AMS Extensions
Version 2.0
Packaging & Availability

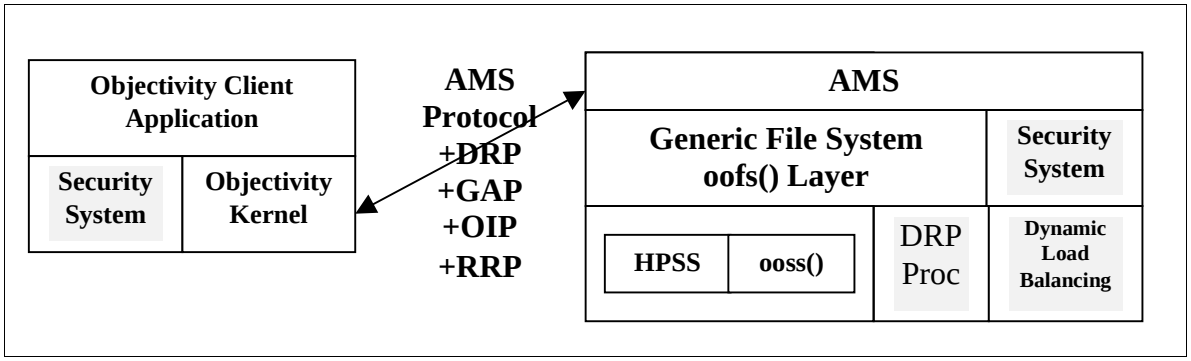
Andrew Hanushevsky
Stanford Linear Accelerator Center
7/24/98

1 Introduction

This document describes the packaging and availability of the Advanced Multithreaded Server (**AMS)** extensions. The extensions provide:

- a general file system interface, **oofs()**,
- Kerberos authentication using the Generic Authentication Protocol (**GAP**),
- processing of client-initiated transfer of opaque information through the Opaque Information Protocol (**OIP**),
- computation of variable request timeout using the Defer Request Protocol (**DRP**), and
- dynamic load balancing using the Request Redirection Protocol (**RRP**).

The relationships between these components are shown in the following figure.



1.1 Acronyms

@sys	Signifies AFS architecture specific name (e.g., sun4x_55)
AFS	Andrew File System
AMS	Advanced Multithreaded Server
AMSC	Advanced Multithreaded Server Collective
DRP	Defer Request Protocol
GAP	Generic Authentication Protocol
OCSK	Objectivity Client-Side Kernel
OOFS	Object Oriented File System
OOSS	Object Oriented Staging System
OIP	Opaque Information Protocol
RRP	Request Redirect Protocol
SLAC	Stanford Linear Accelerator Center

2 Availability

The following table shows the availability of each component (shaded items, either in prvious figure or the table below, are not currently available):

System Component	ID	Available
AMS Version 5[1]	AMS	now
Generic File System Layer	oofs()	now
High Performance Storage System Interface[2]	HPSS	now
Object Oriented Staging System	ooss()	now
Security System	GAP	2Q99
Defer Request Processing	DRP	1Q99
Dynamic Load Balancing Scheduler	RRP	3Q99

3 Packaging

The **AMS** along with the associated extensions are available through the Stanford Linear Accelerator Center (**SLAC**). The programs reside in globally accessible **AFS** rooted at:

/afs/slac.stanford.edu/package/hpss/objectivity

The **AFS** directory is protected and persons wishing access to the **AMS** and associated extensions must obtain a **SLAC** account and demonstrate that they have a current **AMS** license.

AMS extensions are provided in binary and source form. The AMS itself is provided only in binaryt.

[1] The AMS version 5 supporting the extensions is only available for Solaris 2.5 or above.
[2] The HPSS 3.2 interface is available only for AIX version 4.1 or above.

3.1 Distribution Tree Layout

The following diagram shows the distribution tree in detail starting at the root prefix.

Directory Structure	Contents
bin	Binary executables
rs_aix41	Executables for AIX 4.1
rs_aix42	Executables for AIX 4.2
sun4x_55	Executables for Solaris 2.5
include	Include files required for compilation
lib	Binary libraries to build executables
rs_aix41	Libraries for AIX 4.1
rs_aix42	Libraries for AIX 4.2
sun4x_55	Libraries for Solaris 2.5
pftp	Paralellel FTP source and binary
include	Include files to build pftp
src	Source files
hpssxeq	Remote HPSS command executor
pud	Perl Universal Daemon for hpssxeq
oofs	Object Oriented File System Interface
bin	Object files
rs_aix41	Objects files for AIX 4.1
rs_aix42	Objects files for AIX 4.2
sun4x_55	Objects files for Solaris 2.5
ooss	Object Oriented Staging System Interface
bin	Objects files
rs_aix41	Objects files for AIX 4.1
rs_aix42	Objects files for AIX 4.2
sun4x_55	Objects files for Solaris 2.5
mps	Migration / Purge / Staging Subsystem
ootest	Routine to test the oofs() interface
bin	Objects files and executables
rs_aix41	Objects files and executables for AIX 4.1
rs_aix42	Objects files and executables for AIX 4.2
sun4x_55	Objects files and executables for Solaris 2.5
util	Utility routines used by various commands
bin	Objects files and executables
rs_aix41	Objects files and executables for AIX 4.1
rs_aix42	Objects files and executables for AIX 4.2
sun4x_55	Objects files and executables for Solaris 2.5

3.1.1 Contents of bin/

Item	Description
Makefile	Makefile to build executables in sub-directories
rs_aix41	Final executables for AIX 4.1
rs_aix42	Final executables for AIX 4.2
sun4x_55	Final executables for Solaris 2.5

3.1.2 Contents of bin/@sys

The following components are currently available only for sun4x_55 (i.e., Solaris 2.5).

Item	Description
amshpss	AMS linked with the oofs / ooss / hpss_stage components
amstest	Command to test whether the AMS actually works
fs_stat	Command to provide consistent filesystem space information
hpssxeq	Interactive command to issue common HPSS operations
ooams	Standard Objectivity version of the AMS
ooams_hpss	Objectivity's AMS linked to a debugging oofs() interface

3.1.3 Contents of include/

Item	Description
ooConfig.h	Standard objectivity configuration include file
ooPlatform.h	Include file required by ooConfig.h
oofs.h	Standard oofs() interface definition
oofs_api.h	Required definition for oofs_api.c
oofs_types.h	Definitions for architecture specific objects (e.g., 64-bit ints)
ooss.h	The ooss() interface definition
ooss_api.h	Required definitions for ooss_api.c
ooss_config.h	Underlying filesystem configuration for ooss() to use

3.1.4 Contents of lib/@sys/

Library **ooams.a** is only available for sun4x_55 (i.e., Solaris 2.5).

Item	Description
liboofs.a	Compiled an archived oofs() interface
libooss.a	Compiled and archived ooss() interface
ooams.a	AMS that can be linked with the oofs() interface
ooss_hpss.o	Object file that provides the HPSS interface for ooss()

3.1.5 Contents of pftp/

This directory is a modified version of the **pftp** component that comes with **HPSS**. This component is distributed only to **HPSS** license holders and is only available for sun4x_55 (i.e., Solaris 2.5). Only Modified files are described here.

Item	Description
Makefile	Makefile to build executable in this directory
cmds.c	Modifications to provide consistent command error status and allow command line and file provided non-interactive scripts
ftp.c	Modifications to provide consistent error detection
getpass.c	Allow password to come from an alternate input source
main.c	Allow command line and file provided non-interactive scripts
pftp_client	Executable

3.1.6 Contents of src/hpssxeq/

The **hpssxeq** command is only available for rs_aix41 and rs_aix42 (i.e., AIX 4.1 and above).

Item	Description
Makefile	Build an executable in the "bin/@sys/" directory
hpssxeq.c	Source for the interactive HPSS command

3.1.7 Contents of src/hpssxeq/pud//

This directory contains the Perl Universal Daemon (**PUD**) used by the **rxhpss** command to allow the **hpssxeq** command to be issued on nodes that do not have the **HPSS** client API. The **rxhpss** command is used by the **ooss()** staging code.

Item	Description
pud	The remote execution daemon
pud_hpss.cf	Daemon configuration file
pud_hpss.pm	The remote execution interface implementation
pudc	The remote execution client
pudc.conf	Client configuration file
rxhpss	Executable that remotely runs rxhpss

3.1.8 Contents of src/oofs/

This directory contains the Object Oriented File System (**oofs**) interface implementation. The object files are placed in the bin/@sys/ subdirectory while the library, **oofs.a**, is placed in lib/@sys/ directory.

Item	Description
Makefile	Makefile to build object files and the oofs.a library
oofs.c	Defines the oofs() filesystem description vector
oofs.h	The oofs() interface definition
oofs_api.c	The oofs() interface implementation
oofs_api.h	Required definitions for oofs_api.c
oofs_error.c	Error number to string conversion routine
oofs_types.h	Definition of special objects (e.g., 64-bit integers)

3.1.9 Contents of src/oofs/bin/@sys/

Item	Description
oofs.o	Makefile to build executable in this directory
oofs_api.o	Modifications to provide consistent command error status and allow command line and file provided non-interactive scripts
oofs_error.o	Modifications to provide consistent error detection

3.1.10 Contents of src/ooss/

This directory contains the Object Oriented Staging System (**ooss**) interface implementation. The object files are placed in the bin/@sys/ subdirectory while the library, **ooss.a**, is placed in lib/@sys/ directory.

Item	Description
Makefile	Makefile to build object files and the ooss.a library
ooss.h	The ooss() interface definition
ooss_api.c	The ooss() interface implementation
ooss_api.h	Required definitions for ooss_api.c
ooss_config.h	Underlying filesystem configuration definitions
ooss_Error.c	Error routine
ooss_Genpath.c	Local to remote path translation routine
ooss_hpss.c	Staging interface to HPSS
ooss_Serialize.c	Directory and file access serialization routine
ooss_Stage.c	Routine to interface with file staging system

3.1.11 Contents of src/ooss/bin/@sys/

This directory contains the Object Oriented Staging System (**ooss**) object files.

Item	Description
ooss_api.o	Object file of ooss_api.c
ooss_Error.o	Object file of ooss_Error.c
ooss_Genpath.o	Object file of ooss_Genpath.c
ooss_hpss.o	Object file of ooss_hpss.c
ooss_Serialize.o	Object file of ooss_Serialize.c
ooss_Stage.o	Object file of ooss_Stage.c

3.1.12 Contents of src/ooss/mps/

This directory contains the Migration, Purge, and Staging (**MPS**) system.

Item	Description
ooss_MigrPurg	Migration and Purge subsystem
ooss_Stage	Staging subsystem

3.1.13 Contents of ootest/

Item	Description
Makefile	Makefile to create the interactive testing utility
otest.c	Source for routine to test the oofs() interface

3.1.14 Contents of ootest/bin/@sys/

Not all object files are provided for all architectures. The grayed items are available only for sun4x_55 (i.e., Solaris 2.5).

Item	Description
gsz_des.o	DES encryption routine
hpss_BuildLockfileName.o	Routine to build lock file names for HPSS access
otest	Executable that test the oofs() interface
otest.o	Object file of otest.c

3.1.15 Contents of util/

Item	Description
Makefile	Makefile to create various utilities
fs_stat.c	Command to provide filesystem free space in a standard format

3.1.16 Contents of util/bin/@sys/

Item	Description
fs_stat	Excutable that provides filesystem free space information
fs_stat.o	Object file of fs_stat.c