

Version 8 of Icon for CMS - Installation and Recompilation

Tim Hunter
SAS Institute, Inc.

1. INTRODUCTION

Version 8 of Icon for CMS should run on the IBM 30xx and 43xx families of processors and on other 370-type processors that use CMS under Release 4, 5, or 6 of VM/SP and VM/HPO. This version of Icon will also run under all releases of VM/XA SP, in either a 370-mode or an XA-mode virtual machine.

Version 8 of Icon for CMS is distributed on a 1600 BPI unlabeled tape which includes the MODULE files, object code, EXECs, test and example programs, data for the programs, documentation files, and the C source code. Printed documentation is included with tapes distributed by the Icon Project at the University of Arizona.

2. INSTALLATION FILES

The first tape file contains the following CMS files:

1. ICNT MODULE -- the Icon translator
2. ICONX MODULE -- the Icon executor
3. INSTALL LISTING -- a copy of this document
4. ICON LISTING -- a copy of the CMS Icon User's Guide
5. * ICN -- example Icon programs
6. * DAT -- input files for the example programs
7. REGIS ICONFORM -- Icon registration form
8. TROUBLE ICONFORM -- Icon trouble report form

The CMS files in the second tape file form the procedures and programs components of the Icon program library. The procedures component files have been assigned filemode number 1 and the programs component files have been assigned filemode number 2.

1. * ICN -- source for the programs component of the
Icon program library
2. * CSG -- input data for the CSGEN program
3. * KRS -- input data for the CROSS program
4. * LBL -- input data for the LABEL program
5. * LIN -- input data for the LINDEN program
6. * RSG -- input data for the RSG program
7. * SEN -- Farberisms and palindromic sentences
8. * TUR -- input data for the TURING program
9. * TXT -- input data for the MONKEYS program and
other text processing programs
10. * WRD -- word lists
11. * ICN -- the procedures component of the Icon pro-
gram library.

The CMS files in the third tape file are not needed unless you wish to recompile Icon.

1. CCICON EXEC -- an EXEC that compiles one or more Icon
source files
2. LINKICON EXEC -- an EXEC that links the Icon MODULE files
3. TESTICON EXEC -- an EXEC that executes one or more Icon
test programs
4. * C -- the C source code for Icon
5. * H -- the C header files for Icon
6. RSWITCH ASSEMBLE -- assembly-language coswitch function for
the Waterloo C implementation of Icon
7. * ICN -- the Icon test programs
8. * DAT -- the input data for the Icon test programs
9. * OUT -- the expected output from the Icon test
programs
10. * ERR -- the expected stderr output from the Icon
test programs

3. INSTALLING CMS ICON

Allocate and format a minidisk for the Icon files. The following table shows how many disk blocks are required, for minidisks that are formatted using 4K, 2K, or 1K disk blocks.

Number of minidisk blocks required

	Minidisk block size		
	4096	2048	1024
For the first tape file	275	526	1040
For the second tape file	302	493	835
For the third tape file	612	1019	1783

Total	1189	2037	3658

The tape was created with the CMS TAPE command. There are three physical tape files, each containing a number of logical CMS files. The physical files are separated by a single tape mark. The last physical file is followed by two tape marks.

The CMS files in the first tape file use filemode number 5 for the MODULEs and sample programs. The CMS files in the second tape file use filemode numbers 1 and 2. The CMS files in the third tape file use filemode number 0. If you install all of the files on the same minidisk, you may choose to access just the MODULEs and sample programs by issuing the command:

```
ACCESS vaddr fm/fm * * fm5
```

where 'vaddr' is the virtual address of the minidisk and 'fm' is the filemode letter.

Mount the tape at address 181 of your virtual machine. To install only the first tape file, issue the command

```
TAPE LOAD * * fm
```

where 'fm' is the filemode letter of the Icon minidisk.

To install all of the files on the tape, issue the command

```
TAPE LOAD * * fm (EOT
```

4. TESTING THE INSTALLATION

There are a number of Icon programs in the first tape file that can be used for testing the installation and getting a feel for running Icon:

HELLO ICN This program prints the Icon version number, identifies the host computer, prints the date and time, and lists the implemented Icon features. Run this test as

```
icont hello
iconx hello
```

KROSS ICN This program prints all the ways that two words intersect in a common character. The file KROSS DAT contains typical data. Run this test as

```
icont kross
iconx kross <kross.dat
```

MEANDER ICN This program prints the "meandering strings" that contain all subsequences of a specified length from a given set of characters. The file MEANDER DAT contains test data. Run this test as

```
icont meander
iconx meander <meander.dat
```

ROMAN ICN This program converts Arabic numerals to Roman numerals. Run this test as

```
icont roman -x
```

and provide some Arabic numbers from the terminal. Enter the word EOF to stop the program.

If these tests work, the installation is probably correct and it should be a running version of Icon.

5. COMPILING ICON FOR CMS

As of this date, version 8 of Icon for CMS has been successfully compiled with the SAS/C compiler, Release 4.50. The use of another C compiler will certainly involve some work, since there is some code that is conditional on the characteristics of specific C compilers. Additionally, Icon depends on non-ANSI UNIX features such as sbrk, which are often unavailable in mainframe C implementations. For some C compilers, it also may be necessary to make some compromises in the facilities that Icon supports.

If you wish to use another C compiler to compile Icon, you may want to request Technical Report TR 90-5 <1> from the Icon Project. This document describes in detail the steps that are necessary to port Icon to a new system or compiler, and to validate the results.

The third file on the installation tape contains two EXECs to assist in recompilation and relinking of Icon using SAS/C. These EXECs are: CCICON, to recompile either a single Icon source file, or all Icon source files, and LINKICON, to link-edit the translator or executor. These EXECs assume that certain options, such as the NOCOMNEST option, are not changed from the default by GLOBAL variables or site modifications to the SAS/C EXECs. You may need to modify the source of CCICON to override such local defaults.

Note that if you compile Icon without modification, you will need the SAS/C "all-resident" library at link time. The use of the all-resident library in Icon is solely to simplify distribution and installation procedures. If you recompile Icon, it is recommended that you modify TLOCAL C and RLOCAL C to remove the use of this feature, as this will produce smaller load modules. (See the SAS/C User's Guide <2> for further information on the restrictions associated with the all-resident library.)

If you are using release 4.50 of the SAS/C compiler, you may need to obtain a update all-resident library TXTLIB. If your site has not already installed the updated version of LCARES TXTLIB, you can obtain it from SAS Institute Technical Support by calling (919) 677-8008 from 9:00 AM to 8:00 PM Eastern time. You should ask for information about usage note C.REDISTRIB-5746.

A suite of test programs is provided on the third tape file. These files all have a filetype of ICN. The test input files have a filetype of DAT, and the expected output files have a filetype of OUT. Many tests contain errors, or generate traces. The expected error output files have a filetype of ERR. The TESTICON EXEC is provided to rerun one or all of the tests. A few tests depend on time of day or other local conditions, so the results may not be identical for all tests even assuming a completely correct implementation.

Previous versions of Icon have been implemented using the Waterloo C compiler. If you wish to try using the Waterloo C compiler to compile the Icon source, you may want to examine the assembly-language file RSWITCH ASSEMBLE. This file implements coexpression context switching for the Waterloo run-time environment. (When using the SAS/C compiler to compile Icon, the RSWITCH C source file contains this function.) Note that the CCICON and LINKICON EXECs cannot be used with the Waterloo C compiler unless they are modified.

6. THE IMPLEMENTATION BOOK

If you are interested in the larger view of the implementation of Icon, or if you are interested in extending or modifying Icon, you may want to acquire the book The Implementation of the Icon Programming Language. This book concentrates on the run-time system and covers data structures, the virtual machine, the interpreter, the implementation of generators, and storage management. It also contains material specifically related to making modifications to the source code.

The publication information is: The Implementation of the Icon Programming Language, by Griswold and Griswold, Princeton University Press, ISBN 0-691-08431-9, hardbound, 336 pages, \$39.50. The book may be ordered from the Icon Project, a local bookstore, or directly from the publisher:

Princeton University Press
3175 Princeton Pike
Lawrenceville, NJ 08648

(609) 896-1344

The implementation book corresponds to Version 6.2 of the Icon source code. There have been several changes in the source code between Version 6.2 and the present version. A report describing these changes is available free of charge from the Icon Project <2>. Ask for IPD112.

7. THE ICON-C INTERFACE

Version 8 of Icon for CMS supports calling C subroutines from an Icon program, and calling the Icon executor from a C program. See Technical Report TR90-8 <4> for further information on use of this feature.

Two examples of the use of the interface are provided in the Icon sample source in the third tape file. SPFCALL C is a version of the EXTCALL function which implements an interface to ISPF, in conjunction with the Icon procedures contained in ISPF ICN. REXXCALL C is a version of the EXTCALL function which implements an interface to Rexx in conjunction with REXX ICN.

The ICONX MODULE distributed on the tape includes the Rexx interface as a standard feature.

8. REPORTING PROBLEMS

Problems with CMS Icon should be noted on a trouble report form (TROUBLE ICONFORM on the distribution tape) and sent to

Icon Project
Department of Computer Science
Gould-Simpson Building
The University of Arizona
Tucson, AZ 85721
U.S.A.
(602) 621-2018
icon-project@arizona.edu (Internet)
... {uunet, allegra, noao}!arizona!icon-project (uucp)

If a program is involved, a copy of the program will be appreciated. The program may be necessary to provide help.

8. REGISTERING COPIES OF ICON

Those who received a copy of Version 8 of Icon for CMS directly from the Icon Project are registered users and will receive the Icon Newsletter without charge. This Newsletter contains information about new implementations, updates, programming techniques, and information of general interest about Icon.

Those who received a copy of Version 8 of Icon for CMS from a source other than the Icon Project should fill out a registration form (REGIS ICONFORM on the distribution tape) and send it to the Icon Project at the address listed above. This will entitle them to a free subscription to the Icon Newsletter and assure that they receive information about updates.

REFERENCES

1. R. E. Griswold Transporting Version 8 of Icon, The Univ. of Arizona Tech. Rep. 90-5, 1990.
2. SAS/C Compiler and Library User's Guide, Second Edition.
3. R. E. Griswold, Supplementary Information for the Implementation of Version 8 of Icon, The Univ. of Arizona Icon Project Document IPD112, 1990.
4. R. E. Griswold, Icon-C Interfaces The Univ. of Arizona Tech. Rep. 90-8, 1990.