The Icon Newsletter

Contents

Version 9 of Icon

Version 9 of Icon now is available for MS-DOS, UNIX, and VMS. The UNIX and VMS versions support Icon's graphics facilities under the X Window System.

Implementations for other platforms still are in progress.

New Features

The major changes in Version 9, and the ones that are responsible for our assigning a new major version number, are in its graphics facilities.

There are many improvements in this area, including a portable font naming system, facilities for drawing color images, and support for the GIF image file format. We've also changed the names of the graphics functions, dropping the initial X in anticipation of Version 9 implementations with graphics that do not require X. There also are new and changed functions.

This is the first time in the development of Icon that we've made major changes that are not upward compatible. We decided that the advantages of the new graphics facilities outweigh the conversion problems. We have, however, provided conversion aids.

There are other new features in Version 9; not earth-shaking ones, but small useful things like being able to add many elements to a list at one time. There's also support for dynamically loading C functions on some UNIX platforms.

Implementation Improvements

We've made several significant improvements to the implementation of Version 9 that will be helpful to users.

The Icon linker now eliminates declarations (notably procedures) that are not referenced in a program. This may substantially reduce the size of icode files, especially when packages of library procedures are linked but not all the procedures in them are used.

The MS-DOS linker now creates .exe files, so that it is no longer necessary to run iconx. (It still is necessary to have icon.exe on your path, but that's transparent when you're running Icon programs.)

The UNIX implementation now uses shell headers to make icode files executable. This substantially reduces the sizes of icode files on many platforms.

Icon Program Library

There's a new version of the Icon program library to go along with Version 9. It has many additions and improvements, and the graphics portion has been updated to Version 9. The library also includes an improved interface builder for platforms that support Icon's graphic facilities.

Should You Get Version 9?

We recommend that you get Version 9 when it's available for your platform. The improvements in the implementation alone make the upgrade worthwhile.

We also suggest that you get the Version 9 Icon program library. It has a wealth of material that's useful in its own right, as well as examples of Icon programming techniques that are worth studying by novice and expert alike. If you're planning to build graphics applications of any complexity, the library is a virtual necessity, since it now contains many support procedures to supplement the built-in repertoire.

As usual, Icon program material is available by FTP or on physical media. See the order form at the end of this *Newsletter*.

Update Subscriptions

We offer update subscriptions to the source code for Icon and to the Icon program library.

Source updates: If you're interested in the latest developments in Icon, don't want to wait for the next official release, and are willing to compile your own version of Icon, consider subscribing to the updates to the source code. Updates are issued about three times a year. Each update contains complete source code; you can start a subscription at any time. *Note:* Present subscribers to the source-code updates already have Version 9.

Library updates: Icon program library updates are incremental and are based on the most recent official release of the library. Library updates are issued about three times a year. Current subscribers already have most of the material in the new Version 9 Icon program library. They will get the remainder in their next update.

New subscribers to the library updates need Version 9 of the Icon program library. New sub-scribers who do not order the Version 9 program library and do not tell us they already have it will be sent the Version 9 library, counted as two updates. (You can get the Version 9 program library by FTP, but updates are available only by subscription.)

The I con Analyst

We're now into the fifth year of The I con Analyst, a newsletter that is devoted to in-depth coverage of the Icon programming language.

The Analyst has articles on language features, programming techniques, and aspects of the implementation. The Analyst is published six times a year. An issue usually contains two or three lengthy articles, as well as programming tips and other smaller items.

Among the articles planned for future issue of the Analyst are ones on random numbers, string invocation, static and dynamic analysis of Icon programs, and symbol manipulation.

We encourage serious users of Icon to subscribe to the Analyst. If you'd like to see what an Analyst is like before subscribing, ask for a free sample copy.

Back issues of the first four years are available as a package, but new subscriptions can be started with any issue. Unless you tell us otherwise, your new subscription will start with the most recently published issue.

See the order form at the end of this *Newsletter*.

BYTE Article on Icon

In case you haven't seen it, where was a short article on Icon in the May 1994 issue of BYTE magazine in the **Some Assembly Required** feature.

The space available to us was quite limited, so we were able to present only the high points of Icon. Nonetheless, you may find the article interesting or useful for friends who want a brief summary of Icon.

We hope that BYTE will run an article on Icon's new graphics features in an upcoming issue.

Icon RBBS

With the increased ease of access to Internet and FTP electronic e-mail, our electronic bulletin board has had less use.

We're now to the point where we can no longer justify the expense and time spent maintaining our RBBS. We haven't yet decided when we will discontinue it, but if you use our RBBS to get Icon material, you should plan on alternative methods in the future.

In the next section we're re-running an earlier article on how to get FTP material via electronic mail.

FTP Files by Electronic Mail

If you have access to electronic mail but not to FTP, you now can get files from our FTP area through electronic mail. Using a facility called ftpmail, you can send a script of commands to our FTP site. The material specified in these commands is sent back to you. You can get both directory listings and files. Large files are automatically split into smaller pieces to facilitate transfer.

To use ftpmail, send a message to

ftpmail@cs.arizona.edu

Your message must begin with the command open and end with the command quit. (The subject field of the message is ignored.) There are a dozen or so commands altogether. We'll list only the most basic ones here. You can get more information by sending the following message:

open
help
quit

You'll get a UNIX-style manual page with all the particulars by return e-mail.

If you want the reply to go to a different address than the one you're sending from, include the reply–to command in your message, as in

reply–to georgem@ece.foodom.edu

In fact, if you try ftpmail and don't get a reply, it's probably because the return address as it appears in your message header doesn't work.

(We find that to be the case fairly frequently in mail sent to icon–project.) If this happens, use reply–to, being careful to give a proper address.

In order to get directory listings and files, you'll need to specify where they are. The cd (change directory) command does this. To get to Icon material, start with

cd /icon

A directory listing of this area then can be obtained with

dir

The command get is used to get a file. In the Icon FTP area, there are READ.ME files in almost all directories. Thus,

cd /icon
get READ.ME

gets you our top-level READ.ME file.

Binary files, such as archives and executables, are automatically encoded as ASCII text to allow transfer via e-mail. The default encoding is uuencode. btoa encoding also is available. The commands btoa and uuencode can be used to specify the encoding you want. You'll need either uudecode or atob to convert encoded files back into their binary form. The Icon program library contains a program iidecode that has the same functionality as uudecode. It's slow, but it should work on any platform.

Give ftpmail a try. If you have problems, send e-mail to icon–project@cs.arizona.edu for assistance.

Graphics Credits

The image on the back cover of this *Newsletter* was created using Specular TextureScape, an application that supports the construction of repeating patterns.

Unlike most applications for creating repeating patterns, TextureScape works with Postscript elements, has three-dimensional effects, and allows multiple layers. The image on the back cover has two layers. The background one consists of simple circles shaped with TextureScape facilities. The foreground layer is composed of copies of the Icon logo.

Stereograms

The Icon auto-stereogram that we published in Issue 39 of the *Newsletter* has provoked more comments and questions than anything else that's ever appeared in this *Newsletter.* Some persons were able to see the image easily, while others never could. We've even been accused (jokingly) of ruining one reader's eyesight.

Most to the queries we've received have been about computer-generated stereograms in general. We haven't kept up with the subject and don't know enough about it to answer questions ourselves. There is, however, a lot of material on stereograms available on Internet.

Tim Korb, one of the Icon pioneers, recommends the FAQ (frequently asked questions with answers) that is available via Mosaic. It's at URL

http://mphh2.ph.man.ac.uk/gareth/nzfaq.html

Tim also sent along a simple ASCII stereogram done by "DR J" that has an Arizona theme. It appears at the bottom of this page.

To view this stereogram, hold the page close to your nose, gradually move it away from your face while keeping your eyes unfocused. The stereographic images should "jump out" at some point, after which you can view it easily until you change your focus of attention.

We welcome Icon-related stereograms for future inclusion in this *Newsletter.*

Language Archives

As we mentioned in *Newsletter 44*, we've made arrangements to transfer the material that we have on SNOBOL, Icon, and related programming languages to the Charles Babbage Institute.
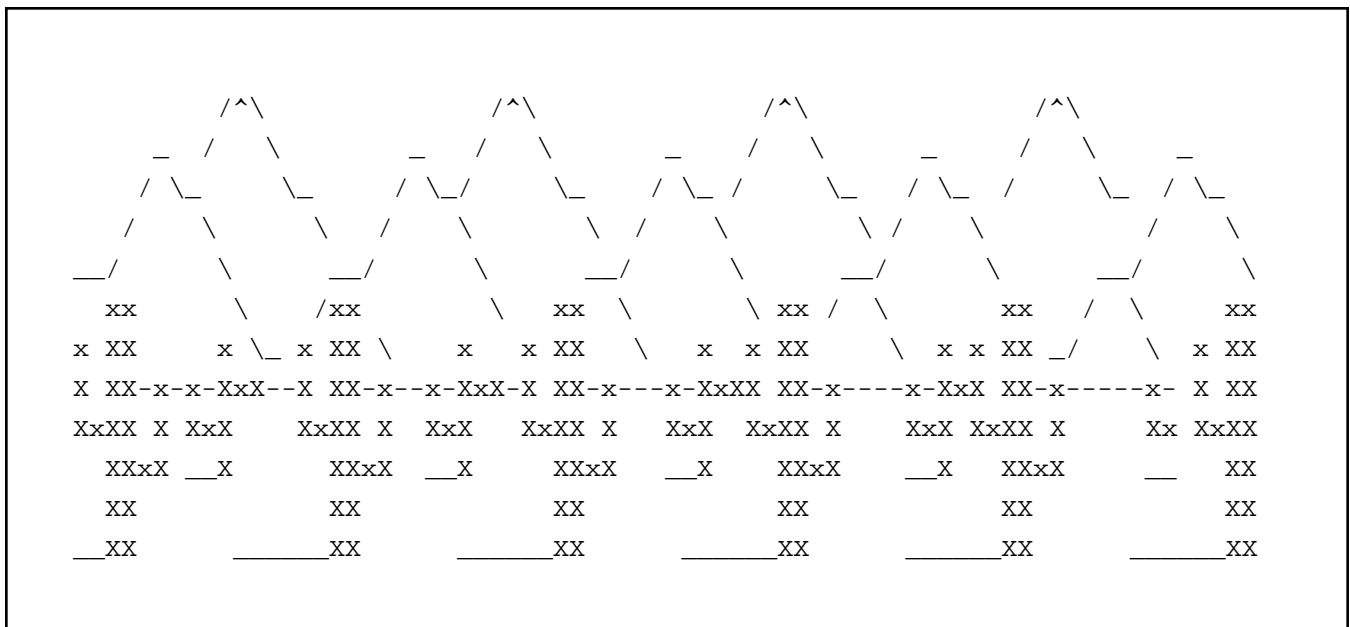
We've already transferred some material, including paper correspondence from the early 1960s through the late 1980s, as well as books and large manuals. We're presently working on organizing older material to send.

This material is available as described in the following article by Bruce Bruemmer.

Access to the Archival Collection at the Charles Babbage Institute

CBI's archival collection is open to all researchers, who are encouraged to contact the archival staff prior to visiting CBI to ensure that relevant materials are available and open to research. While every effort is made to obtain archival materials without restrictions on use, some collections must be restricted because of the nature of the information or the poor physical condition of the records.

CBI's collection is cataloged on the Research Libraries Information Network (RLIN) and the University of Minnesota's LUMINA catalog. LUMINA is accessible from terminals on campus, through computer modems, or through the Internet. Internet users can telnet the library cata-

```
         /^\              /^\             /^\              /^\
     _  /   \         _  /   \        _  /   \         _  /   \       _
    / \_     \_     / \_/     \_    / \_ /     \_    / \_ /     \_    / \_
   /     \     \   /     \     /     \     \   /   / \     /         /     \
 __/       \    __/       \    __/       \    __/       \        __/       \
    xx       \   /xx        \   xx  \     \ xx /  \        xx   /  \      xx
  x  XX     x \_ x XX \    x   x XX    \  x  x XX    \  x x XX _/    \  x XX
  X  XX-x-x-XxX--X XX-x--x-XxX-X XX-x---x-XxXX XX-x----x-XxX XX-x-----x- X XX
  XxXX X XxX     XxXX X  XxX    XxXX X   XxX  XxXX X    XxX XxXX X    Xx XxXX
   XXxX __X        XXxX __X       XXxX   __X     XXxX   __X   XXxX     __     XX
   XX            XX            XX            XX            XX            XX
  __XX         _____XX       _____XX       _____XX       _____XX       _____XX
```

log via pubinfo.ais.umn.edu or the University of Minnesota Gopher (gopher.tc.umn.edu). More detailed information on archival records is available through finding aids ranging from lists of folders to databases that index items in a collection. No finding aid is currently available for the Griswold Papers, but a more detailed list will be developed as records are sent to CBI.

CBI's reading room is open during business hours from Monday through Friday, and on Saturday by special arrangement. A copying service for paper, machine-readable, and photographic formats is available. Same-day service is usually possible for photocopies. Reproduction of photographic material requires five working days.

Researchers unable to visit CBI directly may request photocopies delivered by mail or fax, subject to a fee. CBI is experimenting with other techniques of document delivery, such as the Internet transmission and interlibrary loan of oral history transcripts. The University of Minnesota Libraries Gopher soon will have a section devoted to CBI, and CBI's participation in a World Wide Web server is under development.

CBI serves as an information clearinghouse for all archival collections relating to the history of computing. In 1987 the archives published *Resources for the History of Computing*, the first comprehensive research guide to archival material held by repositories in the U.S. and Canada. CBI continues to maintain information about other repositories' holdings, and researchers have access to a file of finding aids on non-CBI collections.

CBI's archivist, Bruce Bruemmer, can be reached at:

Charles Babbage Institute
103 Walter Library
University of Minnesota
Minneapolis, MN 55455

voice: 612-624-5050
fax: 612-625-8054
e-mail: bruce@fs1.itdean.umn.edu

Downloading Icon Material

Most implementations of Icon are available for downloading via anonymous FTP:

FTP: cs.arizona.edu (cd /icon)

IClip

Editors' note: Art Eschenlauer has written an application that is based on the standalone Version 8.0 Macintosh Icon. Here's his description.

IClip is a Macintosh application that "filters" text data on the clipboard. The user copies text data onto the clipboard from within any application, executes an IClip filter, and pastes the resulting text where it is needed.

IClip filters can be programmed using the Macintosh stand-alone implementation of the Icon programming language, Version 8.0. IClip filters are Icon programs that are slightly modified after translation. Text data on the clipboard serves as the standard input to the IClip filter, and, when execution is completed, the standard output is copied to the clipboard as text data, in place of the input data. (All other IClip operations, including file I/O, work exactly as they do for iconx.) Thus, filters can be designed and tested using iconx and subsequently converted to IClip-filter format. (Since the Icon interpreter is part of the IClip application, iconx is *not* required in addition to IClip in order to execute IClip filters.)

IClip does not support styled text. Bold, italic, underlining, superscripting and subscripting information will be lost, although styling information should not interfere with the operation of IClip. IClip is intended for System 7.

IClip is not intended to be a "full-blown" Macintosh implementation of Icon. Instead, its purpose is to make Icon's text-processing and computation power available for clipboard-filtering tasks.

An Example

Consider a simple Icon program, reverse.icn, that reverses the lines of an input file:

```
procedure main()
    every writes(reverse(read()) )
end
```

This is translated using the command

```
 icont reverse
```

producing the iconx document reverse. If you run reverse and type

```
 Able Was I Ere I Saw Elba
 ^D
```

the output is

ablE waS I erE I saW elbA

To convert reverse from an iconx document to an IClip filter-document, drag-and-drop it onto the icon of the Icon to IClip program. Now, if you are using a word processor (or whatever) and you copy

Able Was I Ere I Saw Elba

onto the clipboard and run reverse

ablE waS I erE I saW elbA

is placed onto the clipboard in its place. For convenience (under System 7), the reverse IClip filter-document can be put into the Apple Menu Items folder and invoked simply by selecting reverse from the Apple menu.

Editors' note: IClip can be obtained by FTP from cs.arizona.edu. cd /icon/iclip *to see what's there.*

Phasing Out 5.25" Floppy Disks

Any organization that distributes software on physical media for many platforms has a problem with the wide variety of media and recording formats that are in common use.

Over a period of years, the Icon Project has provided different forms of magnetic tape, data cartridges, and diskettes. In some cases, different formats are necessary. For example, without special facilities, a Macintosh cannot read a PC DOS diskette, and vice versa. To make matters worse, the two systems have very different ideas of what a file is.

The variety of media and formats presents many problems. For example, the Icon program library not only has to be written differently for the Macintosh and MS-DOS, but different file formats, support scripts, and documentation are needed. Then copies for distribution need to be made using different equipment and procedures, different labels are needed, separate stocks must be maintained, and so on. These are not challenging problems (except that getting everything right is more difficult than it may appear), but they take time and effort.

Such requirements for different platforms are unavoidable. But there are problems even for a single platform. There was a time when low-density, 360 KB 5.25" floppy disks were the standard media for data exchange on PCs. (We try to forget when there were single-sided versions of this type of floppy.)
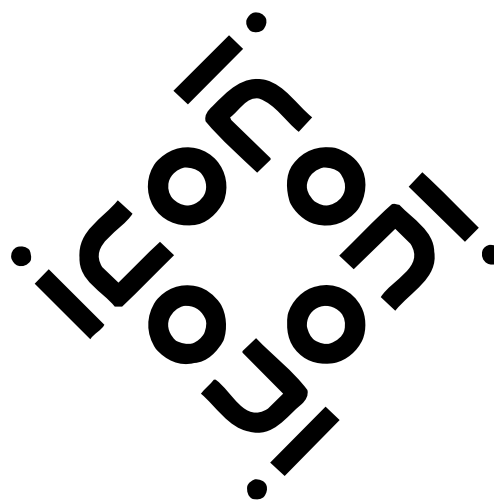
Then came high-density, 1.2 MB 5.25" floppies. These never caught on in a big way, although many recent PCs have drives that can read them. The Icon Project never distributed high-density 5.25" floppies, although we had one potential customer who said he'd wait until we did. For all we know, he may still be waiting.

Low-density, double-sided 720 KB 3.5" diskettes (hardly "floppy") followed. They were attractive because of their smaller size, increased capacity over low-density 5.25" floppies, and physical robustness. 3.5" diskettes made packaging for PCs much easier; not only were fewer diskettes needed, but larger files could be accommodated on a single disk. But we still had to provide 5.25" diskettes, so we had two kinds of diskettes to worry about, with the kinds of problems mentioned above.

Next came high-density, 1.44 MB 3.5" diskettes, with the advantage of even more space. But there were PCs that could read low-density 3.5" diskettes that could not read the high-density ones. Fortunately, as far as we know, all 3.5" drives that can read high-density 3.5" diskettes also can read low-density ones.

There are a variety of other magnetic media, including higher-density 3.5" diskettes, smaller, higher-density diskettes, dismountable disk packs, Bernoulli disks, removable hard-disk cartridges, writable optical disks, and, of course CD-ROM. Of these, only CD-ROM is suitable for general software distribution. This is not to say that we've not been asked to provide Icon on some of the other media.

Our immediate problem is more mundane. We can no longer afford to support the variety of PC diskettes that we've provided in the past. If you look at recent issues of the *Newsletter*, you'll see we've been shifting from 5.25" diskettes to 3.5" ones and, for those, toward high density. We're now to the point of phasing out 5.25" floppies, as you'll see on the order form at the end of this *Newsletter*.

Those of you who have PCs of recent manufacture may wonder what all the fuss is about and why we didn't stop distributing 5.25" floppies long ago. From a purely economic point, we should have (as most commercial companies have done). But we know for a fact that we have users with only 5.25" drives.

We even had one user with a PC that had only one floppy drive and no hard disk. He couldn't use the floppy we sent him because the files were compressed and he had no place to which to decompress them! Well, we can't do much about that kind of problem — without compression, we'd have to use 2 or 3 times as many diskettes as we do, and some uncompressed files are too large to fit on a single diskette anyway.

With apologies for the inconvenience of those few users who have PCs with only 5.25" drives, we've eliminated 5.25" floppies from all our distributions except the executables for Version 9 of Icon for MS-DOS.

We're also be shifting toward high-density 3.5" diskettes as the default, but we'll continue to provide low-density 3.5" diskettes for some distributions.

And we're beginning to see hints of an Icon CD-ROM on the horizon.

From Our Mail

How's the NT implementation of Icon coming?

Clint Jeffery has a beta-test version without out graphics working. Executable binaries are available via FTP to cs.arizona.edu; cd /icon/ binaries/nt.

In the last issue of the Newsletter you said implementations of Icon with graphics capabilities were in progress for Windows and the Macintosh. When will these be done?

Clint Jeffery, who is working on these implementations with his students, reports considerable progress, but it's still too soon to say when they'll be done.

When I try to execute

```
show := open("show", "x")
```

I get a run-time error that says "invalid argument to open". What's going on?

You're trying to open a window in a version of Icon that doesn't support graphics. The argument "x", derived from the original X Window System implementation of graphics, requests a window. We admit the error message is confusing, and in Version 9 an expression like this simply fails if executed in a version of Icon that doesn't support graphics. This is more reasonable, since failure in Icon generally is used for operations that make sense in principle but that can't be carried out successfully. Incidentally, in Version 9, "g" (graphics) is used to open a window, but there also are higher-level graphics procedures for opening windows.

What became of The ProIcon Group, which was listed on ProIcon literature before ProIcon was discontinued as a commercial product?

The ProIcon Group was a joint venture between Catpsaw, Inc. and The Bright Forest Company. It was formed to develop, document, and market ProIcon. The joint venture was dissolved when marketing of ProIcon ceased.

Isn't is a bit unusual to put a commercial application like ProIcon in the public domain?

It's unusual, but not unprecedented. Usually when a commercial application is discontinued, it's either replaced by a new version (for which a freely available former version would be competition) or the company goes out of business. It's also a considerable effort to put a former product in the public domain and users of public-domain software may expect support for it if it had a commercial origin. And, of course, former customers who paid for the application when it was marketed commercially may feel cheated if it becomes freely available.

The situation with respect to ProIcon was un-

usual. Both Catspaw, Inc. and The Bright Forest Company have contributed to the Icon public-domain effort. Catspaw, for example, contributed to the public-domain version of MS-DOS/386 Icon, and The Bright Forest Company contributes to this *Newsletter* and other freely available documentation about Icon. Both organizations felt that ProIcon was a good application that would better be available freely to the programming community than simply being destroyed. Incidentally, sales of ProIcon were discontinued well before it was placed in the public domain to avoid misleading potential customers.

Where do I get support for ProIcon?

You can direct questions to the Icon Project, but we can't promise to help you. A considerable amount of what you pay for a commercial software product goes toward potential support, which may include help with learning the application, debugging user programs, fixing application bugs, and even adding new features. For a public-domain application, any support that's available must be derived from other sources, such as the free contribution of a person's time. There's only so much you can expect without paying for it. The Icon Project does the best it can to support the public-domain software it distributes, but it has very limited resources. In the case of ProIcon, there's another problem. The Icon Project has no one with the knowledge to deal with technical issues in Macintosh applications. We cannot even recompile ProIcon, should a bug show up. On the other hand, ProIcon is a well-written application that has been in use for a long time. We've had very few requests for support since it was placed in the public domain.

Since The ProIcon application is now in the public domain, why isn't the source code in the public domain also?

The source code contains proprietary components that are used in commercial products that Catspaw, Inc. still markets. In addition, there are components of ProIcon that were licensed from a third party that is no longer in business. This does not mean that these components are no longer proprietary, but locating their present owner, much less getting permission to place them in the public domain, is not feasible.

Why hasn't ProIcon been brought up to a more current version?

There was a major change to the implementation of Icon after Version 8.0, which is the ProIcon version. Updating ProIcon to a later version would be a *major* effort for which very few persons are qualified. It also would require access to the source code for ProIcon (see the question and answer immediately preceding). It is extremely unlikely that ProIcon will be upgraded. On the other hand, Version 8.0 is very solid and there have been relatively few changes to the language since 8.0, except for the addition of graphics facilities.

I would like to have ProIcon rebuilt to not enqueue high-level events. Can you put me in touch with Catpsaw?

Sorry, no. Part of the agreement that placed ProIcon in the public domain was that Catspaw, Inc. would not be responsible for any future support or development of ProIcon.

Ordering Icon Material

See the latest *Newsletter* for current ordering information.