SSN/29.5

Starlink Project
Starlink System Note 29.5

A J Chipperfield

20 August 2001

Copyright © 2000 Council for the Central Laboratory of the Research Councils

PCS
The Parameter and Communication Subsystems

Abstract

Application programs often need to obtain parameter values from a variety of sources and to communicate with other programs. The Parameter and Communication Subsystems (PCS) are a set of closely-related subroutine libraries which provide these facilities for many Starlink applications and the associated user-interfaces.

The PCS libraries will not generally be called directly by application programs, but form a basic part of the Starlink Software Environment which is described in Starlink Guide SG/4. Additional notes on using it under Unix are given in Starlink User Note SUN/144.

Copyright © 2000 Council for the Central Laboratory of the Research Councils

Contents

1 The Parameter Subsystem

The Starlink parameter system interface library, PAR (see SUN/114), provides an interface between application programs and an underlying system for handling program parameters.

The current underlying parameter subsystem allows parameters to be obtained from a number of different sources such as the user (via prompts) or values generated by other applications. Facilities are also provided for dynamic generation of default values and value limit checking. Programs may also save parameter values for use in later invocations or by other programs.

The system is implemented by five libraries in the PCS package.

SUBPAR The top level of the underlying parameter system. Many of the basic PAR subroutines are almost straight-through calls to the corresponding SUBPAR subroutines.

PARSECON Parsing of the interface files associated with SUBPAR. The package also includes the interface file compiler, COMPIFL.

LEX A lexical analyser used by SUBPAR in analysing command lines *etc*.

STRING Fortran string manipulation subroutines. These subroutines are also used by software items outside the parameter system, but the library has no published interface.

MISC Miscellaneous routines which do not fit into other libraries. Two handle terminal I/O for programs running directly from a Unix shell and others provide a Fortran interface into the C library for platforms (notably Linux) which do not include them as part of the system.

2 The Communication Subsystems

These libraries provide the system which is currently used to construct programs capable of communicating with each other using the ADAM message protocol. Messages can control the actions of programs or convey information.

SG/4 describes the simple use of the system (which is usually all that is needed for data analysis programs) whilst SUN/134 describes more complicated use in instrumentation control systems.

MSP This library provides an inter-program communication system based upon a system of message 'queues'.

SOCK This library provides a Unix-socket-based message transport system for the MSP system.

AMS This library implements the ADAM message protocol on top of MSP. AMS is written in C but a Fortran interface, FAMS, is provided. SUN/241 is the programmers manul for AMS.

DTASK This provides an application program structure which allows the program to be run directly from the shell or to respond to a specified set of control messages from other programs using the ADAM message protocol. The application may consist of multiple 'actions' which can be controlled separately. DTASK provides the main routine of the program and applications are written as subroutines which are called by the DTASK layer after the communication and parameter systems have been initialised. This is described for system programmers in SSN/77 The package also includes shell scripts to link such applications with PCS and the other Starlink subroutine libraries required.

TASK This library provides an interface between the application code and the DTASK layer so that an application can find out some information about its own status. It also enables programs to control other co-operating programs using the ADAM message protocol and includes subroutines for encoding and decoding data values in messages. For more details, see SUN/134.

ATIMER This library provides a system of millisecond interval timers used by the message system and DTASK. Each timer has an associated handler which is invoked when the timer expires. The library is written in C but a Fortran interface, FATIMER, is provided.

3 HDSPAR

This library is a slight anomaly and should probably be a separate item. It provides a link between the parameter system and the Hierarchical Data System (HDS), enabling object names to be specified by program parameters. Unlike other PCS subroutines, the HDSPAR routines, DAT_ASSOC *etc.*, are expected to be called directly from application programs.

The HDSPAR library is described in SUN/224.

4 Obsolete Components

ADAM This library sends and receives messages using MESSYS. The message content is assembled/disassembled from/to its constituent parts.

MESSYS This level of the message system provides compatibility with earlier systems, thus removing the necessity for wholesale re-writing of higher-level libraries. It consists mainly of the corresponding calls to AMS via the AMS Fortran interface. Many of the important parameters of the message system are defined here.

In PCS V4.0 use of these two libraries has been completely replaced by calls direct to the Fortran interface of AMS. However, include files from both packages are still required so the components are still included in PCS but the subroutine libraries are not built or installed.

This situation needs re-organisation

5 Origins

The PCS libraries have been developed over many years at several different establishments. Initially developed as part of the ADAM environment for instrument control at the Royal

Observatories, notably ROE, the environment was adopted in 1986 by the Starlink project at the Rutherford Appleton Lab to support data analysis programs. Since then further developments have been made by ROE with substantial support and further developments from Starlink and additional support from AAO and JACH.

6 References

Note: Only the first author is listed here.

Lawden, M.D.	SG/4	ADAM – The Starlink Software Environment.
Chipperfield, A.J.	SUN/144	ADAM – Unix Version.
Currie, M.J.	SUN114	PAR – Interface to the ADAM Parameter System.
Kelly, B.D.	SUN/134	ADAM – Guide to Writing Instrumentation Tasks.
Kelly, B.D.	SUN/241	AMS – The Unix ADAM Message System.
Chipperfield, A.J.	SSN/77	ADAM – The Control Subsystem.
Warren-Smith, R.F.	SUN/92	HDS – Hierarchical Data System.