

HOLLOW VOICE

Do you play adventure games by yourself, or as part of a group? I'm used to considering it a solitary activity, so it was a real treat to get together in early January—on the first day of the Great Blizzard of '96, no less!—with several other IF fans in the New York City area. (For those who couldn't make it because of the weather, yes, you definitely had a valid excuse!) Since surfing the Web, playing sneak previews of upcoming games, drinking coffee, and eating chocolate chip cookies are four of my favorite things (not necessarily in that order), I naturally had a great time. I hope to organize another such get-together in the near-term.

I also wanted to give a quick mention to a book that came across my desk recently called *Interactive Writer's Handbook*, by Darryl Wimberley and John Samsel (Los Angeles: The Carronade Group, 1995). Targeting writers, producers, and designers interested in learning about writing interactive screenplays, the book mainly addresses the creation of CD-ROM games and other multimedia entertainment but I think that text adventure game developers would draw inspiration from it too. Among other notable interviews, the book includes a question-and-answer session with Michele Em, screenwriter for *Return to Zork* that includes some telling details about how that project was pieced together. For example, she notes she had five or six weeks to write the dialogue—including deciding where and how to place puzzle clues—for 28 characters (the game had already been designed by this point). Raising the level of ridiculousness up a notch, *Return to Zork's* producer also wanted her to write the screenplay using Excel!

On the brighter side, Em notes that "I think that I got the job because I was a screenwriter who actually played games. Infocom games were my favorites. I had played the text version of *Zork* dozens of times." That should be some consolation for those who've wondered aloud on the rec.arts.int-fiction newsgroup if their love of text adventures could ever pay off for them financially. :)

Until next issue, happy gaming!

Eileen Mullin
eileen@interport.net

Contents:

Newsbriefs/Top 10
IF Web Sites2

Letters3

Tales From the
Code Front: The
Hard Part is Making
It Look Easy.....5

IF in 1995: The Year
in Review7

Article: Adventure
Game Interfaces11

Sneak Previews16

More Tales From the
Code Front:
Rexx-Adventure17

Game Review:
Lethe Flow
Phoenix.....19

What's on the Disk....20

XYZZYnews is published bimonthly by Bran Muffin Communications, 160 West 24th Street, # 7C, New York, NY 10011, USA. Email: eileen@interport.net. Send all inquiries, letters, and submissions to the address above.

Contents © 1996 XYZZYnews. All rights reserved. Published in the United States of America.

Electronic versions: There are currently three versions of XYZZYnews made available online. One is in ASCII and can be viewed with any text reader. You can also download a .PDF file that mirrors the layout of the print version. Use the Adobe Acrobat Reader (available for Windows, Mac, DOS and UNIX) to view the .PDF file; no special fonts or linked graphics are needed. You can obtain Acrobat Reader from ftp.adobe.com in the pub/adobe/applications/Acrobat folder, or <http://www.adobe.com/Software.html>. Thirdly, you can also read this issue of XYZZYnews on the World Wide Web at <http://www.interport.net/~eileen/xyzy.7.html>

Subscriptions: Both electronic versions are available at no cost. You can obtain either one by FTPing to the ftp.gmd.de/if-archive/magazines/XYZZYnews directory. To be added to the mailing list, please write to eileen@interport.net and specify text-only or .PDF version. The print version includes a 3.5" Mac or PC disk and is \$21 (U.S.) for one year (6 issues) or \$3.50 for a sample issue. For print subscriptions outside the U.S. or Canada, please email or write for rates.

All products, names, and services are trademarks or registered trademarks of their respective companies.

Editorial deadline for Issue #8 is February 29, 1996.

Editor:

Eileen Mullin

Contributors to this issue:

Stuart Beach
Phil Darnowsky
Mike DeSanto
C.E. Forman

January/February Top 10 Picks for IF on the World Wide Web

Adventure for the Web

<http://tjwww.stanford.edu/adventure.html>

Dan's Home Page (Lethe Flow Phoenix)

<http://weber.u.washington.edu/~scythe/lethe.html>

hints.infocom.many

<ftp://risc.ua.edu/pub/games/solutions/hints.infocom.many>

New Zork Times/The Status Line List

<http://wuarchive.wustl.edu/doc/misc/if-archive/infocom/articles/NZT%2bTSL.list>

Pages and Pages of MUDs

http://csugrad.cs.vt.edu/soc/mud_page.html

The Rise and Fall of Infocom

<http://speedracer.nmsu.edu/~jholder/if/ch3-if.html>

Text Adventures— Direct Downloadable Games

<http://www.xs4all.nl/~gzbrn/DD/Textad.html>

Virtual Rooms, or a Spatial Metaphor

<http://www.well.com/user/mmcadams/units4.html>

You Found It

<http://bigdog.engr.arizona.edu/~scannelm/prize.html>

Zork Nemesis

<http://www.activision.com/zorknem/zorkhome.html>



Graphic from Zork Nemesis page on Activision's Web site.

Second Reader Survey Results
In Issue #7, reader David Palmer posed a number of questions to follow up on our first reader survey. There were 28 responses to his questions from readers in seven countries (Australia, Denmark, Finland, Great Britain, Spain, Sweden, and USA). The leading responses are summarized below:

- What major are student readers of *XYZZYnews* pursuing? Almost half of the respondents said they were computer science majors. Other responses that came up more than once were math, physics, and archeology.
- Do *XYZZYnews* readers like reading in general, and what are the preferred genres and authors? Science fiction and fantasy were most frequently mentioned, as was fiction in general. The most frequently named favorite authors included Douglas Adams, Isaac Asimov, H.P. Lovecraft, and J.R.R. Tolkien.
- How many of the readers have written any IF? Five of the 28 said they have written an adventure, although all but one included self-disparaging comments about the games' quality! :) Interestingly, 20 said they would like to write a game or are in the middle of writing one.
- Do the readers like other forms of puzzles? The respondents were somewhat split on this question: about half said yes, with some distinguishing between math/logic puzzles and word puzzles, and the other half mainly responded with no or

pleaded a lack of time to play such puzzles.

- What number of IF games has the average reader finished? No consensus emerged here; responses ranged from none to "about 50-55 games."

To *XYZZYnews*:

Would you tell Doug Atkinson that I *loved* his "You Know You've Played Too Much Infocom When..." [Issue #7]

Reminds me of a running *MAD* magazine gag (only this time about something I love, IF, and frankly his were funnier than *MAD's* usually are).

Haven't read all of the issue yet, but I laughed my head off at his list, especially #33, about the afterlife and restoring, restarting or quitting and that you keep everything you own in the living room, have sugar cubes you've named, touch every mirror and look in it being prepared to be insulted...well I could go on and on. But that was priceless, absolutely priceless as good as *or better* than stuff in the old Status Line. Better I think.

—Marnie A. Parker
103005.1513@compuserve.com

To *XYZZYnews*:

In reply to two inquiries (from Finland and Denmark) that arose because of my letter being published in *XYZZYnews*, I have dug out the old floppy disks and (with some kind help) managed to get 32 additional AGT games uploaded to toftp.gmd.de/if-archive. These include a number of contest "honorable mentions" from the second

through seventh contests, as well as several very old games written using GAGS or the very first release of AGT.

By my count, there are now more than 80 games written using GAGS and AGT posted in the interactive fiction archive, about a third posted with complete source code.

The files are currently in ftp.gmd.de/incoming/if-archive but I have sent a message to blasius@gmd.de and I assume the files will shortly be transferred to the appropriate games and source directories.

—Mark J. Welch
i855741@trivalley.com

Dear *XYZZYnews*,

Frankly, I'm a little annoyed at all the AGT bashing I see going around. Here in *XYZZYnews*, the review of TimeSquared cites that its being a good game is amazing in light of its being AGT. On Baf's Guide to the IF archive, he lists the reasons a game could be poor as (and I paraphrase) "Being poorly written, having a picky parser, or being written in AGT."

AGT has produced many excellent games, Klaustrophobia, The Multi-Dimensional Thief, and others I won't even list. People keep on commenting on its "guess-the verb" games. All the included verbs are the most common ones (open, close, look, inventory, score, n, s, e, w, up, down, enter, exit: what else does one need?) but if common synonyms don't work, it's because the *designer* didn't put them in. Without the designer's influence, even the much-praised Inform

wouldn't know that a CD, compact disc, disc, disk, and round flat plastic thing were all the same object!

Despite the fact that AGT is much less powerful than Inform, it has one major power over it: It's easy to use. I have a genius-level IQ, can write in BASIC, and a little Pascal, but I can't seem to do much with Inform. (I even tried typing in an example verbatim from the manual, and it didn't work!) John Menichelli's modification to the source code in the form of AGT v1.83 is as much like the infocom format as inform (well, almost). It has a built-in undo feature, twice-a-turn metacommand processing, and loads of other features that would take more programming in Inform than I care to contemplate.

Thanks for letting me whine...

—L. Ross "Write Me"
Raszewski
rraszews@skipjack.bluecrab.org

Hi, Eileen!

I'm yet another old-time adventurer who's recently discovered the underground IF world. Have you considered providing a checklist to help us newcomers write our opening paragraphs? Something like:

1. Just stumbled onto XYZZY Web page (Y/N)
2. Am old-time Infocom fan (Y/N)
3. Have since rekindled old love thanks to blah blah blah (Y/N)
4. Am planning to try my hand at writing my own blah blah blah (Y/N)

This would probably save all of us a lot of time. <g>

Thanks for doing such a great job with this magazine!

—Kory Heath
103111.3267@compuserve.com

Infocom Bugs List Update

Here's some more info on Zork II bug #7. On the Apple II interpreter which came with Release 7, you get it. It also occurs with ZORK III, Release 15.

The second release of Zork I neatly avoids it by having "IT" refer to the mailbox initially.

Here's how the bug looks in Zork 2, Release 7 (UG3AU5), on the Apple II

```
>LOOK AT I T
I SEE NOTHING SPECIAL
ABOUT THE BUT
ZU.
```

Technical stuff: The bug is actually in the story files—the code attempts to get a property from object 0, which doesn't exist. This is undefined behavior.

—Matthew Russotto
russotto@pond.com

To XYZZYnews:

How embarrassing ... I tried to send this to Graeme Cree and cc it to eileen@interport.net, and managed to misspell the word "net" :-)

Here, as I promised, are some bug reports for Infocom's "Journey," release 26/890316:

While looking for the white stone (after having found the dwarf-, nymph- and elf stones), I sent Hurth down a hole, where he found a key and a skull. Although Esher was at the top of the pit, the descrip-

tion when asking him to examine the key still starts with "Esher took the key and looked it over." This seems even more strange if you consider that the pit was too deep for the rope to reach all the way down.

Earlier in this search, Bergon is wounded. It is possible to heal him, but out of curiosity, I wanted to see what would happen if I left him to die. On getting out of the caves, the party is trapped by some half-sentinent trees. Although Bergon was no longer in the party, he remarked darkly that he had seen those trees before (when looking for west-flake root, or whatever it was that could cure the bite of Nightfang).

To get to the Misty Isle, you have to use the 'wind' spell. If you haven't figured out the direction, the game will prompt you for if you really want to chance it. If you do, the party will perish. If you don't, you still use up one unit of air essence, even though the text says Praxix puts the essence back into his pouch.

On using the 'wind' spell to successfully get to the Misty Isle, Praxix is struck unconscious. Even though he is out cold, he still remarks that the spell used up the last of the air essence.

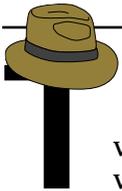
Dropping the rope while Praxix is unconscious still earns a reprimand from him.

That's all I can think of at the moment, but I only replayed parts of it. One of these days, I'll have to go back and play through the whole thing. If nothing else, Praxix's encounter with the talking tree would make it well worth it. :-)

—Torbjorn Andersson
d91tan@minsk.docs.uu.se

The Hardest Part is Making It Look Easy, or, Five Simple Steps to Coding a Stetson

by Phil Darnowsky (pdarnows@rwd.goucher.edu)



his vacation, I decided to amuse myself (and, hopefully, others) by writing my own text adventure. The hardest part, I found, wasn't thinking up a plot line, or writing puzzles. The hardest part was making my objects, the player, and the game world interact in a natural-seeming manner—in other words, making it look easy.

As it turned out, one of the most difficult objects to implement (as of this stage of coding, anyway) was a simple gray hat. Simple? Sure. Read on and see just how simple a hat can be to implement...

1. I get an idea for a puzzle: The player needs to cross a certain obstacle with their hands free, and yet transport a pack of cigarettes across it. The solution that occurs to me? Allow the player to put the cigarettes in their hat! Humming contentedly, I begin coding.
2. I give the hat the “container” attribute (Note: I program in Inform, but I'm sure TADS programmers have had similar experiences). I re-compile and check my inventory. Unfortunately, I had forgotten that the hat had sub-objects (a hat-band, specifically), and now that the hat was a container, the hat-band was described as being in the hat, rather than part of it. No sweat: this is why I keep the Inform Designers' Manual handy. After flipping through it for a minute, I find the property *add_to_scope*, which allows me (among a host of other uses) to define sub-objects of a container, without actually putting the things in the container. I fix that, and re-compile.
3. Upon testing it yet again, I realize my new dilemma. Due to the way that Inform treats containers and scope rules, my choices are two: either have the contents of the hat visible to the player all the time, or only when the hat is “open.” Since neither an invisible hat nor an “openable” one appeal to me, I come up with a clever little routine that “opens” the hat whenever it is removed, and “closes” it (and thus hides any objects inside) whenever it is put on. Re-compile and smoke break.
4. That now works, but ever the perfectionist, I decide that when the hat is “open” (i.e. unworn) EXAMINING the hat should provide a list of whatever is inside it. That's easy enough to implement, but I realize, upon re-compiling yet again, that the description of the hat describes the hat as “empty” when there is nothing in it—which is right in itself, but gives the puzzle away. Another quick consultation of the Designers' Manual provides me with the answer, and I type the command to compile with a deep sense of pessimism.
5. One which is well-founded: although the description of the hat doesn't give away the fact that things can be put inside it, I realize that *anything* can be put in this hat, including such unwieldy things as a mandolin case. A quick slash-and-burn through

the code later, having marked as “small” everything small enough to fit in the hat, and a re-compile, the hat finally works, more or less.

This whole process took me more than an hour, from conception of the puzzle to the final testing. What do I get for my hour of work? A hat that interacts properly with its environment.

Some people would feel that spending that long on such a seemingly unimportant object is foolish. They are overlooking the importance of suspension of disbelief, the magic ingredient that makes fiction work. Suspension of disbelief is a funny thing: people will believe in dragons, fairies, and the honesty of the government, but give them a hat that they can look through while it's on their head and you've lost them. Making it look easy may be the hardest trick, but it's also the most important. ☒

[Author's Note: After writing this, I discovered one or two more small bugs in the code for the hat, which I squelched rapidly. If you like, think of this as “Seven Easy Steps to Coding a Stetson.”]

Source code example: Defining the Stetson

This is all of the code associated with the Stetson that gave me merry hell, with the backslashes omitted for clarity.—P.D.

```

Attribute small;                                ! Will fit in the hat

Object arthur "Arthur, King of the Hats"
  with name "arthur" "king" "hat" "stetson" "fedora" "gray" "heirloom"
        "classic",
  description [; print "This gray Stetson fedora is a classic, first
                        worn by your grandfather. It's an old family
                        heirloom, and you're proud to have it.";
                if (feather in hatband) print " There's a pigeon
                        feather jauntily stuck in the hatband.";
                if ((children(self) ~= 0) && self has open)
                    <<Search self>>;
                " "];
  before [; Receive: if (noun == feather) <<Insert feather hatband>>;
                if (self hasnt open)
                    "You'd have to take the hat off first.";
                if (noun hasnt small)
                    "That won't fit in your hat.";
                Wear: give self ~open;
                Disrobe: give self open;
  invent [; if (inventory_stage == 2)
                if (self hasnt open || children(self) == 0) rtrue;
                rfalse;
  ],
  add_to_scope hatband,
  capacity 3
has clothing worn proper container ~openable;

```

Note that this chunk of code is *still* not bug-free. Back to the text editor...

IF in 1995

The Year in Review

by C.E. Forman

Thanks to everyone on rec..int-fiction who mailed contributions and suggestions!— C.E.F.*

January

“Waystation” by Stephen Granade, becomes the first major IF release of 1995.

February

“John’s Fire Witch” earns rave reviews on rec.games.int-fiction, demonstrating the strong points of a short work of IF (an interesting precursor to the competition).

April

David Baggett unveils the latest Unnkulian Unventure from Adventions, titled “The Legend Lives!”

After bemoaning a shortage of practical Inform code examples, designers on rec.arts.int-fiction begin discussing the possibility of an Inform programming competition, with the intent of increasing the amount of Inform public-domain code. Not wanting to be left out, TADS users raise a plea to create their own entries. At this point, little is agreed upon, and whether the contest will actually be held remains uncertain.

May

Rules for the competition are debated, but remain vague. Authors are eager to begin their entries but frustrated by the lack of consensus.

June

Activision re-releases the classic Infocom adventures, organized into five separate packages by genre and sold for around \$20 each.

Version 1.0 of the Hugo adventure game compiler is released. Updated versions 1.1, 1.2, and 2.0 are released in the months that follow.

The latest version of the Inform compiler (5.5) appears on the archive. Inform 5.5 is a refinement of version 5.4, with few major enhancements. (Graham Nelson happily reports that the Inform language is now quite stable.)

July

Gerry Kevin Wilson takes on administration of the IF competition, solidifying procedures for entries. The deadline is set for the end of August, and prizes begin to be donated.

August

Brendon Wyber's interactive horror game "Theatre" reaches the archive, becoming the first of several 1995 Inform releases.

"Christminster," by Gareth Rees, is released.

IF players worldwide celebrate the 10th anniversary of Brian Moriarty's "Trinity," coinciding with the 50th anniversary of the dropping of the atomic bomb on Nagasaki.

The IF competition reaches its deadline, having amassed a total of twelve entries.

September

Julian Arnold takes on the task of revising and maintaining the rec.arts.int-fiction FAQ.

Votes for the first annual IF competition are counted, and the winners are announced. First-place prizes go to Andrew Plotkin ("A Change in the Weather") for the Inform category, and Magnus Olsson ("Uncle Zebulon's Will") for the TADS category.

After two years of work, Graham Nelson's "Jigsaw" is finally released. In the weeks to follow, players discuss the implications of a non-gender-specific main NPC as a romantic interest in "Jigsaw."

October

David Baggett releases the TADS source code for "The Legend Lives!" to the IF archive.

A new edition of the Inform Designer's Manual is completed and uploaded to GMD. The latest version is nearly twice the size of the last document, with many clarifications, updates, and a larger number of examples.

November

"The Path to Fortune," volume 1 of the fantasy series "The Windhall Chronicles," is released.

A draft of a document describing standards for Z-machine interpreters (the format used in all Inform and Infocom games) is released, with a final version expected shortly after the beginning of the new year. Standardization of the Z-machine has made further progress in 1995 than any previous year.

December

Carl Muckenhoupt's "Baf's Guide to the IF Archive" is substantially revised and now totals 100 entries. The hundredth game reviewed is, appropriately enough, "Jigsaw."

"The Light: Shelby's Addendum" is released to GMD and Compuserve.

General Trends

Judging from Usenet discussions, the first annual IF competition coupled with the release of a large number of newly released titles marked 1995 as a banner year for the entire IF community. In addition to the short works produced through the competition, close to a dozen major works were released over the course of the year, many in the last six months alone.

The main event in '95 was, obviously, the first Internet IF competition. I haven't looked in the CompuServe GAMERS forum for quite some time, but my personal impression is that the volume of newly written IF has shifted from AGT and CIS/GAMERS to the Internet and rec.*.int-fiction. (The annual competitions on CIS were dropped with the support of AGT, and a new series of competitions—I hope!—was started here.)

Increased availability of quality IF has led to a substantial growth of the GMD archive (with a decrease in the CompuServe GAMERS forum, corresponding with the ending of maintenance and support for AGT), and increased traffic on the rec.*.int-fiction newsgroups. My own rough estimate is that the number of daily

posts has almost tripled from the end of 1994. Discussions themselves have begun to venture specifically into the areas of evaluating and critiquing the latest works.

Graham Nelson put it best in an e-mail conversation: "We are slowly moving towards intelligent criticism (rather than, as used to be the case, discussion consisting only of praising old Infocom games to the skies, as though they were all equally good)...At this moment, for instance, people are debating 'isn't Jigsaw a rather patchy game?' and I'm pleased to see this: people are singling out what they like and dislike, in a way which didn't happen a few years ago with Curses, for example." The beginning of the end of the IF shortage is perhaps largely responsible for this turn in players' behavior, as gamers are no longer afraid of "scaring off" any and all potential authors through such scrutinization.

A shift in the TADS-to-Infom ratio marks another interesting aspect of this year. Inform saw its first surplus with five new titles, all within the last six months, while new TADS games were in short supply compared to previous years, with only three or four major releases (although there's no denying the fact that one of these, David Baggett's "The Legend Lives!", drew anticipation unheard of since the days of Infocom). At the time of this writing, I've heard of an almost equal number of upcoming Inform and TADS games, so a major change in the language of preference is not immediately apparent. The two still seem to be quite equally balanced, and there were equal numbers of entries for each (six apiece) in the competition. Now that more public-domain Inform source code is readily available (again, largely through the competition), we may see an increase in fledgling Inform programmers.

Aiding public interest in Inform were large strides toward sound, in interpreters like ZIP2000, and graphics (version 6 of the Z-Machine), which in turn were brought about primarily through the release of the more clearly written and standardized version 0.2 of the "Specification of the Z-Machine." The Inform language has also seen growth, with Inform 6.0 visible on the horizon and a fully-working Inform linker to be released soon. A Spanish-language translation of the Inform system is also essentially finished and awaiting release.

Mainstream interest in text adventures may also be picking up again. This year we've seen a substantial increase in the number of both IF-related Web sites and online hyperfiction (multiple-path stories in which readers follow different links to influence the plot). Numerous Infocom-related home pages and guides have made it easier for users to locate and use the GMD archive. The summer re-release of the classic Infocom adventures through Activision may rekindle players' memories of text adventures, and a number of computer books and magazines have had disks or CDs containing text adventures packaged along with them. The possibility of packaging the entire GMD archive on CD-ROM, and the upcoming InfoMac CD, have also generated discussion on Usenet.

All in all, 1995 was a tremendous year for IF, but (as FrobozzCo International President John D. Flathead would say), 1996 should be even tremendouser! ☒

Interface Changes: A Brief Look at the Evolution of the Adventure Game Engine

by C.E. Forman

Adventure games have gone through many changes since 1977, but is this progress enough to enable all games to be enjoyed by people with vastly different tastes?

Judging from recent discussions on comp.sys.*.games.adventure and rec.*.int-fiction, the answer is still “not yet.” Many players think fondly of the flexibility of text games and see anything GUI-oriented as a step backward in interface design. Others detest typing, stress quickness of play control, and insist that graphics are the only way to go. Gamers everywhere agree that user-friendliness is the key to a satisfying computer adventure, but not all players are in agreement on how the term is defined.

Let’s take a look at some of the more common (and a few of the outstanding yet very obscure) software packages, and investigate some of the advantages and disadvantages inherent in their play systems.

In the Beginning...

The all-text interface was the earliest play system, and, amazingly enough, is still in wide circulation, due largely to the Infocom format and popular imitations such as the AGT and TADS run-time programs. Screen layouts and presentations vary greatly. A quick browse through the IF archive will reveal:

- The Infocom-esque “prompt-and-status-line” display. The general consensus is that this format is the most enjoyable text interface due to readability, spontaneity, and the ability to see the author’s personal tastes reflected in the layout. A quality parser is necessary to maintain player satisfaction with this type of game.
- Split-screens, with one half devoted to room descriptions and the other to player’s commands and game responses. Frequently objects and exits are listed following brief room text. This format includes the Scott Adams classics as well as many adventures that make use of the simple game engine. Two-word, noun-and-verb parsers, are commonly associated with this type of interface, and the somewhat mechanical listing of rooms, objects, and exits may turn

some players off. Although the Scott Adams games are still played and enjoyed, very few new titles use this particular format.

- Windowed text environments, often combining elements of the split-screen and prompt/status-line layout. Often one window supplies valid directions, a second for the command prompt and resulting text, a third for the player's inventory, etc. Three particular commercial titles in this format merit discussion here.

Mastertronic's "Demon's Tomb," a lesser-known commercial PC text game (reviewed in *XYZZYnews* #2) combined optional windows with a prompt interface, essentially a simplified clone of Infocom's. Separate windows could be invoked for screen configuration and word selection. This allowed players to use the cursor and alphabet keys to select verbs, nouns, and prepositions to form a sentence, eliminating a large portion of typing. It's still something of a surprise to me that this interface didn't catch on, particularly with players who enjoyed text adventures but disliked typing. Perhaps, had it come into play during Infocom's height rather than in 1989, it may very well have.

One of the most unique and imaginative uses of the windowed play system was Paragon Software's "Guardians of Infinity," released in 1988, around the time Infocom was putting out their first graphic IF. The player in "Guardians" spent the game in front of a textual control panel, issuing commands to a set of time-travelling agents in an attempt to save President Kennedy. A single reply window displayed different colored text as each agent responded to the player's instructions, and control could be switched between each agent's window at any time. The interface was driven in real-time, similar to Infocom's "Border Zone," although the multiple character aspect was strongly reminiscent of "Suspended." Although I've never actually completed it, the game is definitely worth checking out for the sheer novelty of implementation.

The most common mass-market game of this type was Infocom's "Beyond Zork," which used windows for a map and prompt, as well as a customizable window for room descriptions, the player's inventory, or character statistics. The windowing aspect was optional, giving players the ability to switch it off and use the full-screen prompt instead. Interestingly, this had the side effect of causing the game to mechanically list all the exits from a particular location, since disabling the windows required disabling the map as well. In this respect, "Beyond Zork" is perhaps the only all-text game that makes extensive use of aspects of all three textual interfaces. (Then again, most versions of BZ did incorporate some simple graphics into the window and map, but BZ remains at heart a text game.)

"Text Adventures With Graphics"

Graphic IF (or my preferred term, "text adventures with graphics"), have been around almost as long as text quests themselves. Inspired by the original Crowther and Woods "Colossal Cave," Ken and Roberta Williams founded Sierra Online and, in 1979, released "Mystery House," the first commercial text-and-graphic adventure.

Although primitive—it used a two-word parser, had several puzzles considered generic by today’s standards, and the graphics were all black-and-white—it’s quite able to hold its own with the Scott Adams quests today (I’d recommend that IF history buffs pick up a copy if you can locate one).

The layout consisted of a picture of the player’s location, occupying the entire screen with the exception of about four lines across the bottom, where the prompt was located. Pressing <ENTER> without typing anything temporarily cleared away the picture, allowing the player to refer back to previous text which scrolled up “behind” the graphic. Thus players could alternate between textual and graphic screens at their leisure, and even had the option of playing the game as a pure text adventure (by not tapping <ENTER> a second time to turn the graphics back on).

Sierra continued this format into the early 1980s with titles such as “Ulysses and the Golden Fleece” and “Wizard and the Princess.” During this time, substantial improvements were made to the graphics, although the interface itself was somewhat lacking. The command prompt extended almost halfway across the screen, severely limiting the player’s typing space. Limitations in text display were also apparent—if the game attempted to display more than three lines of text at one time, part of it would inevitably scroll up behind the picture before the player had had a chance to read it, forcing him/her to tap <ENTER> constantly.

In 1984, Adventure International used a similar engine to re-release several of the Scott Adams quests, with graphic enhancements, calling the games S.A.G.A.S. (Scott Adams’ Graphic Adventure Series). (I’ve played “Adventureland” and “Pirate Adventure” in this format, but am uncertain how many of the other titles were released.)

Where Sierra really struck gold, however, was with Roberta Williams’ “King’s Quest” series, which took the split-screen “picture and prompt” format to new heights, adding animation and pseudo-3D locations, and an on-screen alter-ego to represent the player.

The series was enhanced by a parser that, although nowhere near the power of Infocom’s, permitted direct and indirect objects, articles, and reasonable synonyms. Games of this sort fell short of being pure adventure, however, as the need to move the character about the screen using the cursor keys provided ample opportunity for unfair arcade-style puzzles. Among the more irritating were the need to be immediately beside an object to interact with it, avoiding dangerous characters, and, most infuriating, maneuvering the character across tiny, winding paths where one slip meant instant death.

“King’s Quest 4,” the last of this particular class of Sierra games, was released in 1988, at about the time Infocom, through Mediagenic, first went graphical. During the next two years, Infocom surpassed its typical number of releases, issuing a steady stream of RPGs (“BattleTech,” “QuarterStaff,” “Journey”) and graphical IF (“Arthur,” “Shogun,” “Zork Zero”). For the most part, the new IF format kept the classic Infocom features, while adding graphical puzzles, maps, screen borders, a compass rose, and pictures to complement the textual imagery.

When Infocom folded shortly thereafter, authors Steve Meretzky and Bob Bates teamed up to co-found Legend Entertainment, with the intention of continuing the commercial release of IF. By this time, though, graphics were (and still are) the hot topic, and many players, new and old, frowned on typing, despite the fact that countless Infocom fanatics still loved the old method of computer storytelling.

Bates and a couple other Legend employees completely redid the text adventure system architecture, splitting the screen into four parts—a text window, a graphics window (which could also be made to display room descriptions or player inventories), a set of parallel vertical menus (one for nouns, one for verbs and prepositions), and a box containing a clickable compass rose and buttons for customizing the screen layout. If the player desired, it was possible to remove the word menus, leaving half the screen devoted to text, or even to remove everything and bring back the all-text interface.

This game engine gained attention and praise from the adventuring community immediately, because it catered to both typists and non-typists, to both graphics players and text players. One computer magazine even proclaimed that Legend “finally got the adventuring system right.” For a short time longer, text adventures had mass-market appeal.

One disadvantage here was that, as Legend expanded their game engine, the increasing memory required to handle each package meant that players with very old systems could not run them. Legend’s game format was nowhere near the efficiency of Infocom’s method of Z-Code and interpreters. Another problem came about as more and more verbs were added to the menu bars. Non-typists would have to continually scroll down to find the location of the verb they wanted to use. (Although, for the most part, Legend avoided this by putting the commonly used verbs at the top, within easy access. However, in later releases, they then began bringing up obscure verbs when they became relevant to the game, which tended to give away puzzles.)

Legend is not the only company that attempted to make the parser adventure more commercially accessible by adding bells and whistles. Virgin Games, after using a split screen for text and graphics in games such as “The Pawn” and “Jinxter,” brought several of its titles into the “Magnetic Scrolls” format in early 1990. This GUI-based environment behaved similarly to Windows or the Macintosh operating system, with different windows for text, objects in the room, inventory, graphics, the compass, hints, and maps.

Unlike Legend, the Magnetic Scrolls windows were mobile and resizable, allowing greater flexibility in the customization of the interface. A menu bar along the top replaced the status line and provided small, organized lists of common verbs, and a cut-and-paste option allowed players to highlight words in the text itself and copy them to the prompt.

Like Legend’s engine, though, Magnetic Scrolls required more modern hardware to permit the usage of all its features. The animated graphics in particular were taxing to processor speed and memory, and, when using such features, a 386 was barely enough to squeak by.

Earlier attempts (mid- to late-1980's) at graphical IF interfaces included Interplay's "Tass Times in ToneTown" (co-authored by Michael Berlyn), and ICOM's windowed series, both of which appear to have influenced Magnetic Scrolls. ICOM's adventures ("Deja Vu," "Uninvited," "Shadowgate") were initially released solely for the Macintosh platform, but several years back found their way to PCs, and even the Nintendo Entertainment System. (Too bad the latter didn't support keyboards—IF could have made a comeback!)

One other product deserves a special mention here—Stuart Smith's "Adventure Construction Set," published in 1984 by Electronic Arts. ACS was the first serious attempt at distributing a piece of software that permitted users to create and play their own text/graphic adventures.

The computer-aided design system used a screen layout similar to Sierra's, but the interface was all its own. The graphic portion of the screen was divided into "blocks," each of which contained a stack of one or more "things" (there were 12 different classes of these) or creatures. Using the cursor keys (or mouse or joystick), up to four players took turns exploring the game world.

The combat system was quite RPG-oriented, but ACS was suited nicely to the construction of simple puzzles, provided that players were willing to take the time to learn how to make creative use of the design system. ACS had its limitations, but it was truly a remarkable piece of software.

The Graphic Revolution

The turn toward graphics began, not surprisingly, as soon as existing hardware became powerful enough to support their extensive development. In commercial software today, pseudo-parsers typically replace the traditional method of sentence formation. Players select actions from a limited menu of verbs, and nouns from a graphic location picture and/or inventory window.

LucasArts' "Maniac Mansion" and "Secret of Monkey Island" games, as well as Adventuresoft's "Simon the Sorcerer," among others, use this interface. Legend Entertainment's more recent releases ("Companions of Xanth," "Death Gate," and the soon-to-be-released "Mission Critical") use a similar game engine. Activision's attempt, used in "Return to Zork" (and, I expect, in their upcoming "Planetfall" game) has players connect an icon to other icon in a set, where each icon represents a particular item or action.

(Although I can accept most of the graphical games I've played as they are, I must admit, I really missed typing here.)

Others, such as "Myst," "The 7th Guest," and Roberta Williams' latest offering, "Phantasmagoria," veer away from traditional puzzle-solving adventure and toward the newer medium of the "interactive movie." But that's another article. ☒

Lights... Camera... Demo!

With this issue, I'd like to announce two new developments in my Sneak Preview procedures. Where space permits, I'd like to print opening game descriptions, like the one shown below for "Gumshoe." In a similar vein, I now have software for creating animations of on-screen activity, which means that I can create short movies of the opening moves of a text adventure game. I'd like to create such movies for Sneak Preview games and make the movie files available on my Web site. This will allow the game developers to create a short demo of their games without any extra coding, and will be a service for my Web readers who want to download these demo files. Without any other further ado, here are this month's sneak previews—two IF games-in-the-works that should be available soon:

1. **Gumshoe** by Mike Oliphant (oliphant@cogsci.ucsd.edu) is an upcoming Inform game that he says is "about midway through testing and should be released this spring."

Reluctantly, you crack a gluey eyelid to the world. Your head surges with pain, and you quickly shut it again.

The whiskey gods were not kind to you last night.

You search your memory for the reason behind this latest drinking binge. This proves difficult, because there have been many such nights since you were removed from your job as detective on the force. Now it comes back to you. The rent being past due on your new office. Marge badgering you about her paycheck. Your empty bank account. Your complete lack of clients. The money you owe Jimmy Voigt. You wince, not even wanting to think about Jimmy Voigt.

Last night, you took your troubles to 'ole Johnny Walker and he listened good. One drink led to another, and that naturally led to the next. Now you find yourself awakening on a cold, tiled floor. You bravely reopen your eyes and see that you are curled up at the foot of some private investigator's office door.

Hey! It's your office!!

The one you owe rent on... You wince again. Ignoring the warning bells going off in your head, you slowly get to your feet and manage to stand shakily. You're beginning to understand what it means to be a...

Gumshoe
An Interactive Investigation
Copyright (c) 1995 by Mike Oliphant
Release 1 / Serial number 960106 / Inform v1502 Library 5/12

2. **Reality Shift, The Green Crystal, Part 1** is an upcoming AGT game by L. Ross Raszewski (rraszews@skipjack.bluecrab.org) in which you're on a mission to save the world. The game includes pop-up hints, over 100 locations, four player characters and some 15 non-player characters. Beta testers are welcomed. ☒

Porting From OS/2? Rexx-Adventure Steps Up to the Plate

by Mike DeSanto (desantom@io.com)

The idea of writing a text adventure engine (the term interactive fiction is a new one to me) has been wandering around in my head for about 4 years. I played with AGT while I was in college, but got bogged down with classes and never really got anywhere. I also spent some time fiddling with LambdaMOO in the Metaverse at Illuminati Online (io.com). That never really went anywhere either.

I always get frustrated, you see, with “guess the verb” problems. With graphic adventures it got even worse with “find the pixel” garbage. The other problem with most adventures, text or graphics, is telling background objects from objects you can manipulate. The usual routes around this are 1) including only objects you can manipulate in the descriptions or 2) spending a lot of time reading “I don’t know what the XXXX is.”

I wanted a game where it was always clear exactly what you could manipulate and how you could manipulate it. Take, for example, the aquarium in Zork II. You have to break the aquarium by throwing the sword at it. Swords, in my experience, are not made to be thrown. You throw spears, rocks, or knives. You do not throw swords. If throwing is an option for the sword, I bloody well want to know that before I go looking for a hammer!

So, Rexx-Adventure was born. Since 90 percent of creativity is intellectual property theft, I will credit my sources. Much of the data storage design was inspired by LambdaMOO. I loved the simple IF-THEN-ELSE CONTINUE structure of AGT. The screen layout was inspired by the early Legend games (Time Quest in particular), and the Magnetic Scrolls series.

So, I started working on Rexx-Adventure. In the design, I tried to meet the following goals:

1. Simple to use. Playing a game should not require instructions. Simply start the engine, load the source, and play.

2. Simple to program. A simple game should not require extensive programming knowledge.
3. Everything based on objects, not on fixed flags and verb lists. Every object has its own verb list and arbitrary flags.
4. No guessing about what you can and can not manipulate, or how you can manipulate it.

These goals have been met. Rexx-Adventure will automatically load and run a game simply by double-clicking the game source file. The source files are easy to read and reasonably logically laid out. All checks done every turn are stored in a centralized place (a special object just for that task).

Every object can have any number of flags to determine light, weight, open/closed, or any other property you wish. Fewer than five flags are required, and several of those can be ignored if you want to. Programming is simple, with 20 or so functions providing everything you need for the average game. If you need more, outside functions can be added. In fact, any program that will run under OS/2 can be a part of Rexx-Adventure.

However, there are a couple of serious limitations, hopefully to be fixed in the next release. First, there is no implementation of indirect objects. When I started, I said "Adventure didn't need indirect objects, why do I?" Now I know better. The other major limitation is the absence of "object in object" support. For example, "water (in the bottle)" or "fluff (In the thing-your-grandmother-gave-you-that-you-don't-know-what-it-is).

Despite these limitations, I am proud of Rexx-Adventure. It is a very powerful engine with (hopefully) a promising future. I hope all of you out there with OS/2 will try it. Rexx-Adventure is released as freeware. Feel free to write shareware games using the engine, as long as it is clear that the fee is for your game, not my engine.

Rexx-Adventure requires OS/2 Presentation Manager. It was written using VX-REXX from Watcom Inc.

If anyone is interested in porting Rexx-Adventure to other REXX platforms, I will help in any way I can, including giving you the source code. REXX interpreters are available for all IBM mainframes, Macintosh and Amiga. REXX is also included with IBMs PC DOS 7, and MS-DOS interpreters are available.

Matthew Raymond has inquired to me about writing an AGT-to-Rexx-Adventure conversion program. He can be contacted at mraymond@acs.bu.edu.

I would be glad to hear any questions or comments about both Rexx-Adventure and The Resident. I can be reached at desantom@io.com. ☼

Lethe Flow Phoenix: A Flight of Fantasy

release 2

Parser: TADS

Author: Daniel Shiovitz (scythe@u.washington.edu)

Availability: ftp.gmd.de/if-archive/games/tads/
lethe.zip or lth-noio.zip

Requires: TADS run-time interpreter

Set in a refreshingly far-off, sometimes mystical environment, *Lethe Flow Phoenix* has a number of satisfying, not-too-difficult puzzles and some interesting NPCs. The game's highly metaphorical plot and screensful of explanatory background material, however, made it difficult for me to get really drawn into the setting.

As the game begins, your character (a woman, incidentally, although this seems pretty arbitrary) has headed off on a solo camping trip one Halloween for some soul-searching and contemplation. Awakened by strange noises, you venture into the night and accidentally pitch yourself over a cliff. When you awaken, you find yourself in a grassy field, left to discover both how to maneuver about this completely unfamiliar environment and your purpose here. The game reminded me a little of "The Sound of One Hand Clapping" in terms of some similarities in tone and style, and I wasn't surprised to see that game listed in the credits as the author's favorite work of IF.

I'll admit I was at first put off by the timed how-to-avoid-dying-of-starvation puzzle that begins the game, the kind that crops up in so many text adventures. There are a couple of fatal endings to the game—for example, walking off another cliff or carrying around a hand grenade after removing its pin—but none that seem unfair. Overly cautious players may miss some important information, however; for example, for much of the time I played I avoided exploring desert areas for fear of dying of thirst, although I still think that that's a reasonable assumption to make.

Lethe Flow Phoenix really contains only about a dozen or so puzzles in terms of getting around obstacles or obtaining the objects you need to overcome an obstacles. Experienced players should be able to progress most of the way through the game, if not complete it, in the course of a few hours of play.

I really liked the scoring system used in *Lethe Flow Phoenix*. Instead of informing the

player that a certain number of points are awarded for solving a puzzle, the game states how many "major goals" and "minor goals" the player has solved, identifies what each of those goals are, and tallies what percentage of the total this constitutes.

Examining every object and studying room descriptions were extremely helpful for figuring out what to do next. The object descriptions were probably the best guide for how to implement each item. I really didn't experience any "guess-the-verb" problems that sometimes come up in trying to solve IF puzzles.

There is a help command, but the only hints included in the game are fairly general and there are no explicit spoilers. There are a couple of NPCs who will answer questions posed to them, but each has information about only a few limited objects (in my experience, at least). I found it a little frustrating to try to elicit information from these characters, especially because you have only a few turns to ask one of the NPCs any questions. Some more explicit hints would be good to include for beginners.

The game is small in terms of number of rooms, but feels much more extensive with the sudden, concentrated onrush of background material and character development in the main part of the game. While the game's title is explained in a READ.ME 1ST document, the background information necessary to understanding the game's goal is only introduced once you've progressed past the game's beginning section.

I did run across a few technical difficulties in running the game. Two versions are available on the GMD site—one using the file i/o features available in TADS 2.2 to read and write to a file for saving options from game to game. Unfortunately, neither TADS file would run on my Mac, although I was able to get the game to run on a PC. Like so many text adventures, this release had several bugs I ran across that would not be difficult to re-code. For example, at one point I generated a TADS error. At another point in the game, giving the command LOOK UNDER BUSHES resulted in my picking up everything that was under said bushes.

While I was a little disappointed with the game's story development, I did enjoy the puzzles and the NPCs, especially the garden beasts (who seemed like something out of Disney's "Fantasia"). Although I found the story a little rough, I'm looking forward to seeing future games from this author. ☒

—Stuart Beach

The companion disk for XYZZYnews #7 contains the following game files. It's a good deal for people who have slower modems—at 2400 bps, it'd take a heck of a long time to download the contents of the companion disk. It's also a good deal for people with limited or no access to FTP sites or online services as a source for new games. If you're reading an electronic version of this issue, you can obtain this games disk with a print copy of XYZZYnews #7 by enclosing \$3.50 for postage and handling with the coupon on the bottom of this page. If you play and enjoy these games, please pay the shareware fees as applicable.

THE LIGHT: SHELBY'S ADDENDUM—In this TADS game by C.A. McCarthy, as Shelby you're just the low man on the totem pole at the light-house, but you know something's definitely wrong as you make your return there. An intriguing mystery and race against time. Available in separate standalone versions for Mac and PC users.

This issue also includes three games originally written by Jim MacBrayne (jmacb@medusa.u-net.com) in 1989-90 and recently ported to TADS.:

FRUSTRATION—Chasing after a wayward shopping list, you accidentally plummet down an open manhole. What adventures await you at the other end?

THE MISSION—A quest for a shoelace? Well, there's a lot at stake here...and it's up to you whether or not you win the bet and succeed at your mission.

HOLY GRAIL—The consequences to be faced if you fail in your mission are dire indeed. Are you up to the challenge?

XYZZYnews Magazine/Disk Order Form

Checks and money orders preferred. Please send coupon with payment to:

Eileen Mullin
XYZZYnews
160 W. 24th Street, Ste. 7C,
New York, NY 10011.

- Please send me a copy of the print version and companion games disk for XYZZYnews Issue #7. I have enclosed \$3.50 for postage & handling. (Check one: Mac disk ___ or PC disk ___)
- I need just the companion games disk for XYZZYnews. I have enclosed \$2.50 for postage & handling. (Check one: Mac disk ___ or PC disk ___)
- Sign me up for a 6-issue subscription (within the U.S.). I have enclosed US\$21 for postage and handling. (Check one: Mac disk ___ or PC disk ___)
- Please send me a copy of the upcoming XYZZYnews Issue #8. I have enclosed \$3.50 for postage & handling. (Check one: Mac disk ___ or PC disk ___)

Full Name (please print): _____

Street Address: _____

City: _____ State: _____ Zip/Postal Code: _____

Country: _____ Email Address: _____