

Introducing SAS/C Software

Release 6.50



SAS Institute Inc.
SAS Campus Drive
Cary, NC 27513

The correct bibliographic citation for this manual is as follows: SAS Institute Inc., *Introducing SAS/C® Software*, Cary, NC: SAS Institute Inc., 1998. 25 pp.

Introducing SAS/C® Software

Copyright © 1998 by SAS Institute Inc., Cary, NC, USA.

ISBN 1-58025-132-3

All rights reserved. Printed in the United States of America. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, or otherwise, without the prior written permission of the publisher, SAS Institute Inc.

Restricted Rights Legend. Software and accompanying documentation are provided to the U.S. government in a transaction subject to the Federal Acquisition Regulations with Restricted Rights. Use, duplication, or disclosure of the software by the government is subject to restrictions as set forth in FAR 52.227-19 Commercial Computer Software-Restricted Rights (June 1987). The Contractor/Licenser is SAS Institute Inc., located at SAS Campus Drive, Cary, North Carolina 27513.

SAS Institute Inc., SAS Campus Drive, Cary, North Carolina 27513.

1st printing, March 1998

The SAS® System is an integrated system of software providing complete control over data access, management, analysis, and presentation. Base SAS software is the foundation of the SAS System. Products within the SAS System include SAS/ACCESS®, SAS/AF™, SAS/ASSIST®, SAS/CALC®, SAS/CONNECT®, SAS/CPE®, SAS/DMI®, SAS/EIS®, SAS/ENGLISH®, SAS/ETS®, SAS/FSP®, SAS/GRAPH®, SAS/IML®, SAS/IMS-DL/I®, SAS/INSIGHT®, SAS/IntrNet™, SAS/LAB®, SAS/MDDB™, SAS/NVISION™, SAS/OR®, SAS/PH-Clinical®, SAS/QC®, SAS/REPLAY-CICS®, SAS/SESSION®, SAS/SHARE®, SAS/STAT®, SAS/TOOLKIT®, SAS/TUTOR®, SAS/Warehouse Administrator™, SAS/DB2™, SAS/GEO™, SAS/GIS®, SAS/PH-Kinetics™, SAS/SHARE*NET™, SAS/SPECTRAVIEW™, and SAS/SQL-D™ software. Other SAS Institute products are SYSTEM 2000® Data Management Software, with basic SYSTEM 2000, CREATE™, Multi-User™, QueX™, Screen Writer™, and CICS interface software; InfoTap® software; JMP®, JMP IN®, and JMP Serve® software; SAS/RTERM® software; the SAS/C® Compiler; Video Reality™ software; Warehouse Viewer™ software; Budget Vision™, Campaign Vision™, CFO Vision™, Enterprise Miner™, Enterprise Reporter™, HR Vision™, IT Charge Manager™ software, and IT Service Vision™ software; Scalable Performance Data Server™ software; SAS OnlineTutor™ software; and Emulus® software. MultiVendor Architecture™, MVA™, MultiEngine Architecture™, and MEA™ are trademarks of SAS Institute Inc. SAS Institute also offers SAS Consulting® and SAS Video Productions® services. *Authorline*®, Books by UsersSM, The Encore Series®, *ExecSolutions*®, *JMPer Cable*®, *Observations*®, *SAS Communications*®, *sas.com*™, *SAS OnlineDoc*™, *SAS Professional Services*™, the SASware Ballot®, *SelecText*™, and *Solutions@Work*™ documentation are published by SAS Institute Inc. The SAS Video Productions logo, the Books by Users SAS Institute's Author Service logo, the SAS Online Samples logo, and The Encore Series logo are registered service marks or registered trademarks of SAS Institute Inc. The Helplus logo, the SelecText logo, the Video Reality logo, the Quality Partner logo, the SAS Business Solutions logo, the SAS Rapid Warehousing Program logo, the SAS Publications logo, the Instructor-based Training logo, the Online Training logo, the Trainer's Kit logo, and the Video-based Training logo are service marks or trademarks of SAS Institute Inc. All trademarks above are registered trademarks or trademarks of SAS Institute Inc. in the USA and other countries. ® indicates USA registration.

IBM and System/370 are registered trademarks or trademarks of International Business Machines Corporation. ® indicates USA registration.

Other brand and product names are registered trademarks or trademarks of their respective companies.

Contents

Chapter 1 △ **SAS/C Software Overview** 1

| | |
|--|---|
| SAS/C Compiler and Library | 1 |
| SAS/C Debugger | 2 |
| SAS/C CICS Command Language Translator | 3 |
| SAS/C Connectivity Support Library | 4 |
| SAS/C Full-Screen Support Library | 5 |
| SAS/C C++ Development System | 6 |
| SAS/C C++ Foundation Libraries | 6 |
| SAS/C Cross-Platform Compiler | 7 |
| Resident Library Product | 8 |

Chapter 2 △ **Highlights of SAS/C Release 6.50** 9

| | |
|---------------------------------|----|
| Introduction | 9 |
| C++ Foundation Libraries | 9 |
| Compiler and Library Changes | 10 |
| C++ Translator Changes | 12 |
| Cross-Platform Compiler Changes | 13 |

Chapter 3 △ **SAS/C Documentation** 15

| | |
|----------------------|----|
| Online Documentation | 15 |
| Books | 16 |

Chapter 4 △ **Contacting SAS Institute** 19

| | |
|------------------------------------|----|
| SAS/C Software Sales and Marketing | 19 |
| SAS/C Customer Service Department | 19 |
| SAS/C Technical Support | 19 |
| SAS Publications | 20 |
| World Wide Web Pages | 20 |

Index 22

SAS/C Software Overview

| | |
|---|---|
| <i>SAS/C Compiler and Library</i> | 1 |
| <i>SAS/C Debugger</i> | 2 |
| <i>SAS/C CICS Command Language Translator</i> | 3 |
| <i>SAS/C Connectivity Support Library</i> | 4 |
| <i>SAS/C Full-Screen Support Library</i> | 5 |
| <i>SAS/C C++ Development System</i> | 6 |
| <i>SAS/C C++ Foundation Libraries</i> | 6 |
| <i>SAS/C Cross-Platform Compiler</i> | 7 |
| <i>Resident Library Product</i> | 8 |

SAS/C Compiler and Library

The SAS/C Compiler provides a versatile C development environment for IBM® System/370™ mainframes. The features of the compiler and library include full compliance with the ANSI standard for the C programming language, and a wealth of library extensions that enable you to fully exploit the power of the mainframe. The SAS/C Compiler has over a decade of proven reliability in the demanding development environments of business, industry, education, and government. One indication of the power and flexibility of the SAS/C Compiler is that SAS Institute has relied on it to engineer the SAS System for Information Delivery.

The primary elements of the SAS/C Compiler are the compiler, debugger and the run-time library. Additionally, the compiler product includes a number of utility programs, as well as several configurations of the run-time library for special environments such as the C Systems Programming Environment (SPE), which allows supervisor calls (SVCs), operating system exits and other routines that are traditionally written in assembler language to be written in C. Although the compiler is heavily oriented for use in large software systems, it is an efficient tool for any software project that is written in the C language.

Traditionally, SAS/C applications have been developed in one of the System/370 operating environments: TSO, CMS, or MVS batch. With the introduction of the OpenEdition subsystem from IBM, the SAS/C Library supports the operating system interface defined by the International Standard Organization (ISO) POSIX 1003.1 standard. This support extends the mainframe development environment to the OpenEdition shell under MVS.

The SAS/C Library also provides TCP/IP support with a complete implementation of the Berkeley Software Distribution (BSD) UNIX Socket Library interface. The socket library is integrated with SAS/C support for UNIX file I/O to provide the same type of integration between file and network I/O that is available on BSD UNIX systems.

In addition to ISO/ANSI C, traditional UNIX, and POSIX support, the SAS/C Library includes the following special features:

- dynamic loading support
- inline machine code interface
- CMS low-level I/O functions
- MVS low-level I/O functions
- low-level system interfaces
- low-level MVS multitasking functions
- inter-user communications vehicle (IUCV) functions
- advanced program-to-program communication/virtual machine (APPC/VM) functions
- advanced signal handling, including support for true asynchronous signals and for user-defined signals
- the subcommand interface to EXECs and CLISTs
- the CMS REXX SAS/C interface.

The interlanguage communication features and the inline machine code interface further extend the capabilities of your SAS/C development projects by enabling you to write complicated multilanguage applications or incorporate machine instructions that cannot normally be generated with a high-level language.

In addition to providing you with an exceptionally powerful C development system, SAS Institute wants to make the distribution of your applications affordable. To facilitate this distribution, you may need to redistribute certain run-time files provided by SAS Institute. Many of these SAS/C programs and libraries are contained in the SAS/C Limited Distribution Library, which is redistributable at no charge. Additional programs and libraries may be redistributed by licensing the SAS/C Redistribution Package. No royalties or notification need be reported to SAS Institute upon your authorized redistribution of these run-time files.

SAS/C Debugger

The SAS/C Debugger, which is included with the base SAS/C product, offers a flexible, easy-to-use full screen user interface that supports multiple overlapping color windows on an IBM 3270, or similar, terminal display system.

The following are a few of the many features offered by the extensive command set of the SAS/C Debugger:

- Programmer controlled breakpoints can be set in an executing program. When a breakpoint is reached, control of execution transfers to the debugger. Breakpoints

can be set conditionally and over a range of source lines. A group of commands can be automatically executed at a breakpoint.

- Program execution can be single-stepped, and program flow can be traced.
- Values can be assigned to objects, and values and storage contents can be displayed.
- Data objects can be checked for arbitrary changes in value using the Watch window.
- Stack and heap storage areas can be examined to check for storage overlays, with the results being displayed in a debugger window.
- Terminal I/O can be captured and diverted to a debugger window, allowing program output to be browsed in coordination with debugger output.
- User I/O exits may be specified for specialized debugging needs.

The debugger also supports remote debugging, which enables the debugger and the program that is being debugged to execute in different address spaces or environments. Remote debugging can be used to debug programs running under CICS or the OpenEdition shell.

If you are developing applications in a cross-compilation environment, you can use a distributed file system to access include, source, and debugger files that may reside on a workstation or PC. This feature saves you the trouble of having to maintain duplicate source files on your cross-development platform and on the mainframe.

SAS/C CICS Command Language Translator

The SAS/C CICS Command Language Translator is another powerful tool that is included with the base SAS/C product. The CICS translator enables you to develop application programs under MVS or VM and target them to run under CICS. This application-programming interface enables you to request CICS services by placing CICS commands anywhere within your C or C++ source code. The SAS/C CICS translator translates these commands into appropriate function calls for communication with CICS. All CICS commands are supported, including Distributed Program Link (DPL) commands and Front End Programming Interface (FEPI) commands.

The following are some of the unique features offered by the SAS/C CICS run-time library:

- CICS/ESA Version 5.1 and all previous releases back to CICS/VS 1.7 are supported on MVS. In addition, CICS/VSE Version 2.3 and all previous releases back to CICS/VSE1.7 are supported on VSE systems.
- Support for the CICS Systems Programming Environment allows CICS systems programs to be written in C.
- All-resident support allows a module to be built containing all the transient library routines needed by the program.
- I/O support for JES spool files enables you to retrieve a file from the JES spool, write a file directly to the JES spool, and send a file to a remote destination using standard C library I/O functions.
- Dynamic loading support provides compatibility with MVS and CMS.
- For applications programs link-edited as AMODE 31, the C stack and heap can be allocated above the 16Mb line.
- Most of the SAS/C run-time library for CICS can reside above the 16Mb line.

SAS/C Connectivity Support Library

The SAS/C Connectivity Support Library (CSL) is an add-on product that facilitates network communications between SAS/C programs and other applications. CSL is actually a set of libraries and related utilities that support the use of several popular application protocols for distributed (client/server) computing. CSL operates in a heterogeneous computing environment using TCP/IP, and it is designed to ease the development of mainframe programs that interact with a wide range of other computer systems, including PCs, workstations, large servers, and supercomputers.

The SAS/C CSL libraries provide the programming support for the three most popular application protocols in a TCP/IP network environment:

- X Window System
- Sun Remote Procedure Call (RPC)
- Sun Network File System (NFS) protocols.

CSL also provides a **rexec** client function, some Simple Network Management Protocol (SNMP) agent customization functions, and a few supporting POSIX functions.

SAS/C CSL has several important advantages:

- It is designed to work with SAS/C. Thus, a large body of existing applications can be enhanced to use these protocols. Standard SAS/C techniques are used for compiling, linking, and running CSL programs.
- UNIX compatibility is high. The SAS/C CSL design has placed a high priority on running existing UNIX programs with few or no changes. This design goal also maximizes portability among implementations of other operating systems such as DOS and OpenVMS.
- Because SAS/C CSL uses the SAS/C socket library for TCP/IP, programs that are written with the SAS/C CSL will run on a TCP/IP implementation that works with

the SAS/C socket library. (The TCP/IP products developed by IBM, Fibronics, and Interlink are examples of such implementations.) Therefore, software developers can build applications without regard for the TCP/IP vendor in use at a customer site.

- As with the base SAS/C product, the run-time components of SAS/C CSL are freely redistributable. This means that applications written with CSL can be run at other sites without requiring those sites to license SAS/C or CSL.

SAS/C Full-Screen Support Library

Another add-on product, the SAS/C Full-Screen Support Library, provides a high-level interface for developing full-screen applications on IBM mainframe systems.

Traditionally, programs involving terminal communications had to be developed using either structured screen management packages or complicated low-level commands to construct, send, and receive data streams (the data transferred between the terminal and the application program). SAS/C FSSL functions free you from handling details of IBM terminal communication, enabling more focus and flexibility in full-screen applications development.

The primary capabilities provided by the SAS/C FSSL include:

- provision of a high-level interface for IBM data stream programming, freeing the programmer from IBM 3270 data stream details
- the SAS/C Windowing Interface, which manages the overhead associated with controlling multiple windows in an IBM 3270 environment
- generic or abstract screen manipulation
- portability between MVS and CMS
- IBM 3270 device independence.

Applications for the Full-Screen Support Library include:

- data entry and verification programs
- development of full-screen editor software
- development of logon screens and screens to introduce users to applications packages
- development of system maintenance and update screens (for example, selection and display of disk data sets for management)
- file updates (reading, displaying, and updating selected fields)
- menu development for software packages
- assistance in system prototyping by producing sample screens
- research and modeling applications where parameters are entered and results displayed interactively.

SAS/C C++ Development System

The SAS/C C++ Development System is an add-on product that extends the capabilities of the base SAS/C product, enabling you to develop and run C++ programs on the mainframe under TSO, CMS, MVS batch, the OpenEdition shell, and CICS. The system implements the C++ language by means of a translator that translates C++ to C. The translated C code must be compiled with the SAS/C Compiler, resulting in an object module. By default, when you invoke the translator, both the translation and compilation are performed; that is, you do not have to call the SAS/C Compiler as a separate step.

The C++ language accepted by the SAS/C C++ Development System is generally compatible with that specified by Bjarne Stroustrup in the *The C++ Programming Language, Second Edition*, except that several ANSI C++ features have been added and exception handling has not been implemented. Code written for previous releases of the SAS/C C++ Development System will also generally be accepted by this release. However, this release has increased compatibility with the ANSI/ISO C++ draft standard by implementing new features and by tightening language rules for certain obscure or unsafe constructs.

Release 6.50 of the SAS/C C++ Development System includes these new features:

- template support
- new cast operators
- run-time type identification
- overload resolution rules updated for clarity.

If you have a C++ program that is compliant with AT&T C++ version 3.0, your code should generally work with the SAS/C C++ Development System.

If you have a C++ program that is compliant with AT&T C++ version 2.1, with the exception of some features that are now considered outdated, your code should generally work with the SAS/C C++ Development System. The main differences between AT&T C++ 2.1 and the SAS/C C++ Development System are the following:

- SAS/C C++ does not support overloading both the prefix and postfix ++ or – operators with a single function.
- SAS/C C++ does not support the transitional model of nested classes. That is, a class that is nested is always treated as nested, even if there is not a file-scope class of the same name.

SAS/C C++ Foundation Libraries

The SAS/C C++ Foundation Libraries is an add-on product designed to extend the capabilities of the SAS/C C++ Development System. This product was built using two of Rogue Wave's popular C++ packages: the Standard C++ Library and Tools. h++. The SAS/C C++ Foundation Libraries provide a custom build of Rogue Wave's implementation of the emerging ISO/ANSI Standard C++ Library, including the Standard Template Library. Tools.h++ builds on the Standard C++ Library, providing a powerful collection of production-proven classes.

The Standard C++ Library is currently being developed by ISO/ANSI standards committees and within a few years all ANSI-conforming C++ compilers will deliver a standard set of classes and libraries. The Rogue Wave Standard C++ Library avoids encapsulation and uses almost no inheritance, making it a highly efficient tool that can be used whenever performance is a major concern. The Tools.h++ package, which is also built on the Rogue Wave Standard C++ Library, encapsulates many of the features of that library so that they may be used with complex projects that require sophisticated object-oriented programming techniques.

SAS/C Cross-Platform Compiler

The SAS/C Cross-Platform Compiler and the SAS/C Cross-Platform C++ Development System run on a workstation and produce prelinked output files that can be transferred to an IBM System/370 mainframe. On the mainframe, the files can then be linked to produce an executable load module.

Like the mainframe versions of the SAS/C Compiler and C++ Development System, the cross-platform products are a portable implementation of the high-level C and C++ languages. The SAS/C Cross-Platform Development System provides the same functionality under the UNIX or Windows NT operating systems as the SAS/C Compiler and the C++ Development System does under the MVS or CMS operating systems.

The features of the SAS/C Cross-Platform Development System include the following:

- All SAS/C and C++ preprocessor capabilities.
- All SAS/C and C++ code generation capabilities, including optimization and other features.
- All of the prelinking functionality provided by the SAS/C COOL utility, including extended names processing.
- An object module disassembler that can be used to generate an object module disassembly listing file.
- The capability of producing object modules suitable for debugging with the SAS/C Debugger.
- Support for the SAS/C CICS preprocessor.
- Support for portable AR370 archive libraries.

There are several benefits to using the SAS/C Cross-Platform Compiler and C++ Development System to generate code for the mainframe.

Reduced mainframe load

By moving compilations off the mainframe, mainframe CPU cycles are preserved for other users. This can amount to a significant reduction in mainframe requirements, directly translating into a cost savings.

Improved source management

Developers may take advantage of improved source management tools, as well as the UNIX hierarchical file system.

Improved build management

Developers may take advantage of improved build management tools, such as **make** files.

Improved compilation turnaround

In a heavy development environment, developers often find that performing compilations locally can result in better turnaround time.

Resident Library Product

The SAS/C Resident Library Product is a subset of the SAS/C Compiler and Library. This product enables sites to develop C programs for both the MVS and CMS platforms without having a full, licensed version of the SAS/C Compiler and Library on both platforms. The SAS/C Resident Library Product includes portions of the resident and transient libraries as well as the COOL and CLINK utilities so that you can link and run your programs on the platform that does not have the full version of the SAS/C Compiler and Library. You cannot, however, compile your programs on the platform that has only a SAS/C Resident Library Product license. Compilation must be performed on the platform that has the full SAS/C Compiler and Library license.

Highlights of SAS/C Release 6.50

| | |
|--|-----------|
| <i>Introduction</i> | 9 |
| <i>C++ Foundation Libraries</i> | 9 |
| <i>Compiler and Library Changes</i> | 10 |
| <i>Enhancements</i> | 10 |
| <i>New OpenEdition Functions</i> | 10 |
| <i>New OpenEdition Signals</i> | 12 |
| <i>Additional Library Support</i> | 12 |
| <i>C++ Translator Changes</i> | 12 |
| <i>Cross-Platform Compiler Changes</i> | 13 |

Introduction

SAS/C Release 6.50 introduces several significant changes and enhancements to the compiler, library, C++ translator, cross-platform compiler, and the debugger. Only the highlights are presented in this chapter. For detailed documentation, you should refer to *SAS/C Software: Changes and Enhancements, Release 6.50* and the *SAS/C C++ Development System User's Guide, Release 6.50*.

C++ Foundation Libraries

The SAS/C C++ Foundation Libraries extend the capabilities of the SAS/C C++ Development System, providing a custom build of Rogue Wave's implementation of the emerging ISO/ANSI Standard C++ Library.

Compiler and Library Changes

Compiler enhancements, fifty new functions, and ten new signals have expanded the capabilities of the SAS/C Compiler and Library.

Enhancements

The compiler and library enhancements include the following:

- Stack space can now be allocated above the line.
- Partitioned Data Set (PDS) sharing for file I/O has been improved.
- VSAM Record Level Sharing (RLS) is now supported.
- C++ style comments in C code are now supported by the SAS/C Compiler.
- International Standard Organization (ISO) digraph translation is now supported in addition to the SAS/C digraph extensions.
- Character string qualifiers enable you to specify either ASCII or EBCDIC character strings.
- Compiler support for the at sign (@) has been extended to operate in the same contexts as the ampersand (&).

New OpenEdition Functions

The following OpenEdition SAS/C Library Functions have been added for Release 6.50:

- chpriority** — Change process priority
- endgrent** — Close the group database
- endpwent** — Close the user database
- extlink** — Define an external link
- getgrent** — Access group database sequentially
- getitimer** — Obtain interval timer values
- getpgid** — Get process group id for a process
- getpriority** — Determine process priority
- getpwent** — Access user database sequentially
- getrlimit** — Obtain OpenEdition resource limits
- getrusage** — Obtain OpenEdition usage statistics
- getsid** — Get session leader id for a process
- getwd** — Get working directory pathname
- lchown** — Change file or link ownership
- mmap** — Map an HFS file into memory

mprotect — Change memory protection mapped to file

msgctl — Control a message queue

msgget — Create or find a message queue

msgrcv — Receive a message from a message queue

msgsnd — Send a message to a message queue

msgxrcv — Receive message with sender information

msync — Synchronize a memory mapped HFS file

munmap — Cancel mapping of a memory area to a file

oetaskctl — Control subtasks with OpenEdition

readextlink — Read an external link

realpath — Return absolute pathname

semctl — Control a semaphore set

semget — Create or find a set of semaphores

semop — Update semaphores automatically

setgrent — Reposition the group database

setitimer — Define an interval timer

setpriority — Change process priority

setpwent — Reposition the user database

setregid — Set real and/or effective group ID

setreuid — Set real and/or effective user ID

setrlimit — Define OpenEdition resource limits

shmat — Attach a shared memory segment

shmctl — Control a shared memory segment

shmdt — Detach a shared memory segment

shmget — Create or find a shared memory segment

sigblkjmp — Intercept **longjmp** without signal mask change

spawn — Spawn a new process

spawnp — Spawn a new process

tcgetsid — Get session leader id for a process

truncate — Truncate an HFS file.

Several OpenEdition functions that were provided with SAS/C Release 6.00, **alarmd**, **blkjmp**, **siglongjmp**, **sleepd**, and **tcsetpgrp**, have been enhanced to provide additional capabilities for Release 6.50.

New OpenEdition Signals

Support for a number of new signals that are defined by OpenEdition MVS in Release 5.2.2 and later has been added to SAS/C Release 6.50. (Many of these signals are defined for compatibility with UNIX operating systems but do not have a defined meaning under MVS. Consult UNIX documentation for further information on these signals.) The new signals and their meanings are as follows:

- SIGBUS** — UNIX compatibility
- SIGIOER** — IBM C compatibility
- SIGPOLL** — UNIX compatibility
- SIGPROF** — Profiling timer expired
- SIGSYS** — UNIX compatibility
- SIGURG** — UNIX compatibility
- SIGVTALRM** — Virtual timer expired
- SIGWINCH** — UNIX compatibility
- SIGXCPU** — CPU time limit exceeded
- SIGXFSZ** — File size limit exceeded.

Additional Library Support

In addition to the enhancements to our OpenEdition support, the following functions have been added or enhanced:

- DOM** — New Delete Operator Message function.
- DOM_TOK** — New Delete Operator Message with Tokens function.
- WTO** — New Write to Operator function.
- WTOR** — New Write to Operator with Reply function.
- addsrch** — Enhanced to provide new MVS search capability.

C++ Translator Changes

The following enhancements to the SAS/C C++ Development System have been implemented for Release 6.50:

- Templates allow you to define a generic family of types or functions by creating parameterized types. Templates are particularly useful for providing general

container types such as stacks, lists, and vectors. Class templates are used to create a family of classes. Function templates are used to create an unbounded set of overloaded functions.

- Run-Time Type Identification (RTTI) allows the actual run-time type of an object to be determined. The **typeid** operator is used to identify types and access type information. The **dynamic_cast** operator allows you to perform a cast with run-time type checking. An important aspect of the **dynamic_cast** operator is the ability to implement casts from a virtual base class to a derived class, which is impossible with other cast operators.
- Three new ANSI C++ cast operators offer improved type checking over the old style cast operator. The **static_cast** operator is primarily used for conversions such as casting from a base pointer to a derived pointer. The **reinterpret_cast** operator is used for typical implementation-dependent conversions such as pointer to unrelated pointer, integral type to pointer type, and pointer type to integral type. The **const_cast** operator is used to change the const and volatile qualifiers on pointers or references. Any old style cast can be expressed as a combination of these new casts.

Cross-Platform Compiler Changes

The SAS/C Cross-Platform Compiler and the SAS/C Cross-Platform C++ Development System now supports the Windows NT platform, extending the cross-platform development environment to Windows NT workstations.

SAS/C Documentation

Online Documentation 15

Books 16

Online Documentation

With Release 6.50, we are continuing our move to online documentation. You will find the following documentation on a CD-ROM titled *SAS/C OnlineDoc, Release 6.50*, which is included with the SAS/C software distribution package:

- *Introducing SAS/C Release 6.50*
- *SAS/C Compiler and Library User's Guide, Fourth Edition, Release 6.00*
- *SAS/C C++ Development System User's Guide, Release 6.50*
- *SAS/C Software Diagnostic Messages, Release 6.50*
- *SAS/C Software: Changes and Enhancements, Release 6.50*
- *A Guide for the SAS/C Compiler Consultant, Release 6.50*
- *SAS/C Library Reference, Third Edition, Volume 1, Release 6.00*
- *SAS/C Library Reference, Third Edition, Volume 2, Release 6.00*
- *SAS/C Cross-Platform Compiler and C++ Development System: Usage and Reference, First Edition, Release 6.00*
- SAS/C C++ Foundation Libraries
 - *Standard C++ Library User's Guide and Tutorial*
 - *Standard C++ Library Class Reference*
 - *Tools.h++ User's Guide*
 - *Tools.h++ Class Reference*

The CD is readable on PC, Macintosh, and UNIX platforms. It is organized as indicated below:

| Directory | Contains |
|-------------|---|
| overview | This book. |
| changes | The Release 6.50 changes and enhancements technical report. |
| cplus | C++ Development System user's guide. |
| diag | Diagnostic messages manual. |
| techsup | The compiler consultant's guide. |
| lr1 and lr2 | The library reference manual. |
| cross | The cross-platform compiler user's guide. |
| postscript | Postscript files for each online book. |
| rogue | Documentation for the SAS/C C++ Foundation Libraries |
| clug | The compiler and library user's guide. |

The directories and files contained on the CD can be installed on a web server at your site or viewed from a stand-alone PC or workstation. The **index.htm** file is your entry point to the SAS/C online documentation, and it contains links to each of the books. You can open this file in any HTML 3.2 compatible browser.

You are authorized to print copies of these books, from the files in the **postscript** subdirectory, for use at your licensed SAS/C site. These books are also available for purchase from SAS, as described in the next section.

Books

In addition to the online documentation provided with the SAS/C software, you can order the following books:

Introducing SAS/C Software, Release 6.50 (order #A55936)

This introductory booklet is intended to give you a quick overview of SAS/C software products.

SAS/C Compiler and Library User's Guide, Fourth Edition, Release 6.00 (order #A55156)

Explains how to compile, link, and execute SAS/C programs under TSO, CMS, MVS batch, and the OpenEdition shell. This user's guide should be your starting point when learning how to use the SAS/C Compiler and Library and it will be a valuable source of information after you are familiar with the product.

SAS/C Library Reference, Third Edition, Volume 1, Release 6.00 (order #A55049)

The first of two books that comprise the authoritative reference to the SAS/C Library. Volume 1 describes all the commonly used C functions and provides a detailed overview of the SAS/C Library's organization. It also provides extensive

coverage of key topics such as signal handling, input/output facilities, and environmental variables.

SAS/C Library Reference, Third Edition, Volume 2, Release 6.00 (order #A55178)

The second volume of the library reference describes all the special purpose features of the SAS/C Library such as low-level I/O and IUCV functions. This volume also provides a complete reference for the SAS/C Socket Library and the POSIX functions.

SAS/C Debugger User's Guide and Reference, Third Edition (order #A56120)

Intended for both new and experienced debugger users, this book documents the SAS/C Debugger. The guide provides an extensive tutorial and essential reference material.

SAS/C CICS User's Guide, Second Edition, Release 6.00 (order #A55117)

This user's guide provides essential information for programmers who are developing CICS applications with the SAS/C Compiler.

SAS/C Full-Screen Support Library User's Guide, Second Edition (order #A56124)

This reference book provides all the information you will need to exploit the powerful SAS/C window interface to develop full-screen windowing applications in the IBM 3270 environment.

SAS/C Compiler Interlanguage Communication Feature User's Guide (order #A5684)

This user's guide explains how your SAS/C programs can communicate with programs written in other high-level languages such as FORTRAN, COBOL, PL/1, and Pascal.

SAS/C C++ Development System User's Guide, Release 6.50 (order #A55694)

This user's guide is the essential reference for the SAS/C C++ Development System. It explains how to compile C++ programs and also includes: detailed discussions of the basics of the product, such as header files and options; environment-specific details for using the Development System under each of the four supported environments (TSO, CMS, MVS batch, and the OpenEdition shell); information on how to debug C++ files using the SAS/C debugger.

SAS/C Cross-Platform Compiler and C++ Development System: Usage and Reference, First Edition, Release 6.00 (order #A55388)

Explains how to develop mainframe C and C++ applications from a UNIX workstation. This user's guide provides complete reference information for the SAS/C Cross-Platform Compiler and the SAS/C Cross-Platform C++ Development System.

SAS/C Software Diagnostic Messages, Release 6.50 (order #A56466)

Documents all the diagnostic messages generated by the various SAS/C software products, including error and warning messages.

SAS/C Compiler and Library Quick Reference Guide, First Edition, Release 6.00 (order #A55182)

This book provides handy reference tables for compiler and linker options, debugger commands, and library functions.

SAS/C Software: Changes and Enhancements, Release 6.50 (order #A55693)

Documents all the release 6.50 changes and enhancements that have not been incorporated into books that have been updated for Release 6.50. This technical report also included the documentation for the Release 5.50 and Release 6.00 changes to the SAS/C Debugger.

A Guide for the SAS/C Compiler Consultant, Release 6.50 (order #A55937)

This guide provides the essential information for installing software zaps, contacting and working with the SAS/C Technical Support Department, and using the SAS/C Web page and electronic support facilities.

SAS Technical Report C-113, *SAS/C Connectivity Support Library, Release 1.00* (order #A59018)

Documents the SAS/C Connectivity Support Library, which provides an interface for SAS/C programs to the X Window System, the Network File System (NFS), Remote Procedure Call (RPC), and Simple Network Management Protocol/Distributed Program Interface (SNMP DPI).

SAS Technical Report C-115, *The Generalized Operating System Interface for the SAS/C Compiler Run-Time System, Release 5.50* (order #A59025)

Documents the Generalized Operating System (GOS) features of the SAS/C Compiler, which enable you to write interface routines for the MVS and CMS operating systems.

To order, call 1-800-727-3228. (Customers outside the United States, contact your local SAS office, or order from our Web site: www.sas.com/pubs .)

Contacting SAS Institute

SAS/C Software Sales and Marketing 19

SAS/C Customer Service Department 19

SAS/C Technical Support 19

SAS Publications 20

World Wide Web Pages 20

SAS/C Software Sales and Marketing

To find out more about the SAS/C Compiler products, pricing, or to get a free 30-day trial, please contact your SAS/C Compiler Account Representative at SAS Institute (919) 677-8000.

As a user of SAS/C, the Marketing Department will periodically send a survey to your site which, when completed and returned, will provide valuable feedback to us. Please respond to these surveys, either by FAX or by email, in order to help us plan for future enhancements.

SAS/C Customer Service Department

In order to redesignate a CPU or change the contact names for those people at your site who receive SAS/C software or invoices, please call the Customer Service Department at (919) 677-8003.

SAS/C Technical Support

SAS Institute Inc. provides its SAS/C customers with free technical support, offered directly to your on-site SAS/C Software Consultant or Representative. Your on-site consultant should be an experienced user who can assist you with questions and problems concerning SAS/C software. If your question or problem requires the expertise of an Institute consultant, your on-site consultant should place the call to the SAS Institute SAS/C Technical Support Department. Consultants are available weekdays between 9:00 a.m. and 5:00 p.m., eastern standard time. When you call, please identify your problem as a SAS/C software problem. Also, please refer to *A Guide for the SAS/C Compiler Consultant, Release 6.50*, which contains information that you will need when you talk to the technical support consultant.

The following table lists the various telephone numbers that you can call to obtain support for the SAS/C product.

| | |
|--|----------------|
| Main Phone | (919) 677-8000 |
| FAX | (919) 677-4444 |
| Technical Support Phone ¹ | (919) 677-8008 |
| Automated Call Processing (ACP) ² | (919) 677-8118 |
| Customer Service (Contracts Department) | (919) 677-8003 |

1 For SAS/C Software Consultants and Representatives only.

2 ACP is a menu oriented system designed to be used with touch-tone phones only.

SAS/C technical support can also be accessed on the World Wide Web at www.sas.com/techsup/intro.html .

SAS Publications

For a complete list of SAS publications, refer to the current *Publications Catalog*. You can order a free copy of the catalog by writing, calling, or faxing the Institute:

SAS Institute Inc.
Book Sales Department
SAS Campus Drive
Cary, NC 27513
Telephone: 1--800--727--3228
Fax: 919--677--4444
E-mail: sasbook.sas.com
WWW: www.sas.com/pubs

World Wide Web Pages

Please visit our web site at www.sas.com for additional information about SAS/C software, including an online version of our *C Notes* newsletter. If you would like to share information about your SAS/C application, and at the same time freely advertise your product, please call our Marketing Department as a “Real World Spotlight” candidate for *C Notes*.

You may also order SAS software documentation through the Internet by accessing the *Publications Catalog* on the World Wide Web at www.sas.com/pubs . From this same address, you can also download the *Publications Catalog* to install on your intranet.

Your Turn

If you have comments or suggestions about *Introducing SAS/C Software*, please send them to us on a photocopy of this page or send us electronic mail.

Send comments about this book to

SAS Institute Inc.
Publications Division
SAS Campus Drive
Cary, NC 27513
email: **suggest@unx.sas.com**

Send suggestions about the software to

SAS Institute Inc.
Technical Support Division
SAS Campus Drive
Cary, NC 27513
email: **suggest@unx.sas.com**