziona bene se il comando implica, ad esempio, una discussione con un PNG (personaggio non giocante, in inglese NPC) a proposito di un oggetto distante. Intatti, ASK GUARD ABOUT STONE WALL (chiedi alla guardia del muro di pietra) produrrebbe l'output: "The stone wall is too far away" (il muro di pietra è troppo distante). Sistemare il problema ci porterebbe nel regno della programmazione avanzata di Inform. Se stai scrivendo un gioco dove questo aspetto diventa un problema, invia un messaggio su rec.arts.int-fiction e qualcuno potrebbe metterti nella giusta direzione. Potresti anche dare un'occhiata all'estensione chiamata "Far Away" di Jon Ingold.

Nota l'uso del test “if”, che è incluso nel testo di output della regola “Instead”. Se scriviamo semplicemente, “[The noun] is too far away” ([The noun] è troppo distante), gli oggetti distanti che sono plural-named, come le oche che volano nel cielo, non funzionerebbero bene. Leggeremmo questo:

```
>touch geese  
The geese is too far away.  
(tocca le oche. Le oche è troppo distante.)
```

Per le oche, potremmo scegliere di creare un oggetto distante chiamato lo “stormo di oche” e non renderle plural-named (poiché “stormo” è singolare). Una mandria di mucche si può gestire nello stesso modo, ma bisogna abituarsi ad usare questo tipo di test “if” ogni volta che si scrivono messaggi che possono applicarsi a molti oggetti differenti. Se installi “Plurality”¹⁵, un'estensione di Emily Short, avrai a disposizione un sistema più semplice per risolvere questo particolare problema; qui è stato mostrato il metodo base perché questo è un libro di testo. Imparare come farlo da soli è uno degli scopi del libro. Se includi “Plurality” nel codice sorgente (vedi “Estensioni per Inform” nel Capitolo 1, pag. 35), questo tipo di output è più lineare e si può scrivere:

Instead of doing anything other than examining to a distant thing:
say “[The noun] [is-are] too far away.”

[In italiano:

Instead of doing anything when the second noun is a distant thing:
say “[The noun] [is-are] fuori portata.”

Aggiungere stanze alla mappa

E' possibile scrivere brevi giochi completi che hanno luogo in una sola stanza¹⁶, ma molti giochi necessitano di più stanze. Il giocatore si sposta da una all'altra usando le normali direzioni della bussola — NORD (NORTH in inglese, o semplicemente N), SUDOVEST (SOUTHWEST in inglese, abbreviabili in SO e SW) e così via.

Altri metodi per spostarsi sono stati provati da diversi autori e, ad un certo punto, vorrai sperimentarli. In “Blue Lacuna” di Aaron Reed, puoi viaggiare da un luogo all'altro scrivendo il nome di qualcosa che si trova nella direzione in cui vuoi andare. In un gioco impostato in questo modo, CASTLE (castello) ti porterà vicino al castello (o ti farà entrare, se gli sei già vicino) e così via. Aaron ha pubblicato questo sistema come un'estensione, così puoi usarlo nel tuo gioco senza la necessità di conoscerne ogni dettaglio... sebbene sia più complesso della media delle estensioni, infatti devi aspettarti di doverlo studiare e di fare qualche test per farlo funzionare come desideri.

Se la tua storia si svolge su una nave, potresti rimpiazzare N, S, E, and W con FORWARD, AFT, PORT, and STARBOARD (l'**esempio 40** della Documentazione, “Fore”, mostra come farlo. Il **par. 3.26**, “Directions” offre qualche ulteriore suggerimento), ma nel tuo primo gioco, suggerisco di sfruttare le direzioni della bussola. Tutti i giocatori sanno come usarle.

15 Non è utile nel caso degli autori italiani, perché il meccanismo è già compreso nell'estensione “Italian” [N.d.T.]

16 Esistono addirittura competizioni tra autori in cui l'avventura testuale DEVE svolgersi in una sola stanza [N.d.T.]

Dove è finito?

La numerazione degli esempi nella Documentazione di Inform potrà cambiare nelle future versioni. Lo stesso avverrà per la numerazione delle pagine. Comunque, i nomi degli esempi e dei paragrafi dovrebbero rimanere inalterati. Se non riesci a trovare una pagina a cui ci si riferisce in questa *Guida*, considera la possibilità che la numerazione sia cambiata.

Inform permette di creare molto facilmente una mappa contenente stanze che sono connesse mediante le direzioni della bussola. Il **par. 3.2 e 3.3** della Documentazione, “Rooms and the map” (le stanze e la mappa) e “One-way connections” (connessioni unidirezionali) spiegano come farlo, ma qui ne daremo un'idea e suggeriremo modi per risolvere i problemi che si possono presentare.

Una volta creata la prima stanza, puoi crearne altre semplicemente descrivendo la mappa ad Inform, come in questo caso:

Forest Path is a room. “Tall old trees surround you. The path continues north and south from here.”

[In italiano:

The sentiero nella foresta is a room. “Vecchi alberi ti circondano. Il sentiero si dirama a nord e a sud.”]

Canyon View is north of Forest Path. “The path from the south ends here at the top of a cliff, from which you have a spectacular view of a canyon.”

[In italiano:

The vista sul canyon is north of the sentiero nella foresta. “Il sentiero termina a sud sulla cima di una collina da cui si può ammirare il canyon.”

Haunted Grove is south of Forest Path. “The trees press close around you, moaning faintly and waving their branches in an unsettling way. A path leads north.”

[In italiano:

The radura stregata is south of sentiero nella foresta. “Gli alberi si stringono intorno a te, gemendo debolmente e agitando i loro rami in modo inquietante. Un sentiero conduce a nord.”

Il testo crea una mappa con tre stanze — da nord a sud, la vista sul canyon, il sentiero nella foresta e la radura stregata. Inform è abbastanza intelligente da capire che quando affermi: “Canyon View is north of Forest Path” (la vista sul canyon è a nord del sentiero nella foresta), Canyon View deve a sua volta essere una stanza, perché le stanze e le porte sono le uniche cose che possono essere in relazione con altre stanze mediante le direzioni della bussola. Non abbiamo detto a Inform che Canyon View è una porta, così ipotizza che sia una stanza.

Inform assume anche che le connessioni descritte possano essere utilizzate all'andata e al ritorno. Cioè, poiché Haunted Grove è a sud di Forest Path, Forest Path sarà posizionato automaticamente a nord di Haunted Grove. Il giocatore che va a sud di Forest Path si troverà in Haunted Grove, mentre se si sposta a nord di Haunted Grove ritornerà a Forest Path.

Le direzioni che occorre seguire per lasciare una stanza sono spesso dette le uscite della stanza, però “exits” (uscite) non è un termine comprensibile per Inform, a meno che non si scriva del codice o si includa un'estensione che definisca tale parola.

E' facile creare delle connessioni a senso unico o che cambiano direzione. Mostriamo come farlo più avanti in questo stesso capitolo. Alcune connessioni di questo tipo possono essere utili per rendere il tuo mondo più verosimile, ma aggiungerne troppe confonde i giocatori.

Tutte le descrizioni delle stanze sopra riportate dicono al giocatore in quale direzione occorre spostarsi per lasciare la stanza. Questo è un buon metodo per informare il giocatore, sebbene possa far sembrare la scrittura un po' ripetitiva. Come è già stato detto, puoi scaricare ed installare un'estensione per Inform (“Exit Lister” di Eric Eve) che mostrerà le uscite disponibili nella barra di stato in cima alla finestra di gioco.

Offesa capitale (era Capital Offense)

In molte situazioni, Inform non è fiscale quando si tratta di usare le maiuscole con le parole scritte nel sorgente, ma non si può usare la maiuscola con le direzioni della bussola (a meno che non si tratti della prima parola di una frase). Qui seguono tre esempi. I primi due producono esattamente la stessa mappa, mentre il terzo è un bug e non si compila.

The Lounge is a room.
The Dining Room is north of the Lounge.

The Lounge is a room.
North of the Lounge is the Dining Room.

The Lounge is a room.
The Dining Room is North of the Lounge. [Error!]

Il **par. 3.2** della Documentazione, “Rooms and the map” (le stanze e la mappa), spiega come risolvere un piccolo, ma talvolta fastidioso, problema nel fornire i nomi alle stanze e definire le connessioni della mappa¹⁷: cosa succede se hai una stanza chiamata Hut (capanna) e vuoi chiamare un'altra stanza “South Of The Hut” (sud della capanna)? Questo non funziona:

South Of The Hut is south of the Hut. [Errore!]

E neppure questo funziona:

South Of The Hut is a room. South Of The Hut is south of the Hut. [Errore!]

¹⁷ Anche in questo caso, il problema non si pone con le avventure testuali scritte in italiano [N.d.T.]

Entrambe le frasi funzionerebbero se Inform notasse il modo in cui abbiamo usato le maiuscole, ma non lo fa. La soluzione è quella di utilizzare la parola “called” (chiamato):

[South of the Hut is a room called South Of The Hut.](#)

Questa frase sembra scritta da Gertrude Stein (una scrittrice degli inizi del ventesimo secolo, famosa per il verso “Rose is a rose is a rose”). Questa frase può essere letta come “‘South of the Hut’ è il nome di una nuova stanza. Questa nuova stanza è chiamata (anche se l'abbiamo già detto e ci stiamo ripetendo) South Of The Hut”. Se Inform avesse letto la frase in questo modo, avremmo una nuova stanza chiamata “South Of The Hut”, ma non sarebbe connessa con nessun'altra stanza della mappa.

Come vedi, la sintassi del “linguaggio naturale” di Inform può talvolta confondere o essere difficile da leggere... per un umano. La frase ha senso se la leggi in un modo differente: “South (la direzione sud) della Hut (una stanza che abbiamo già definito in Inform) è una stanza (un'altra — una nuova, questa volta) chiamata 'South Of The Hut'”.

Lo stesso problema può sorgere se vogliamo chiamare una stanza “Inside the Stone House” (dentro la casa di pietra). Ricorda che Inform interpreta “inside” e “outside” (o “in” e “out”) come direzioni in cui spostarsi. Così non possiamo scrivere:

[Inside the Stone House is inside from Haunted Grove. \[Error!\]](#)

[Mentre in italiano: [The interno della casa di pietra is inside from the radura stregata. \[Corretto!\]](#)]

Ecco come si può fare:

[Haunted Grove is south of Forest Path. "The trees press close around you, moaning faintly and waving their branches in an unsettling way. A path leads north, and a house built of stone stands here."](#)

[Inside from Haunted Grove is a room called Inside the Stone House. The description of Inside the Stone House is "A small, dimly lit room. A doorway to the north leads deeper into the house."](#)

Nota la frase “The description of Inside the Stone house is” (ovvero la descrizione di “Inside the Stone house” è...). Di solito, quando si definisce una stanza quella frase è superflua, ma qui è necessaria. Infatti, se iniziamo una frase semplicemente con “A small, dimly lit room” (una piccola stanza debolmente illuminata), Inform penserà che stiamo fornendo la descrizione di Haunted Grove, ma poiché abbiamo già assegnato una descrizione a Haunted Grove e non ci è permesso fornirne un'altra, comparirà un messaggio di errore. La regola è: Inform considera la *prima* stanza nella frase precedente (Haunted Grove) e ipotizza che sia ciò di cui stiamo parlando se iniziamo una nuova frase che comincia con una virgoletta. Poiché vogliamo scrivere la descrizione di “Inside the Stone House”, dobbiamo aiutare Inform a capire cosa abbiamo in mente e lo possiamo fare in modo esplicito con “the description is...”.

Scenari in cui si può entrare

Quando la stanza è un luogo esterno e un edificio è nelle vicinanze, definire l'esterno dell'edificio come scenario è una buona idea. Non è difficile, ma può portare a due piccoli problemi che è bene discutere.

In primo luogo, non possiamo attribuire all'esterno del palazzo lo stesso nome della stanza che rappresenta l'interno. Questo può capitare nel caso di un edificio con una sola stanza come una capanna o una stalla. Questo non funzionerebbe:

The Farmyard is a room. "A muddy farmyard. A little red barn stands to the north."

[*In italiano:* The cortile is a room. "Un cortile fangoso. A nord c[']è una piccola stalla rossa."]

The little red barn is scenery in the Farmyard. The description is "The barn is freshly painted a cheerful bright red."

[*In italiano:* The stalla rossa is a female scenery in the cortile. The description is "La stalla è pitturata di fresco di un allegro rosso brillante." (Nota "female": serve per attribuire l'articolo corretto [N.d.T.]

Little Red Barn is north of the Farmyard. "In the barn you can see some stalls and some hay." [Error!]

[*In italiano:* The stalla rossa is north of the cortile. "Nella stalla ci sono..." [Errore!]

Il compilatore obietterà, giustamente, che abbiamo tentato di nominare due cose — uno scenario e una stanza — con lo stesso nome. Ecco come si può risolvere il problema:

The barn-exterior is scenery in the Farmyard. The description is "The barn is freshly painted a cheerful bright red." Understand "little", "red", and "barn" as the barn-exterior. The printed name of the barn-exterior is "little red barn".

[*In italiano:* The esterno-stalla is a female scenery in the cortile. The description is "La stalla è pitturata di fresco di un allegro rosso brillante." Understand "rossa", and "stalla" as the esterno-stalla. The printed name of the esterno-stalla is "stalla rossa".]

Ora lo scenario ha il suo nome, esterno-stalla, ma abbiamo aggiunto "rossa" e "stalla" al vocabolario e abbiamo detto ad Inform che, se il gioco deve stampare il suo nome, deve chiamarlo "stalla rossa".

Il secondo problema che dobbiamo risolvere è questo: il giocatore può tentare di usare ENTER BARN (entra nella stalla) come comando. A meno che non specifichiamo diversamente, Inform dirà al giocatore che gli scenari non sono accessibili. Quando lo scenario è l'esterno di un edificio, questo comportamento confonde e non è elegante. La correzione è semplice. Permettiamo al giocatore di usare IN come comando, quando si trova nel cortile, in modo che possa entrare nella stalla:

Instead of entering the barn-exterior, try going north.

Instead of going inside in the Farmyard, try going north.

Quando l'oggetto in cui si può entrare è piccolo (come una cabina telefonica) potremmo definirlo come un contenitore accessibile (*enterable container*) e non come una stanza (vedi il paragrafo "Contenitori e supporti accessibili" nel Capitolo 3). Con oggetti più grandi, renderli stanze a se stanti funziona meglio.

Porte

Nel mondo reale, la maggior parte delle stanze sono separate da porte. In Inform, una porta è un oggetto speciale perché si trova in due stanze allo stesso tempo e le mette in comunicazione. Inform considera le porte in modo molto elementare, ma scopriremo un paio di modi per migliorare il loro comportamento base.

Per creare una porta, prima occorre creare le stanze che questa mette in comunicazione, ma *senza* connetterle mediante le direzioni della bussola. Infatti se si utilizzano le connessioni direzionali tra le stanze, Inform non permette di creare una porta tra di esse. Dopo aver creato le stanze, si può creare la porta e dichiarare a Inform le sue direzioni rispetto alle due stanze:

The Entry Hall is a room. [In italiano: The *ingresso* is a room.]
The Billiard Room is a room. [In italiano: The *sala da biliardo* is a room.]
The oak door is a door. The oak door is north of the Billiard Room and south of the Entry Hall.
[In italiano: The *porta di quercia* is a door. The *porta di quercia* is north of the *sala da biliardo* and south of the *ingresso*.]

Non possiamo chiamare la porta di quercia semplicemente porta e sperare che Inform capisca: dobbiamo aggiungere una frase extra per dire che la porta è una porta. Il codice che segue si compilerà, ma è un errore:

The Entry Hall is a room. [In italiano: The *ingresso* is a room.]
The Billiard Room is a room. [In italiano: The *sala da biliardo* is a room.]
The oak door is north of the Billiard Room and south of the Entry Hall. [Error!]
[In italiano: The *porta di quercia* is north of the *sala da biliardo* and south of the *ingresso*.]
(In italiano l'errore commesso è subito evidente [N.d.T.]

Se scriviamo in questo modo, Inform penserà che la “porta di quercia” è il nome di una stanza. Creerà una terza stanza chiamata “porta di quercia”, che si trova tra le altre due stanze. Spero che in una futura versione di Inform, chiamare qualcosa “door” la renda automaticamente una porta, ma per ora, occorre scrivere la frase aggiuntiva.

Se includi nel tuo gioco il codice mostrato sopra, molto presto noterai due cose. In primo luogo, la porta stessa sarà citata nell'output del gioco assieme agli altri oggetti nella stanza. In altre parole, una porta non fa parte dello scenario a meno che se ne dichiari la proprietà “scenery”. In secondo luogo, e questo risulta più fastidioso, se provi ad andare a sud dall'ingresso (o a nord dalla sala da biliardo), ti sarà risposto: “You can't, since the oak door is in the way” (non puoi, perché la porta di quercia è sul cammino). Forzare il giocatore a scrivere OPEN OAK DOOR (apri la porta di quercia) come un comando separato è ridondante, in special modo perché, di base, Inform non dice neppure che il problema consiste nel fatto che la porta è chiusa; dice solo che la porta è “sul cammino”. L'ideale sarebbe che, se la porta è chiusa e si può aprire senza bisogno di una chiave, questo venga fatto automaticamente.

Quello che segue è un modo per aggirare il problema (workaround) utilizzando una regola “Before”:

Before going through the oak door:

if the oak door is closed:

say "(first opening the oak door)[paragraph break]";

now the oak door is open.

[*In italiano:*

Before going through the *porta di quercia*:

if the *porta di quercia* is closed:

say "(prima apri la *porta di quercia*)[paragraph break]";

now the *porta di quercia* is open.]

Poiché questa regola “Before” termina senza un risultato, cioè non ha successo e neppure fallisce, come è spiegato nella Documentazione nel **par. 12.2**, “How actions are processed” (come vengono eseguite le azioni) e nel **par. 18.10**, “Success and failure” (successo e fallimento), l'azione continuerà. Dopo che la regola “Before” ha aperto la porta, il giocatore potrà passarci attraverso. Un'altra soluzione, se hai molte porte nel tuo gioco, è quella di non crearle per nulla come porte. Quando devi specificare una connessione nella descrizione di una stanza, chiamala semplicemente “doorway” (in italiano basta chiamare l'oggetto porta, ma senza dichiararlo come “door” [N.d.T.]). A molti giocatori non importa se viene sacrificato un po' di realismo in modo da rendere il tutto più facile da giocare.

Sono disponibili due utili estensioni che risolvono il problema sopra descritto: “Locksmith” di Emily Short e “Implicit Actions” di Eric Eve (in italiano occorre usare “Locksmith IT” e “Implicit Actions IT” di Leonardo Boselli, che corrispondono ai precedenti ma sono tradotti [N.d.T.]

Puoi usare il genere “door” in Inform per creare altri oggetti che connettono stanze, come cancelli, scale e ponti. Una scala o un ponte probabilmente non si potranno aprire: saranno permanentemente aperti. La ragione di crearli come “door” è quella di renderli visibili in entrambe le stanze connesse. Per esempio:

The wooden bridge is a scenery door. The wooden bridge is south of the Meadow and north of the Forest Path. The wooden bridge is open and not openable. The description is "It's a handsome old bridge. Curiously, there's no creek running beneath it."

[*In italiano:* The *ponte di legno* is a scenery door. The *ponte di legno* is south of the *brughiera* and north of the *sentiero nella foresta*. The *ponte di legno* is open and not openable. The description is "*Un bel ponte di legno. Curiosamente, nessun torrente scorre al di sotto di esso.*"]

E' una comodità che Inform comprenda il comando CROSS BRIDGE (attraversa il ponte), perché non hai bisogno di programmarlo da solo, ma se vuoi creare una scala utilizzando il genere “door”, devi scrivere una regola “Instead” per gestire CLIMB LADDER (sali la scala):

Instead of climbing the rickety ladder:

try entering the rickety ladder.

[*In italiano:*

Instead of climbing the *scala ripida*:

try entering the *scala ripida*.]

Porte chiuse a chiave

Una porta chiusa a chiave (locked door) ha un comportamento leggermente differente. Personalmente, non ritengo che sia un puzzle particolarmente interessante quello che coinvolge una porta chiusa a chiave e la cui chiave corrispondente sia stata nascosta. Centinaia di giochi hanno utilizzato le porte chiuse a chiave e, se nelle vicinanze c'è uno zerbino o una pianta in vaso, i giocatori sanno che occorre usare i comandi LOOK UNDER MAT (guarda sotto lo zerbino) o SEARCH POT (cerca nel vaso). Semplicemente non c'è più divertimento. Ho modificato questo cliché in un paio di modi. In un gioco ho incluso sia una porta chiusa a chiave che non può essere aperta perché non c'è la chiave e un'altra in cui un altro personaggio fornisce al giocatore la chiave spontaneamente senza che il giocatore glielo chieda. Le porte che possono essere aperte solo da un lato (dopo aver scoperto l'entrata segreta ad una stanza) sono un po' più interessanti.

Ecco come si può creare una porta chiusa a chiave in Inform:

The oak door is north of the Billiard Room and south of the Entry Hall. The oak door is a door. The oak door is scenery. The oak door is lockable and locked. The brass key unlocks the oak door.

[In italiano:

The *porta di quercia* is north of the *sala da biliardo* and south of the *ingresso*. The *porta di quercia* is a female door. The *porta di quercia* is scenery. The *porta di quercia* is lockable and locked. The *chiave di ottone* unlocks the *porta di quercia*.]

Before going through the oak door:

if the oak door is closed:

if the oak door is not locked:

say "(first opening the oak door)[paragraph break]";

now the oak door is open;

otherwise if the player carries the brass key:

say "(first unlocking the oak door with the brass key, then opening the door)

[paragraph break]";

now the oak door is unlocked;

now the oak door is open.

[In italiano:

Before going through the *porta di quercia*:

if the *porta di quercia* is closed:

if the *porta di quercia* is not locked:

say "(prima apri la *porta di quercia*)[paragraph break]";

now the *porta di quercia* is open;

otherwise if the player carries the *chiave di ottone*:

say "(prima apri la *porta di quercia* con la *chiave di ottone*, poi apri la porta)

[paragraph break]";

now the *porta di quercia* is unlocked;

now the *porta di quercia* is open.]

The player carries the brass key. **[In italiano:** The player carries the *chiave di ottone*.]

Nell'esempio sopra riportato sono stati introdotti alcuni elementi. In primo luogo, se qualcosa è aperto o chiuso, possiamo renderlo passibile di chiusura a chiave (lockable). "Lockable" è una proprietà di determinati tipi di oggetti: porte e contenitori possono essere "lockable" e nient'altro (tecnicamente non è vero. Possiamo anche creare nuovi tipi di oggetti — ad esempio un detonatore — e permettere che oggetti di quel tipo siano forniti di chiusura a chiave e chiusi a chiave — "lockable" and "locked"). Inoltre, se qualcosa è "lockable", all'inizio del gioco può essere sia "locked" o "unlocked" (chiuso a chiave o meno). Durante il gioco, il giocatore che ha la chiave giusta può aprirlo — o il tuo codice può farlo in risposta a determinate azioni del giocatore. Per esempio, una porta con un chiavistello vecchio-stile può essere bloccata e sbloccata (ancora "locked" and "unlocked") per mezzo dei comandi BAR DOOR (sbarra la porta) o UNBAR DOOR (sblocca la porta). Una porta di questo tipo potrebbe non avere alcun tipo di chiave.

Puoi scrivere del codice che faccia in modo che una porta che si può chiudere a chiave non lo sia più, ma questo può diventare utile solo se c'è un modo, durante il gioco, per rompere il lucchetto. Più spesso, una cosa che può essere chiusa a chiave resterà tale (allo stesso modo se non può essere chiusa a chiave, ovviamente) durante tutto il corso del gioco.

Se leggi il **par. 3.13** della Documentazione, "Locks and Keys" (lucchetti e chiavi), imparerai tre modi differenti per dire a Inform che una certa chiave può essere usata per chiudere un oggetto che si può chiudere a chiave. La frase che è stata usata più sopra, "The brass key unlocks the oak door", è la più semplice delle tre.

Nella versione attuale di Inform, un oggetto che può essere chiuso a chiave può avere solo una chiave. Di solito questo non è un grave problema. Se ne hai bisogno, tuttavia, puoi scrivere una regola "Instead" che permette anche ad una seconda chiave l'apertura. Qui segue un esempio piuttosto banale. Supponiamo di aver già detto a Inform che la "piccola chiave" apre l'"amuleto d'oro", ma vogliamo usare come chiave anche la banana:

```
Instead of unlocking the gold amulet with the banana:
  if the player does not carry the banana:
    say "You don't seem to have the banana.";
  otherwise if the amulet is not locked:
    say "The gold amulet doesn't seem to be locked.";
  otherwise:
    now the amulet is not locked;
    say "You unlock the gold amulet with the banana."
```

[In italiano:

```
Instead of unlocking the amuleto d'oro with the banana:
  if the player does not carry the banana:
    say "Non hai la banana.";
  otherwise if the amuleto is not locked:
    say "L'amuleto d'oro non sembra chiuso a chiave.";
  otherwise:
    now the amuleto is not locked;
    say "Hai aperto l'amuleto d'oro con la banana."]
```

Se crei la porta di quercia come è stato spiegato nella pagina precedente, sarà prima automaticamente sbloccata con la chiave e poi aperta, se il giocatore possiede la chiave. L'unica controindicazione di quell'esempio è che si dà per scontato che il giocatore sappia che la chiave d'ottone apra la porta di quercia. Se vuoi fare in modo che il giocatore sia forzato a scoprire questa proprietà, devi trovare un modo per registrare ciò che il giocatore conosce. Tenere una traccia delle conoscenze del giocatore (o comunque del personaggio del giocatore) non è eccessivamente difficile, ma i dettagli possono cambiare da un gioco all'altro, così lo lascio come esercizio per i programmatori esperti di Inform.

Porte segrete

Una porta segreta è ancora più interessante di una porta chiusa a chiave — qualcosa che non sembra per niente una porta finché la sua presenza non viene rivelata. "Secret Doors" di Andrew Owen è una semplice estensione che permette di crearle. Se includi questa estensione e poi crei la porta segreta, la porta pretenderà di non esistere fino a che nel gioco non accade qualcosa che la riveli. Per esempio, se la porta è nascosta da un pannello di quercia, si potrebbe scrivere:

[Include Secret Doors by Andrew Owen.](#)

The Billiard Room is a room. "Hand-rubbed oak paneling adds a warm glow above the broad green felt surface of the billiard table." *[In italiano: The sala da biliardo is a room. "Pannelli di quercia rifiniti a mano creano una calda atmosfera introno al pannello verde del tavolo da biliardo."]*

The Small Windowless Room is a room. "It smells dusty in here, as if the secret door hasn't been opened in ages." *[In italiano: The piccola stanza senza finestre is a room. "Qui c'è odore di chiuso, come se la porta segreta non fosse stata mai aperta per secoli."]*

The oak door is north of the Billiard Room and south of the Small Windowless Room. The oak door is a secret door. *[In italiano: The porta di quercia is north of the sala da biliardo and south of the piccola stanza senza finestre. The porta di quercia is a secret door.]*

The oak wall paneling is scenery in the Billiard Room. The description is "Richly carved oak paneling covers the north wall[if the oak door is open]. One of the panels has been opened; it's actually a door[otherwise if the oak door is revealed]. One of the panels has an unusually wide seam around it[end if]." Understand "carved" and "panel" as the paneling.

[In italiano:

The pannelli di quercia is plural-named scenery in the sala da biliardo. The description is "Pannelli di quercia finemente intarsiati ricoprono la parete a nord[if the porta di quercia is open]. Uno dei pannelli è stato aperto; si tratta di una porta[otherwise if the porta di quercia is revealed]. Uno dei pannelli è circondato da un'insolita fessura[end if]." Understand "intarsiato" and "pannello" as the paneling.]

After examining the oak wall paneling for the first time:

now the oak door is revealed;

say "One of the panels has an unusually wide seam around it. On closer inspection, the panel proves to be a door!"

[In italiano:

After examining the pannelli di quercia intarsiati for the first time:

now the porta di quercia is revealed;

say "Uno dei pannelli è circondato da un'insolita fessura. Dopo un'attenta ispezione, il pannello si rivela essere una porta!"

Il codice riportato produce esattamente quel tipo di interazione che ci aspettiamo da una porta segreta:

```
>n  
You can't go that way.  
(n. Non puoi andare in quella direzione.)
```

```
>open door  
You can't see any such thing.  
(apri la porta. Non vedi nulla di simile.)
```

```
>x paneling  
Richly carved oak paneling covers the north wall.  
(x pannelli. Pannelli finemente intarsiati ricoprono la parete a nord.)
```

```
One of the panels has an unusually wide seam around it. On closer  
inspection, the panel proves to be a door!  
(Uno dei pannelli è circondato da un'insolita fessura. Ad un'attenta ispezione si rivela essere una porta!)
```

```
>open door  
You open the oak door.  
(apri la porta. Apri la porta di quercia.)
```

```
>n
```

Small Windowless Room

```
It smells dusty in here, as if the secret door hasn't been opened in ages.  
(n. Piccola stanza senza finestre. Qui c'è odore di chiuso, come se la porta segreta non fosse...)
```

Porte pericolose

Uno dei miei studenti mi ha recentemente chiesto come creare una porta che si chiude alle spalle del giocatore. Ho pensato che fosse un puzzle interessante, così l'ho programmato. L'esempio riportato sotto mostra come usare una regola "After" per influenzare ciò che avviene quando il giocatore si sposta da una stanza all'altra:

The Corridor is a room. "The corridor stretches east and west from here. A massive stone door stands invitingly open to the north." [*In italiano:* The corridoio is a room. "Il corridoio si estende da est a ovest. Una massiccia porta di pietra si apre in modo invitante verso nord."]

The Dank Cell is a room. "This cramped chamber smells of mold, and other things that are a lot less pleasant than mold. You can hear rats scurrying behind the walls. A massive stone door to the south is the only visible exit ... using the term 'exit' loosely, as the door [one of]has just slammed shut with a sound suggestive of finality and doom[or]is firmly shut[stopping]."

[*In italiano:* The umida cella is a room. "Questa cella puzza di muffa e di altre cose ancora più spiacevoli della muffa. Una massiccia porta di pietra a sud è la sola uscita visibile... anche se il termine 'uscita' è improprio, infatti la porta [one of]si è improvvisamente chiusa con un fragore che aveva un sentore di definitivo e apocalittico[or]è solidamente chiusa[stopping]."]

The massive stone door is a door. The massive stone door is scenery. The massive stone door is north of the Corridor and south of the Dank Cell. The description is "It's quite an imposing-looking stone door." The massive stone door is open, lockable, and locked.

[In italiano: The *massiccia porta di pietra* is a female door. The *massiccia porta di pietra* is scenery. The *massiccia porta di pietra* is north of the *corridoio* and south of the *umida cella*. The description is "*Una porta di pietra alquanto imponente.*" The *massiccia porta di pietra* is open, lockable, and locked.]

Instead of closing the massive stone door:

if the massive stone door is open:

say "...but it looks so inviting! Why not just step through it and see what's on the other side?";

otherwise:

say "It seems already to have closed itself without your lifting a finger."

[In italiano:

Instead of closing the *massiccia porta di pietra*:

if the *massiccia porta di pietra* is open:

say "... è così invitante! Perché non attraversarla per vedere cosa c'è dall'altro lato?";

otherwise:

say "*Sembra che si sia già chiusa senza bisogno di muovere un dito.*"

After going north from the Corridor:

now the massive stone door is closed;

say "As you step through the stone door, it swings shut with a terrible loud BOOM!";

continue the action.

[In italiano:

After going north from the *corridoio*:

now the *massiccia porta di pietra* is closed;

say "*Appena oltrepassata la porta, questa si chiude con un sonoro BOOM!*";

continue the action.]

Noterai che la *massiccia porta di pietra* è inizialmente aperta — ma è anche chiusa a chiave. Inform non si pone problemi quando deve creare porte chiuse a chiave che sono aperte. Appena la porta è chiusa, la sua condizione di essere chiusa a chiave impedisce che venga aperta nuovamente.

Per imparare qualcosa in più sulla sintassi della descrizione delle stanze, che include “[one of]” (uno di) e “[stopping]”, vedi “Inserimenti nel testo” a pag. 176.

Se questa porta fosse usata in un vero gioco, ci dovrebbe essere qualche tipo di uscita nascosta nella cella. Nascondere un'uscita occulta in una cella che sembra non avere uscite... beh, diciamo che ho visto parecchi autori provarci ed alcuni tentativi sono più convincenti di altri. L'idea che ci sia una botola sotto un tappeto¹⁸ mi sembra che non funzioni: perché un carceriere di una certa esperienza dovrebbe costruire una cella con una botola nel pavimento? Fornire al giocatore un attrezzo con cui segare le sbarre alla finestra potrebbe essere un puzzle migliore.

18 In “**Enigma**”, famosa avventura in italiano di Marco Vallarino (<http://www.marcovallarino.it>), il giocatore si trova inizialmente in una cella. In questo caso, la botola non è sotto un tappeto, ma sotto il letto. [N.d.T.]

Finestre

Le stanze reali hanno spesso finestre. Le finestre hanno alcune interessanti caratteristiche. Quando guardi dalla finestra, normalmente, vedi cosa c'è fuori. Se la finestra è aperta, puoi anche calarti da essa (oppure no). Una finestra si trova normalmente in due stanze, come una porta. Se la stanza che si trova da una parte è illuminata, è molto improbabile che quella dall'altra parte non lo sia (per maggiori dettagli sulle stanze buie, vedi pag. 69).

Il modo più semplice per ottenere una finestra attraverso cui il giocatore si può calare è definirla come "door". Questa scelta gestirà automaticamente il passaggio del giocatore e manterrà i due lati della finestra "sincronizzati" relativamente al fatto di essere aperti o chiusi. Poiché Inform non comprende "climb through" (calati da), creeremo una nuova azione per permettere al giocatore di usare la finestra:

The Lab is a room. "Welcome to the Test Lab. Many devious tests are conducted here. There's a wide window in the north wall." [*In italiano:* The Lab is a room. "Benvenuti nel Test Lab. Qui si conducono molti test inquietanti. Sulla parete nord si apre un'ampia finestra."]

The Porch is a room. "There's a wide window in the south wall."
[*In italiano:* The portico is a room. "Sulla parete nord si apre un'ampia finestra."]

The wide window is a door. The wide window is scenery. The wide window is north of the Lab and south of the Porch. [*In italiano:* The ampia finestra is a door. The ampia finestra is scenery. The ampia finestra is north of the Lab and south of the portico.]

Climbing through is an action applying to one thing. Understand "climb through [something]", "climb in [something]", "climb out [something]", and "climb out of [something]" as climbing through.
[*In italiano:* Climbing through is an action applying to one thing. Understand "calati da [something]", "calati dal/dall/dallo/dalla/dai/dagli/dalle [something]" as climbing through.]

Check climbing through:
if the noun is not a door:
say "You can't climb through [a noun]!" instead;
otherwise:
try entering the noun instead.

[*In italiano:*
Check climbing through:
if the noun is not a door:
say "Non puoi calarti da [a noun]!" instead;
otherwise:
try entering the noun instead.]

I comandi LOOK THROUGH WINDOW (guarda attraverso la finestra) e LOOK IN WINDOW (guarda dentro la finestra) fanno in modo che Inform esegua l'azione SEARCH (cerca). Comunque, LOOK OUT WINDOW e LOOK OUT OF WINDOW (guarda fuori dalla finestra) non sono compresi. Possiamo usare una regola "Instead" per fornire una descrizione di ciò che si trova fuori e aggiungere una direttiva "Understand" che dia al giocatore altri due modi per guardare dalla finestra.

Instead of searching the wide window:

if the location is the Porch:

say "There seems to be a laboratory in there, but you can't make out any details.";

otherwise:

say "Through the window you can see the porch, but the view isn't good enough for you to make out any details."

[*In italiano:*

Instead of searching the *ampia finestra*:

if the location is the *portico*:

say "*Sembra che ci sia un laboratorio all'interno, ma non riesci a distinguere i dettagli.*";

otherwise:

say "*Attraverso la finestra puoi vedere il portico, ma non riesci a distinguere i dettagli.*"]

Understand "look out of [something]" and "look out [something]" as searching.

[*In italiano:* Understand "guarda fuori da [something]" as searching.]

Se vogliamo menzionare qualunque oggetto grande abbastanza perché possa essere visto attraverso la finestra, come un razzo su una rampa di lancio, possiamo crearlo come scenario distante, utilizzando le tecniche già mostrate in precedenza in questo capitolo.

“Non puoi andare in quella direzione.”

Un giocatore che cerca, in una determinata stanza, di andare in una direzione della mappa lungo la quale non ci sono connessioni con altre stanze, si sentirà rispondere: “You can’t go that way” (non puoi andare in quella direzione). Questo messaggio ha un vantaggio: è perfettamente chiaro. Il giocatore sa che non deve più tentare di spostarsi in quella direzione, ma non è un messaggio molto descrittivo. Inoltre, in un ambiente all'aperto, come un campo, la frase “You can’t go that way” non è realistica e anche piuttosto stupida.

Fortunatamente, è facile scrivere messaggi sostitutivi più interessanti:

Instead of going nowhere from Forest Path:

say "You take a few steps into the forest, but deciding it might not be safe, you return to the path."

[*In italiano:*

Instead of going nowhere from the *sentiero nella foresta*:

say "*Muovi alcuni passi nella foresta, ma decidi che può essere pericoloso, così torni sul sentiero.*"]

Nota che la regola dice “going nowhere *from*” (andando in nessun luogo da). Scrivendo “going nowhere *in*” non funzionerebbe. Comunque, ci sono altre situazioni in cui “going in” (andare verso) è necessario e “going from” non funziona. Il concetto base è: “going from” assume che il movimento ha avuto successo. Nel codice precedente, il giocatore è riuscito... ad andare da nessuna parte.

Questo codice funziona finché ci si sposta camminando, ma il giocatore otterrà lo stesso output in risposta di UP o DOWN, e non è la stessa cosa. Possiamo ulteriormente personalizzare i messaggi “You can’t go that way” in questo modo:

Instead of going up in Forest Path:

say "The trees have no branches low enough for you to reach them."

[In italiano:

Instead of going up in the *sentiero nella foresta*:

say "*Gli alberi non hanno rami abbastanza bassi perché tu possa raggiungerli.*")

Instead of going down in Forest Path:

say "There are no gaping holes or open mineshafts in the vicinity."

[In italiano:

Instead of going down in the *sentiero nella foresta*:

say "*Non ci sono buche o gallerie nelle vicinanze.*")

A seconda delle caratteristiche delle stanze nella tua mappa, potresti voler scrivere dei messaggi "You can't go that way" differenti per ciascuna di esse. Però prendiamo in prestito un'idea del prossimo paragrafo, sulle regioni, per suggerire un approccio più efficiente. Dopo aver detto a Inform che Forest Path (sentiero nella foresta), Canyon View (vista sul canyon) e Haunted Grove (radura stregata) sono tutti nella regione chiamata Forest Area (area della foresta), possiamo scrivere regole "Instead" che si applicano a tutta la regione:

Instead of going nowhere from the Forest Area:

say "You take a few steps into the forest, but deciding it might not be safe, you return to the path."

Instead of going up in the Forest Area:

say "The trees have no branches low enough for you to reach them."

Instead of going down in the Forest Area:

say "There are no gaping holes or open mineshafts in the vicinity."

Se stai usando l'estensione "Secret Doors", dovrai fare del lavoro extra, perché una porta segreta che non è stata scoperta produrrà il messaggio base "You can't go that way", *non* il messaggio modificato per mezzo della regola "Instead" scritta sopra. Inoltre, scrivere un messaggio personalizzato può essere sfruttato per fornire al giocatore un indizio. Seguendo il codice scritto in precedenza nel paragrafo sulle porte segrete, puoi aggiungere qualcosa del tipo:

Before going through the oak door when the oak door is unrevealed:

say "You bump your nose on the oak paneling. Odd -- you had somehow absent-mindedly thought there ought to be a door there.";

rule succeeds.

[In italiano:

Before going through the *porta di quercia* when the *porta di quercia* is unrevealed:

say "*Hai sbattuto il naso sul pannello di quercia. Strano – hai pensato sbadatamente che in quella direzione ci potesse essere una porta.*";

rule succeeds.]

Quello sarà l'output se il giocatore cerca di spostarsi attraverso la porta segreta prima che sia scoperta.

Doppie uscite e connessioni serpeggianti

Supponiamo di aver creato una stanza descritta in questo modo:

The Cellar is down from the Kitchen. "The low ceiling of this little room is festooned with cobwebs, and the floor is dirt. A rough doorway leads east, and the stairs up to the kitchen are built into the north wall."

[In italiano:

The cantina is down from the cucina. "Il basso soffitto di questa piccola stanza è ricoperto di ragnatele ed il pavimento è sporco. Un rozzo passaggio porta ad est, mentre le scale che portano alla cucina sono costruite sulla parete a nord."]

Poiché abbiamo detto che la cantina è sotto la cucina, Inform crea una connessione alto/basso tra le due stanze. Noterai però che la descrizione della stanza (per l'importanza di aggiungere dettagli) sostiene che le scale sono a nord. Tuttavia, se il giocatore va verso nord, gli sarà detto: "You can't go that way" (non puoi andare in quella direzione), cosa che risulta poco realistica, ma che è facile da correggere:

Instead of going north in the Cellar, try going up.

[In italiano: Instead of going north in the *cantina*, try going up.]

Da notare che Inform insiste che si scriva "try going up" (cerca di andare su), non semplicemente "go up" (vai su). Il motivo è dovuto al fatto che non si può garantire che l'azione abbia successo. Andare in su può non riuscire per diverse ragioni: magari le scale potrebbero crollare, intrappolando il giocatore nella cantina! Ecco perché la parola "try" (cerca) viene utilizzata così spesso nel codice Inform.

Talvolta abbiamo bisogno di una connessione a senso unico tra due stanze. Può darsi che ci sia uno scivolo nella cantina che il giocatore può scendere, ma che non è in grado di risalire. E' bene avvertire il giocatore che lo scivolo è pericoloso (comunque, questo esempio è simile ad uno dei primi puzzle di Zork). In Inform ci sono due modi per farlo.

The Cellar is down from the Kitchen. "The low ceiling of this little room is festooned with cobwebs, and the floor is dirt. A rough doorway leads east, and the stairs up to the kitchen are built into the north wall. In the southeast corner is a hole that looks wide enough to enter, but if you climb down it, there's no guarantee you'll be able to get back up."

[In italiano:

The cantina is down from the cucina. "Il basso soffitto di questa piccola stanza è ricoperto di ragnatele ed il pavimento è sporco. Un rozzo passaggio porta ad est, mentre le scale che portano alla cucina sono costruite sulla parete a nord. Nell'angolo di sudest c'è un buco che sembra abbastanza largo per consentire il passaggio, ma se ti cali attraverso di esso, non è garantito che riuscirai a risalire."]

The Spider-Infested Cavern is down from the Cellar. "Spiders! Thousands of them!" Up from the Spider-Infested Cavern is nowhere.

[In italiano:

The caverna infestata di ragni is down from the cantina. "Ragni! A migliaia!" Up from the caverna infestata di ragni is nowhere.]

In questo esempio abbiamo creato una connessione a senso unico dicendo a Inform che la direzione verso l'alto dalla caverna non porta da nessuna parte. L'altro modo per farlo è una regola "Instead":

*Instead of going up in the Spider-Infested Cavern, say "The hole is too steep for you to climb back up."
[In italiano: Instead of going up in the caverna infestata di ragni, say "Il buco è troppo ripido perché tu possa risalirlo."]*

Il vantaggio di usare una regola "Instead" è che puoi informare il giocatore su qual è il problema.

Nel gioco "Adventure" originale, alcune stanze non erano connesse normalmente, ma piuttosto con connessioni "serpeggianti". Uno dei puzzle principali del gioco consisteva nel comprendere come realizzare una mappa affidabile. Questo tipo di puzzle non è molto usato nei giochi moderni, ma una volta ogni tanto puoi voler creare una connessione tra due stanze che è tortuosa piuttosto che diretta. Per esempio, la descrizione della stanza può informare il giocatore dicendo: "Puoi andare ad est, girato l'angolo del palazzo". Per ritornare dopo essere andato ad est, potresti dover andare a nord. Ecco come creare questo tipo di connessione:

*Deeper in the Cellar is east of the Cellar. "This little room smells awful."
The Cellar is north of Deeper in the Cellar.
West of Deeper in the Cellar is nowhere.
South of the Cellar is nowhere.*

*[In italiano:
The fondo della cantina is east of the cantina. "Questa piccola stanza puzza terribilmente."
The cantina is north of fondo della cantina.
West of fondo della cantina is nowhere.
South of the cantina is nowhere.]*

Da notare che abbiamo detto a Inform che dopo essere andati ad est dalla cantina, non possiamo tornare dove eravamo andando a ovest — dobbiamo andare a nord. Ecco il motivo per cui le due linee "nowhere" (da nessuna parte) sono state aggiunte. Questo rende il mondo virtuale un poco più realistico. E' una buona idea, in questi casi, includere "Exit Lister" di Eric Eve. Questa estensione elencherà le uscite da ogni stanza nella barra di stato — e se una stanza con una sola vera uscita ne mostra due sulla barra di stato, il giocatore può confondersi.

Per una sintassi più concisa che produce connessioni "ad angolo" fra stanze, vedi l'esempio 7 ("Port Royal 2") nella Documentazione. Possiamo realizzarlo in questo modo:

*East of the Cellar is north of Deeper in the Cellar.
[In italiano: East of the cantina is north of fondo della cantina.]*

Una connessione ancora più confusa è quella che connette l'uscita di una stanza con la stanza stessa, facendo tornare indietro il giocatore. Per farlo, occorre scrivere qualcosa del tipo:

*East of Deeper in the Cellar is Deeper in the Cellar.
[In italiano: East of fondo della cantina is fondo della cantina.]*

Cambiare la mappa durante il gioco

In molti giochi, il mondo virtuale funzionerà come vuoi se crei le connessioni tra le stanze all'inizio del gioco (cioè, quando crei le stanze nel tuo sorgente) e poi non le modifichi più. Qualche volta invece, potresti far scomparire qualche connessione durante l'esecuzione del gioco — forse perché le stanze fanno parte della struttura di una caverna e una frana ha bloccato un tunnel. O magari una porta magica è improvvisamente apparsa in una stanza, creando una connessione che prima non c'era. La porta magica potrebbe persino avere una sua volontà e sparire di nuovo (le porte non possono essere spostate in Inform; se vuoi che una porta magica si sposti, devi scrivere del codice aggiuntivo per gestire un oggetto che risponde ai comandi del giocatore come se fosse una porta).

Per eliminare un'uscita da una stanza, cambia l'uscita per farla connettere a “nowhere”, cioè da nessuna parte (è una buona idea farlo da entrambe le parti, nel caso ci sia un altro percorso che può portare il giocatore dall'altro lato dell'uscita bloccata). Per ripristinare un'uscita o crearne una nuova, modificala in modo che punti alla stanza di destinazione.

Qui di seguito puoi vedere un semplice gioco che spiega come ottenere l'effetto. Ci sono due connessioni nord-sud (tra il soggiorno e la cucina e tra la stanza da bagno e la camera da letto), ma solo una di esse è presente allo stesso momento nella mappa, a seconda di quale pulsante premi. Per rendere il cambiamento facile da verificare, ho messo due pulsanti su un “backdrop” (sfondo — leggi più avanti per capire di cosa si tratta), così sono sempre presenti non importa in quale stanza ci troviamo. Ho anche incluso “Exit Lister”, che elenca le uscite attive sulla barra di stato. Nota che la barra di stato verrà aggiornata ogni volta che si preme un pulsante.

[Include Exit Lister by Eric Eve.](#)

The Living Room is a room.	[<i>In italiano:</i> The <i>soggiorno</i> is a room.]
The Kitchen is south of the Living Room.	[<i>In italiano:</i> The <i>cucina</i> is south of the <i>soggiorno</i> .]
The Bedroom is east of the Living Room.	[<i>In italiano:</i> The <i>camera</i> is east of the <i>soggiorno</i> .]
The Bathroom is east of the Kitchen.	[<i>In italiano:</i> The <i>bagno</i> is east of the <i>cucina</i> .]

The floating button holder is a backdrop. It is everywhere. It is not scenery.
[*In italiano:* The *tastiera fluttuante* is a backdrop. It is everywhere. It is not scenery.]

Rule for writing a paragraph about the floating button holder:
say "A button holder is floating in mid-air here. On it are a red button and a green button."

[*In italiano:*
Rule for writing a paragraph about the *tastiera fluttuante*:
say "C'è una *tastiera fluttuante* a mezz'aria, con un *pulsante rosso* e uno *blu*."]]

The red button is part of the floating button holder. The green button is part of the floating button holder.
[*In italiano:* The *pulsante rosso* is part of the *tastiera fluttuante*. The *pulsante verde* is part of the *tastiera fluttuante*.]

Instead of pushing the red button:

change the south exit of the Living Room to nowhere;
change the north exit of the Kitchen to nowhere;
change the south exit of the Bedroom to the Bathroom;
change the north exit of the Bathroom to the Bedroom;
say "Living Room south is gone, Bedroom south is open."

[In italiano:

Instead of pushing the *pulsante rosso*:

change the south exit of the *soggiorno* to nowhere;
change the north exit of the *cucina* to nowhere;
change the south exit of the *camera* to the *bagno*;
change the north exit of the *bagno* to the *camera*;
say "*Il soggiorno a sud è sparito, la camera si apre a sud.*"

Instead of pushing the green button:

change the south exit of the Living Room to the Kitchen;
change the north exit of the Kitchen to the Living Room;
change the south exit of the Bedroom to nowhere;
change the north exit of the Bathroom to nowhere;
say "Living Room south is open, Bedroom south is gone."

[In italiano:

Instead of pushing the *pulsante verde*:

change the south exit of the *soggiorno* to the *cucina*;
change the north exit of the *cucina* to the *soggiorno*;
change the south exit of the *camera* to nowhere;
change the north exit of the *bagno* to nowhere;
say "*Il soggiorno si apre a sud, la camera è sparita.*"

Sfortunatamente, questo piccolo trucco non funziona con le porte. Inform ha idee piuttosto rigide su come funzionano le porte. E' possibile fare in modo che una porta sparisca durante il gioco, anche se è ancora presente. Farlo è difficile e può causare errori, ma il posto dove iniziare per capire come fare è l'estensione "Secret Doors", che è discussa più avanti in questo capitolo. Un approccio migliore potrebbe essere, come già è stato detto, quello di creare un oggetto ordinario che risponde ai comandi del giocatore come se fosse una porta.

Regioni

Come spiegato nel **par. 3.4** della Documentazione, "Regions and the index map" (le regioni e l'indice della mappa), utilizzare le regioni è un modo elegante per organizzare una mappa molto grande. Dopo aver definito alcune regioni, potrai usare più facilmente alcune delle funzionalità di Inform relative alla costruzione del mondo. Questo perché puoi verificare in quale regione si trovi il giocatore ed utilizzare quell'informazione. Per esempio, dopo aver creato una regione chiamata Forest, puoi scrivere:

Every turn when in the Forest:

say "[one of]A bird chirps.[or]You hear a soft rustling in the bushes.[or]A butterfly flits past you.
[at random]".

[In italiano:

Every turn when in the Foresta, say "[one of]Un uccellino cinguetta.[or]Odi un lieve fruscio tra i cespugli.[or]Una farfalla ti volteggia accanto.[at random]".]

(Gli elementi inseriti tra parentesi quadre e racchiusi tra le virgolette permettono ad Inform di produrre testo che cambia. Per maggiori dettagli su questo argomento, vedi “Inserimenti nel testo” nel Capitolo 8 della *Guida*). Creare una regione e assegnarle stanze è semplice. Puoi fare così:

Forest is a region. Forest Path, Haunted Grove, and Canyon View are in Forest.

[In italiano:

Foresta is a region. Sentiero nella foresta, radura stregata and vista sul canyon are in Foresta.]

Ho notato che è più sicuro definire una regione *dopo* aver creato le stanze che la compongono — cioè, la definizione della regione deve seguire le stanze nel codice sorgente. Questo perché le stanze sono create appena sono citate nel codice. Inform sa che le uniche cose che si possono trovare in una regione sono le stanze, così creerà una nuova stanza quando si crea una regione, se non conosce già quella stanza. Alcuni modi di scrivere frasi che creano stanze confondono Inform se già conosce una stanza che ha lo stesso nome. Inoltre, devi essere sicuro di usare esattamente lo stesso nome nella lista della regione che hai usato per creare la stanza. Se c'è un errore di battitura, Inform creerà una seconda stanza senza funzionalità il cui nome contiene un errore. Tutto ciò può portare a bug difficili da trovare.

Per uniformità e rendere il codice facile da leggere, suggerisco di usare sempre la parola “Area” nel nome delle tue regioni. Così io modificarei il codice scritto sopra in questo modo:

Forest Area is a region. Forest Path, Haunted Grove, and Canyon View are in Forest Area.

Puoi voler creare una regione che si trova completamente all'interno di un'altra regione, ricordando che Inform non permette che due regioni si intersechino, ma consente l'inclusione completa. Quando crei una regione che si trova all'interno di un'altra, è importante menzionare le stanze in una sola delle due definizioni. Il codice che segue non funziona:

Room1 is a room. Room2 is a room. Room3 is a room. Room4 is a room. Room5 is a room.
The Big Area is a region. Room1, Room2, Room3, Room4, and Room5 are in the Big Area.
The Little Area is a region. Room1, Room2, and Room3 are in the Little Area. [Error!]

Ecco come ottenere il risultato desiderato:

Room1 is a room. Room2 is a room. Room3 is a room. Room4 is a room. Room5 is a room.
The Little Area is a region. Room1, Room2, and Room3 are in the Little Area.
The Big Area is a region. The Little Area, Room4, and Room5 are in the Big Area.

Da notare che “Little Area” è definito prima di “Big Area”. In questo modo possiamo utilizzare “Little Area” nella definizione di regione di “Big Area”. Se proviamo il contrario, Inform pensa che “Little Area” sia una stanza — perché qualunque cosa si trovi in una regione viene considerata, in assenza di altre specificazioni, una stanza. Così si confonderà quando diremo che “Little Area” è una regione, mentre funzionerà se opereremo come mostrato sopra.

Sfondi

Le stanze in Inform sono fatte... di nulla — sono semplicemente vuoti contenitori senza caratteristiche in cui si possono far entrare ed uscire oggetti. In particolare, ti potresti aspettare che una stanza abbia un soffitto, dei muri e un pavimento. Se provi i comandi X CEILING, X FLOOR o X WALL (x soffitto, x pavimento o x muro) nel tuo gioco, ti sarà risposto: “You can’t see any such thing” (non puoi vedere nulla di simile). Allo stesso modo, in un ambiente esterno, X SKY e X GROUND (x cielo e x terreno) non funzionerà. Molti giocatori potrebbero accettare questa convenzione e non penserebbero neppure di interagire con i muri, il soffitto ed il pavimento. Ma se citi un muro, il soffitto, il pavimento, il cielo o il terreno nella descrizione della stanza, crearli come scenario potrebbe essere la cosa migliore da fare.

Gli scenari sono sempre in una sola stanza, ma è disponibile anche uno scenario chiamato *backdrop* (sfondo), che è un oggetto insolito perché si può trovare in luoghi diversi nello stesso momento.

Un modo semplice per aggiungere un po' di realismo al gioco è quello di usare gli sfondi per creare il cielo, il terreno, il soffitto o il pavimento, solo quando sono necessari:

The sky is a backdrop. "A clear and cloudless blue." The sky is in the Forest Area.

The ground is a backdrop. "It's rather dirty." Understand "dirt" as the ground. The ground is in the Forest Area.

[*In italiano:* The *cielo* is a backdrop. "Un cielo blu, chiaro e senza nubi." The *cielo* is in the area della foresta. The *terreno* is a backdrop. "E' piuttosto sporco." Understand "sporco" as the *terreno*. The *terreno* is in the area della foresta.]

Ora se il giocatore scrive X SKY (x cielo) in qualsiasi stanza nell'area della foresta, il gioco risponderà: “Un cielo blu, chiaro e senza nubi”. Un miglioramento notevole! (Nota che “The description is” non è richiesto nelle descrizioni degli sfondi). Quando creiamo stanze al chiuso, ad esempio in un castello, non si troveranno nell'area della foresta, così se un giocatore prova X SKY mentre è nel castello, otterrà la risposta appropriata: “You can’t see any such thing” (non puoi vedere nulla di simile).

Gli sfondi sono più versatili di quanto si possa credere. Uno sfondo può essere usato, ad esempio, per creare un fiume presente in più stanze. In un gioco avevo bisogno di una vetrina (un supporto — vedi pag. 79) che potesse essere raggiunto sia all'interno che all'esterno di una stanza. Su suggerimento di Emily Short, ho reato la finestra come uno sfondo, così poteva trovarsi in due stanze nello stesso momento e poi ho creato la vetrina come parte della finestra.

Rimuovere uno sfondo

Rimuovere completamente uno sfondo durante il gioco è facile. Se abbiamo creato, ad esempio, della spessa nebbia, quando il giocatore compie un'azione che faccia soffiare il vento, possiamo scrivere:

remove the fog from play; [*In italiano:* remove the *nebbia* from play;]

Rimuovere uno sfondo da certe stanze, ma lasciarlo in altre è, comunque, un po' più difficile. Inform non ha comandi dedicati per farlo, ma possiamo creare una routine che lo faccia. Ciò che dobbiamo fare è attribuire una proprietà a tutte le stanze dove si può trovare lo sfondo. Poiché parliamo ancora di nebbia, le stanze saranno foggy (nebbiose) o non foggy. Forniamo al giocatore anche ventagli giganti. Agitando i ventagli la nebbia si disperde, ma solo nella stanza in cui vengono agitati.

Include Plurality by Emily Short. [This is so we can write a more streamlined Check rule for our new action.]

A room can be foggy or not foggy. A room is usually not foggy.

The Desolate Moor is a room. "A gloomy treeless waste stretches out on all sides[if the fog is in the Moor], but you can't see very far, because the fog is closing in[otherwise]. You can see a path that extends north and south from here[end if]." The Desolate Moor is foggy.

The Haunted Grove is north of the Desolate Moor. "Thin, widely spaced trees of a mournful character surround you[if the fog is in the Grove]. It's difficult to see where you might go from here, because the fog presses close among the trees[otherwise]. A path leads south out of the grove[end if]." The Haunted Grove is foggy.

The Bog is south of the Desolate Moor. "The ground here is quite moist[if the fog is in the Bog], and the fog is thicker[otherwise]. A path extends out of the bog to the north[end if]." The Bog is foggy.

The thick gray fog is a backdrop. The description is "Tendrils of gray fog drift across the land."
When play begins:
 move the fog backdrop to all foggy rooms.

The player carries some giant bellows. The indefinite article of the giant bellows is "a".

Pumping is an action applying to one thing. Understand "pump [something]" as pumping.
Check pumping:
 say "[The noun] [is-are] not something you can pump."

Instead of pumping the giant bellows:
 if the location is foggy:
 say "As you pump the bellows with great vigor, the fog blows away!";
 otherwise:
 say "You've already dispelled the fog here.";
 now the location is not foggy;
 update backdrop positions.

L'elemento importante in questo esempio è la regola "Instead". Rende il luogo non nebbioso e poi aggiorna le posizioni dello sfondo (vedi il **par. 8.8** nella Documentazione, "Moving backdrops", cioè dedicato allo spostamento degli sfondi).

Stanze buie

La prima vera avventura testuale, un gioco chiamato “Adventure”, si svolgeva in una caverna. Come molte caverne, era buia. Se il giocatore trasportava una lanterna, si poteva giocare normalmente, mentre in ogni stanza dove non era presente una sorgente di luce ci sarebbe stato buio pesto, con le ovvie difficoltà del caso.

In una stanza buia il giocatore non può esaminare nulla, non è in grado di vedere cosa c'è sul pavimento, non è in grado di usare LOOK per leggere la descrizione della stanza, probabilmente non può leggere ciò che c'è scritto sui libri o sui pezzi di carta e non può trovare un'uscita per tornare ad una stanza illuminata. Dovresti ancora essere in grado di ispezionare l'inventario in una stanza buia (a seconda di come è scritto il gioco) e potresti essere in grado di toccare gli oggetti.

Oggi giorno, le stanze di molti giochi sono illuminate, così non c'è bisogno di lanterne. Le stanze in Inform sono sempre illuminate a meno che non si specifichi il contrario.

Il **par. 3.15** della Documentazione, “Light and darkness” (luce ed oscurità), spiega come creare una stanza buia e come creare oggetti (come le torce) che la possono illuminare. E' una convezione dell'IF che le stanze buie siano completamente buie, ma l'esempio 322, “Zorn of Zorna”, spiega come creare un sistema in cui la quantità di luce in una stanza può essere bassa.

Una stanza buia in cui devi portare una sorgente di luce è un tipo standard di puzzle che molti giocatori non avranno problemi a risolvere. Dovresti considerare qualche variazione per rendere più difficile l'uso o il ritrovamento di una sorgente di luce. Due idee ovvie sono la torcia elettrica per la quale occorre trovare delle batterie e la candela che si spegne a causa del vento. Tuttavia, entrambe sono state usate in molti giochi. Per creare un puzzle della stanza buia veramente originale, devi trovare qualcosa impiegando più immaginazione — forse una giara di vetro piena di lucciole, una leva segreta che apre un lucernario, o un fantasma amico che risplende nel buio e deve essere convinto a seguirti.

Per creare un oggetto in Inform che fornisce luce, tutto ciò che devi fare è dire:

`now the torch is lit;`

[In italiano: `now the torcia is lit;`]

In un vero gioco, potresti volere che la torcia si spenga dopo un certo numero di mosse, ottenendo un puzzle temporizzato. Oppure trovare i fiammiferi con cui accendere la torcia potrebbe essere un puzzle distinto. Se l'oggetto è acceso sia dall'inizio del gioco, puoi scrivere:

`The ghost is a person. The ghost is lit.`

[In italiano: `The fantasma is a person. The fantasma is lit.`]

Stanze apparentemente identiche

Ogni tanto, vorrai creare un'area della mappa in cui ci sono diverse stanze con lo stesso nome e la stessa descrizione, come ad esempio: “Ti trovi in un labirinto fatto di piccoli passaggio tortuosi, tutti uguali”. Internamente, nel tuo codice, ciascuna stanza deve avere il suo nome, ma è facile assegnare a diverse stanze lo stesso nome stampato e la stessa descrizione:

Land1 is a room. "The landscape stretches out to the horizon on all sides." The printed name of Land1 is "Landscape".

Land2 is south of Land1. "The landscape stretches out to the horizon on all sides." The printed name of Land2 is "Landscape".

Land3 is west of Land1. "The landscape stretches out to the horizon on all sides." The printed name of Land3 is "Landscape".

... e così via. Per il giocatore, tutte queste stanze sembreranno inizialmente identiche. Ma come nel gioco “Adventure”, da cui è presa in prestito la sequenza di piccoli passaggi tortuosi tutti uguali, il giocatore può facilmente lasciare oggetti diversi in ogni stanza e successivamente disegnare una mappa basata sui luoghi dei diversi oggetti. Quando io stesso progetto un puzzle di questo tipo (che non sopporto), cerco una ragione plausibile perché il personaggio del giocatore non voglia lasciare nulla, come: “Meglio di no. Potresti non ritrovarlo più”. Questo almeno forza il giocatore a cercare un approccio originale al problema della mappatura. Non è un puzzle molto originale, ma potresti volerlo provare, così valeva la pena descriverlo.

Appendice B: Licenza

L'OPERA (COME SOTTO DEFINITA) È MESSA A DISPOSIZIONE SULLA BASE DEI TERMINI DELLA PRESENTE LICENZA "CREATIVE COMMONS PUBLIC LICENCE" ("CCPL" O "LICENZA"). L'OPERA È PROTETTA DAL DIRITTO D'AUTORE E/O DALLE ALTRE LEGGI APPLICABILI. OGNI UTILIZZAZIONE DELL'OPERA CHE NON SIA AUTORIZZATA AI SENSI DELLA PRESENTE LICENZA O DEL DIRITTO D'AUTORE È PROIBITA.

CON IL SEMPLICE ESERCIZIO SULL'OPERA DI UNO QUALUNQUE DEI DIRITTI QUI DI SEGUITO ELENCATI, TU ACCETTI E TI OBBLIGHI A RISPETTARE INTEGRALMENTE I TERMINI DELLA PRESENTE LICENZA AI SENSI DEL PUNTO 8.f. IL LICENZIANTE CONCEDE A TE I DIRITTI QUI DI SEGUITO ELENCATI A CONDIZIONE CHE TU ACCETTI DI RISPETTARE I TERMINI E LE CONDIZIONI DI CUI ALLA PRESENTE LICENZA.

Creative Commons Public License

1. Definizioni. Ai fini e per gli effetti della presente licenza, si intende per

- a. **"Collezione di Opere"**, un'opera, come un numero di un periodico, un'antologia o un'enciclopedia, nella quale l'Opera nella sua interezza e forma originale, unitamente ad altri contributi costituenti loro stessi opere distinte ed autonome, sono raccolti in un'unità collettiva. Un'opera che costituisce Collezione di Opere non verrà considerata Opera Derivata (come sotto definita) ai fini della presente Licenza;
- b. **"Opera Derivata"**, un'opera basata sull'Opera ovvero sull'Opera insieme con altre opere preesistenti, come una traduzione, un arrangiamento musicale, un adattamento teatrale, narrativo, cinematografico, una registrazione di suoni, una riproduzione d'arte, un digesto, una sintesi, o ogni altra forma in cui l'Opera possa essere riproposta, trasformata o adattata. Nel caso in cui un'Opera tra quelle qui descritte costituisca già Collezione di Opere, essa non sarà considerata Opera Derivata ai fini della presente Licenza. Al fine di evitare dubbi è inteso che, quando l'Opera sia una composizione musicale o registrazione di suoni, la sincronizzazione dell'Opera in relazione con un'immagine in movimento ("synching") sarà considerata Opera Derivata ai fini di questa Licenza;
- c. **"Licenziante"**, l'individuo o l'ente che offre l'Opera secondo i termini e le condizioni della presente Licenza;
- d. **"Autore Originario"**, il soggetto che ha creato l'Opera;
- e. **"Opera"**, l'opera dell'ingegno suscettibile di protezione in forza delle leggi sul diritto d'autore, la cui utilizzazione è offerta nel rispetto dei termini della presente Licenza;
- f. **"Tu"/"Te"**, l'individuo o l'ente che esercita i diritti derivanti dalla presente Licenza e che non abbia precedentemente violato i termini della presente Licenza relativi all'Opera, o che, nonostante una precedente violazione degli stessi, abbia ricevuto espressa autorizzazione dal Licenziante all'esercizio dei diritti derivanti dalla presente Licenza;
- g. **"Elementi della Licenza"**, gli attributi fondamentali della Licenza scelti dal Licenziante ed indicati nel titolo della Licenza: Attribuzione, Non Commerciale, Condividi allo stesso modo.

2. Libere utilizzazioni. La presente Licenza non intende in alcun modo ridurre, limitare o restringere alcun diritto di libera utilizzazione o l'operare della regola dell'esaurimento del diritto o altre limitazioni dei diritti esclusivi sull'Opera derivanti dalla legge sul diritto d'autore o da altre leggi applicabili.

3. Concessione della Licenza. Nel rispetto dei termini e delle condizioni contenute nella presente Licenza, il Licenziante concede a Te una licenza per tutto il mondo, gratuita, non esclusiva e perpetua (per la durata del diritto d'autore applicabile) che autorizza ad esercitare i diritti sull'Opera qui di seguito elencati:

- a. riproduzione dell'Opera, incorporazione dell'Opera in una o più Collezioni di Opere e riproduzione dell'Opera come incorporata nelle Collezioni di Opere;
- b. creazione e riproduzione di un'Opera Derivata;
- c. distribuzione di copie dell'Opera o di supporti fonografici su cui l'Opera è registrata, comunicazione al pubblico, rappresentazione, esecuzione, recitazione o esposizione in pubblico, ivi inclusa la trasmissione audio digitale dell'Opera, e ciò anche quando l'Opera sia incorporata in Collezioni di Opere;
- d. distribuzione di copie dell'Opera o di supporti fonografici su cui l'Opera Derivata è registrata, comunicazione al

pubblico, rappresentazione, esecuzione, recitazione o esposizione in pubblico, ivi inclusa la trasmissione audio digitale di Opere Derivate.

I diritti sopra descritti potranno essere esercitati con ogni mezzo di comunicazione e in tutti i formati. Tra i diritti di cui sopra si intende compreso il diritto di apportare all'Opera le modifiche che si rendessero tecnicamente necessarie per l'esercizio di detti diritti tramite altri mezzi di comunicazione o su altri formati. Tutti i diritti non espressamente concessi dal Licenziante rimangono riservati, ivi inclusi quelli di cui ai punti 4(e) e (f).

4. Restrizioni. La Licenza concessa in conformità al precedente punto 3 è espressamente assoggettata a, e limitata da, le seguenti restrizioni:

- a. Tu puoi distribuire, comunicare al pubblico, rappresentare, eseguire, recitare o esporre in pubblico l'Opera, anche in forma digitale, solo assicurando che i termini di cui alla presente Licenza siano rispettati e, insieme ad ogni copia dell'Opera (o supporto fonografico su cui è registrata l'Opera) che distribuisce, comunichi al pubblico o rappresenti, esegui, reciti o esponi in pubblico, anche in forma digitale, devi includere una copia della presente Licenza o il suo Uniform Resource Identifier. Non puoi proporre o imporre alcuna condizione relativa all'Opera che alteri o restringa i termini della presente Licenza o l'esercizio da parte del beneficiario dei diritti qui concessi. Non puoi concedere l'Opera in sublicenza. Devi mantenere intatte tutte le informative che si riferiscono alla presente Licenza ed all'esclusione delle garanzie. Non puoi distribuire, comunicare al pubblico, rappresentare, eseguire, recitare o esporre in pubblico l'Opera, neanche in forma digitale, usando misure tecnologiche miranti a controllare l'accesso all'Opera ovvero l'uso dell'Opera, in maniera incompatibile con i termini della presente Licenza. Quanto sopra si applica all'Opera anche quando questa faccia parte di una Collezione di Opere, anche se ciò non comporta che la Collezione di Opere di per sé ed indipendentemente dall'Opera stessa debba essere soggetta ai termini ed alle condizioni della presente Licenza. Qualora Tu crei una Collezione di Opere, su richiesta di qualsiasi Licenziante, devi rimuovere dalla Collezione di Opere stessa, ove materialmente possibile, ogni riferimento in accordo con quanto previsto dalla clausola 4.d, come da richiesta. Qualora tu crei un'Opera Derivata, su richiesta di qualsiasi Licenziante devi rimuovere dall'Opera Derivata stessa, nella misura in cui ciò sia possibile, ogni riferimento in accordo con quanto previsto dalla clausola 4.c, come da richiesta.
- b. Tu puoi distribuire, comunicare al pubblico, rappresentare, eseguire, recitare o esporre in pubblico un'Opera Derivata, anche in forma digitale, solo assicurando che siano rispettati i termini di cui alla presente Licenza, di una versione successiva di questa Licenza con gli stessi Elementi della Licenza come questa Licenza o di una licenza Creative Commons iCommons che contenga gli stessi Elementi della Licenza come questa Licenza (ad es. Attribuzione-NonCommerciale-Condividi allo stesso modo 2.5. Giappone). Insieme ad ogni copia dell'Opera Derivata (o supporto fonografico su cui è registrata l'Opera Derivata) che distribuisce, comunichi al pubblico o rappresenti, esegui, reciti o esponi in pubblico, anche in forma digitale, devi includere una copia della presente Licenza o dell'altra Licenza di cui alla frase precedente o il suo Uniform Resource Identifier. Non puoi proporre o imporre alcuna condizione relativa all'Opera Derivata che alteri o restringa i termini della presente Licenza o l'esercizio da parte del beneficiario dei diritti qui concessi e devi mantenere intatte tutte le informative che si riferiscono alla presente Licenza ed all'esclusione delle garanzie. Non puoi distribuire, comunicare al pubblico, rappresentare, eseguire, recitare o esporre in pubblico l'Opera Derivata, neanche in forma digitale, provvista di misure tecnologiche miranti a controllare l'accesso all'Opera ovvero l'uso dell'Opera, in maniera incompatibile con i termini della presente Licenza. Quanto sopra si applica all'Opera Derivata anche quando questa faccia parte di una Collezione di Opere, ma ciò non comporta che la Collezione di Opere di per sé ed indipendentemente dall'Opera Derivata debba esser soggetta ai termini ed alle condizioni della presente Licenza.
- c. Tu non puoi esercitare alcuno dei diritti a Te concessi al precedente punto 3 in una maniera tale che sia prevalentemente intesa o diretta al perseguimento di un vantaggio commerciale o di un compenso monetario privato. Lo scambio dell'Opera con altre opere protette dal diritto d'autore, per mezzo della condivisione di file digitali (c.d. filesharing) o altrimenti, non è considerato inteso o diretto a perseguire un vantaggio commerciale o un compenso monetario privato, a patto che non ci sia alcun pagamento di alcun compenso monetario in connessione allo scambio di opere coperte da diritto d'autore.
- d. Qualora Tu distribuisca, comunichi al pubblico, rappresenti, esegua, reciti o esponga in pubblico, anche in forma digitale, l'Opera o qualsiasi Opera Derivata o Collezione di Opere, devi mantenere intatte tutte le informative sul diritto d'autore sull'Opera. Devi riconoscere una menzione adeguata rispetto al mezzo di comunicazione o supporto che utilizzi: (i) all'Autore Originale (citando il suo nome o lo pseudonimo, se del caso), ove fornito; e/o (ii) alle terze parti designate, se l'Autore Originale e/o il Licenziante hanno designato una o più terze parti (ad

esempio, una istituzione finanziatrice, un ente editoriale) per l'attribuzione nell'informativa sul diritto d'autore del Licenziante o nei termini di servizio o con altri mezzi ragionevoli; nella misura in cui sia ragionevolmente possibile, l'Uniform Resource Identifier, che il Licenziante specifichi dover essere associato con l'Opera, salvo che tale URI non faccia alcun riferimento alla informazione di protezione di diritto d'autore o non dia informazioni sulla licenza dell'Opera; inoltre, in caso di Opera Derivata, devi menzionare l'uso dell'Opera nell'Opera Derivata (ad esempio, "traduzione francese dell'Opera dell'Autore Originario", o "sceneggiatura basata sull'Opera originaria dell'Autore Originario"). Tale menzione deve essere realizzata in qualsiasi maniera ragionevole possibile; in ogni caso, in ipotesi di Opera Derivata o Collezione di Opere, tale menzione deve quantomeno essere posta nel medesimo punto dove viene indicato il nome di altri autori di rilevanza paragonabile e con lo stesso risalto concesso alla menzione di altri autori di rilevanza paragonabile.

- e. Al fine di evitare dubbi è inteso che, se l'Opera sia di tipo musicale
- i. **Compensi per la comunicazione al pubblico o la rappresentazione o esecuzione di opere incluse in repertori.** Il Licenziante si riserva il diritto esclusivo di riscuotere compensi, personalmente o per il tramite di un ente di gestione collettiva (ad es. SIAE), per la comunicazione al pubblico o la rappresentazione o esecuzione, anche in forma digitale (ad es. tramite webcast) dell'Opera, se tale utilizzazione sia prevalentemente intesa o diretta a perseguire un vantaggio commerciale o un compenso monetario privato.
 - ii. **Compensi per versioni cover.** Il Licenziante si riserva il diritto esclusivo di riscuotere compensi, personalmente o per il tramite di un ente di gestione collettiva (ad es. SIAE), per ogni disco che Tu crei e distribuisi a partire dall'Opera (versione cover), nel caso in cui la Tua distribuzione di detta versione cover sia prevalentemente intesa o diretta a perseguire un vantaggio commerciale o un compenso monetario privato.
- f. **Compensi per la comunicazione al pubblico dell'Opera mediante fonogrammi.** Al fine di evitare dubbi, è inteso che se l'Opera è una registrazione di suoni, il Licenziante si riserva il diritto esclusivo di riscuotere compensi, personalmente o per il tramite di un ente di gestione collettiva (ad es. IMAIE), per la comunicazione al pubblico dell'Opera, anche in forma digitale, nel caso in cui la Tua comunicazione al pubblico sia prevalentemente intesa o diretta a perseguire un vantaggio commerciale o un compenso monetario privato.
- g. **Altri compensi previsti dalla legge italiana.** Al fine di evitare dubbi, è inteso che il Licenziante si riserva il diritto esclusivo di riscuotere i compensi a lui attribuiti dalla legge italiana sul diritto d'autore (ad es. per l'inserimento dell'Opera in un'antologia ad uso scolastico ex art. 70 l. 633/1941), personalmente o per tramite di un ente di gestione collettiva (ad es. SIAE, IMAIE), se l'utilizzazione dell'Opera sia prevalentemente intesa o diretta a perseguire un vantaggio commerciale o un compenso monetario privato. Al Licenziante spettano in ogni caso i compensi irrinunciabili a lui attribuiti dalla medesima legge (ad es. l'equo compenso spettante all'autore di opere musicali, cinematografiche, audiovisive o di sequenze di immagini in movimento nel caso di noleggio ai sensi dell'art. 18-bis l. 633/1941).

5. Dichiarazioni, Garanzie ed Esonero da responsabilità

SALVO CHE SIA ESPRESSAMENTE CONVENUTO ALTRIMENTI PER ISCRITTO FRA LE PARTI, IL LICENZIANTE OFFRE L'OPERA IN LICENZA "COSÌ COM'È" E NON FORNISCE ALCUNA DICHIARAZIONE O GARANZIA DI QUALSIASI TIPO CON RIGUARDO ALL'OPERA, SIA ESSA ESPRESSA OD IMPLICITA, DI FONTE LEGALE O DI ALTRO TIPO, ESSENDO QUINDI ESCLUSE, FRA LE ALTRE, LE GARANZIE RELATIVE AL TITOLO, ALLA COMMERCIALIZZABILITÀ, ALL'IDONEITÀ PER UN FINE SPECIFICO E ALLA NON VIOLAZIONE DI DIRITTI DI TERZI O ALLA MANCANZA DI DIFETTI LATENTI O DI ALTRO TIPO, ALL'ESATTEZZA OD ALLA PRESENZA DI ERRORI, SIANO ESSI ACCERTABILI O MENO. ALCUNE GIURISDIZIONI NON CONSENTONO L'ESCLUSIONE DI GARANZIE IMPLICITE E QUINDI TALE ESCLUSIONE PUÒ NON APPLICARSI A TE.

6. Limitazione di Responsabilità. SALVI I LIMITI STABILITI DALLA LEGGE APPLICABILE, IL LICENZIANTE NON SARÀ IN ALCUN CASO RESPONSABILE NEI TUOI CONFRONTI A QUALUNQUE TITOLO PER ALCUN TIPO DI DANNO, SIA ESSO SPECIALE, INCIDENTALI, CONSEGUENZIALE, PUNITIVO OD ESEMPLARE, DERIVANTE DALLA PRESENTE LICENZA O DALL'USO DELL'OPERA, ANCHE NEL CASO IN CUI IL LICENZIANTE SIA STATO EDOTTO SULLA POSSIBILITÀ DI TALI DANNI. NESSUNA CLAUSOLA DI QUESTA LICENZA ESCLUDE O LIMITA LA RESPONSABILITÀ NEL CASO IN CUI QUESTA DIPENDA DA DOLO O COLPA GRAVE.

7. Risoluzione

- a. La presente Licenza si intenderà risolta di diritto e i diritti con essa concessi cesseranno automaticamente, senza necessità di alcuna comunicazione in tal senso da parte del Licenziante, in caso di qualsivoglia inadempimento dei termini della presente Licenza da parte Tua, ed in particolare delle disposizioni di cui ai punti 4.a, 4.b, 4.c e 4.d, essendo la presente Licenza condizionata risolutivamente al verificarsi di tali inadempimenti. In ogni caso, la risoluzione della presente Licenza non pregiudicherà i diritti acquistati da individui o enti che abbiano acquistato da Te Opere Derivate o Collezioni di Opere, ai sensi della presente Licenza, a condizione che tali individui o enti continuino a rispettare integralmente le licenze di cui sono parte. Le sezioni 1, 2, 5, 6, 7 e 8 rimangono valide in presenza di qualsiasi risoluzione della presente Licenza.
- b. Sempre che vengano rispettati i termini e le condizioni di cui sopra, la presente Licenza è perpetua (e concessa per tutta la durata del diritto d'autore sull'Opera applicabile). Nonostante ciò, il Licenziante si riserva il diritto di rilasciare l'Opera sulla base dei termini di una differente licenza o di cessare la distribuzione dell'Opera in qualsiasi momento; fermo restando che, in ogni caso, tali decisioni non comporteranno recesso dalla presente Licenza (o da qualsiasi altra licenza che sia stata concessa, o che sia richiesto che venga concessa, ai termini della presente Licenza), e la presente Licenza continuerà ad avere piena efficacia, salvo che vi sia risoluzione come sopra indicato.

8. Varie

- a. Ogni volta che Tu distribuisi, o rappresenti, esegui o reciti pubblicamente in forma digitale l'Opera o una Collezione di Opere, il Licenziante offre al destinatario una licenza per l'Opera nei medesimi termini e condizioni che a Te sono stati concessi dalla presente Licenza.
- b. Ogni volta che Tu distribuisi, o rappresenti, esegui o reciti pubblicamente in forma digitale un'Opera Derivata, il Licenziante offre al destinatario una licenza per l'Opera originale nei medesimi termini e condizioni che a Te sono stati concessi dalla presente Licenza.
- c. L'invalidità o l'inefficacia, secondo la legge applicabile, di una o più fra le disposizioni della presente Licenza, non comporterà l'invalidità o l'inefficacia dei restanti termini e, senza bisogno di ulteriori azioni delle parti, le disposizioni invalide o inefficaci saranno da intendersi rettificcate nei limiti della misura che sia indispensabile per renderle valide ed efficaci.
- d. In nessun caso i termini e le disposizioni di cui alla presente Licenza possono essere considerati rinunciati, né alcuna violazione può essere considerata consentita, salvo che tale rinuncia o consenso risultino per iscritto da una dichiarazione firmata dalla parte contro cui operi tale rinuncia o consenso.
- e. La presente Licenza costituisce l'intero accordo tra le parti relativamente all'Opera qui data in licenza. Non esistono altre intese, accordi o dichiarazioni relative all'Opera che non siano quelle qui specificate. Il Licenziante non sarà vincolato ad alcuna altra disposizione addizionale che possa apparire in alcuna comunicazione da Te proveniente. La presente Licenza non può essere modificata senza il mutuo consenso scritto del Licenziante e Tuo.
- f. **Clausola iCommons.** Questa Licenza trova applicazione nel caso in cui l'Opera sia utilizzata in Italia. Ove questo sia il caso, si applica anche il diritto d'autore italiano. Negli altri casi le parti si obbligano a rispettare i termini dell'attuale Licenza Creative Commons generica che corrisponde a questa Licenza Creative Commons iCommons.