FD – MAS Operate and Maintain Manager

Content

History

Version	Date	Adjustments
A	2006-10-16	First version

1 Introduction

This document describes the Operate and Maintain Manager (OMM) component.

OMM provides the ability to operate on MAS and Monitoring functionality.

2 Function Structure

The OMM is used to communicate to MAS and monitor status of registered services.

For example:

- Lock/Unlock MAS component.
- Start/Stop/Restart MAS component
- Monitor provided services in MAS
- Monitor consumed services in MAS
- Supply the SNMP-agent with status information
- Monitor functionality to display session information on screen

The OMM component is providing interfaces for MAS to use for register services for OMM to monitor.

2.1 Overview

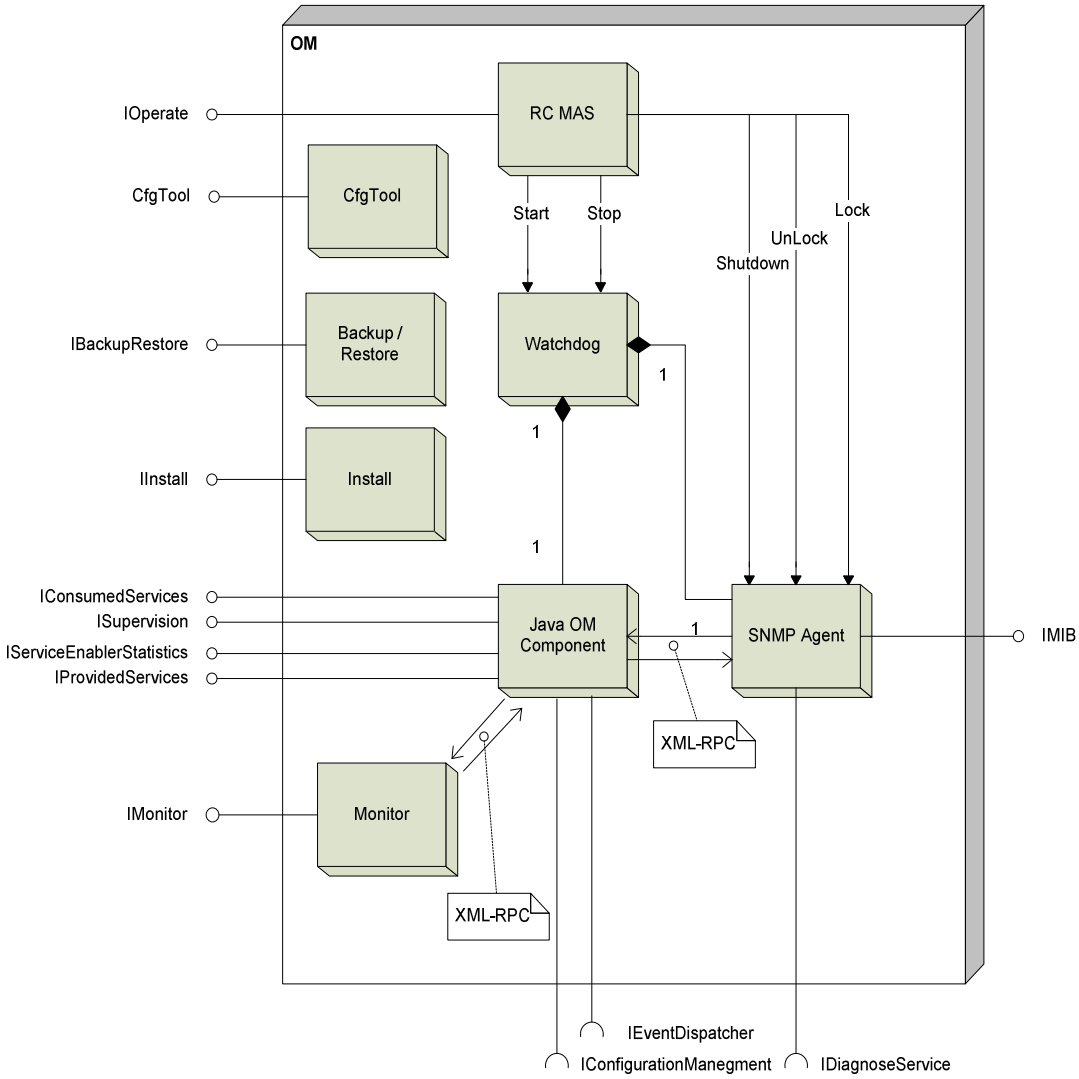

Figure 1

Figure 1 shows a coarse overview of the OMM component. There are several different objects in this component.

The RC MAS, Backup/Restore and Install objects is script based objects.
The Watchdog is a component provided by the Solaris 10.
Monitor, SNMP Agent and OM Component is java based objects.

2.2 Class structure

This is an overview of java OM Component.

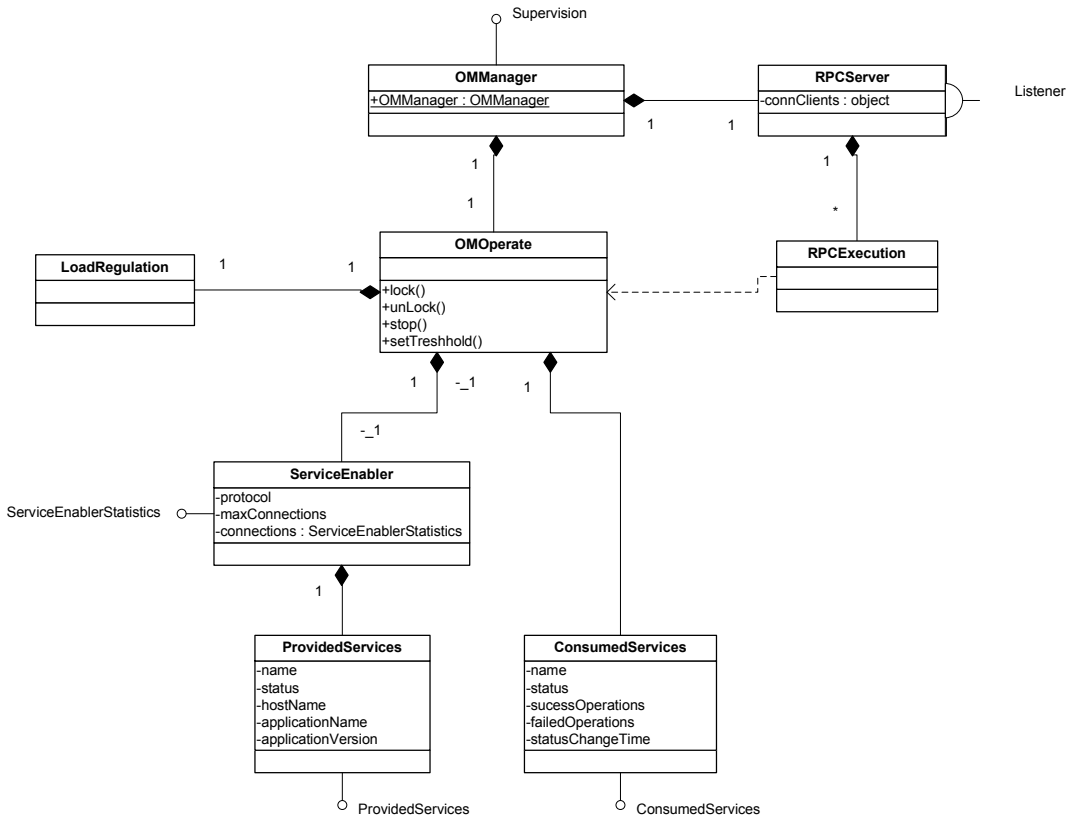

Figure 2

2.3 Script structure

2.3.1 Overview

This is a brief overview of what the installation does when installing on a Solaris 10 system.

When pkgadd is executed, it copies all files to its destination directories. Then pkgadd executes postinstall script to finish installation.

2.3.2 Solaris 10 install scheme

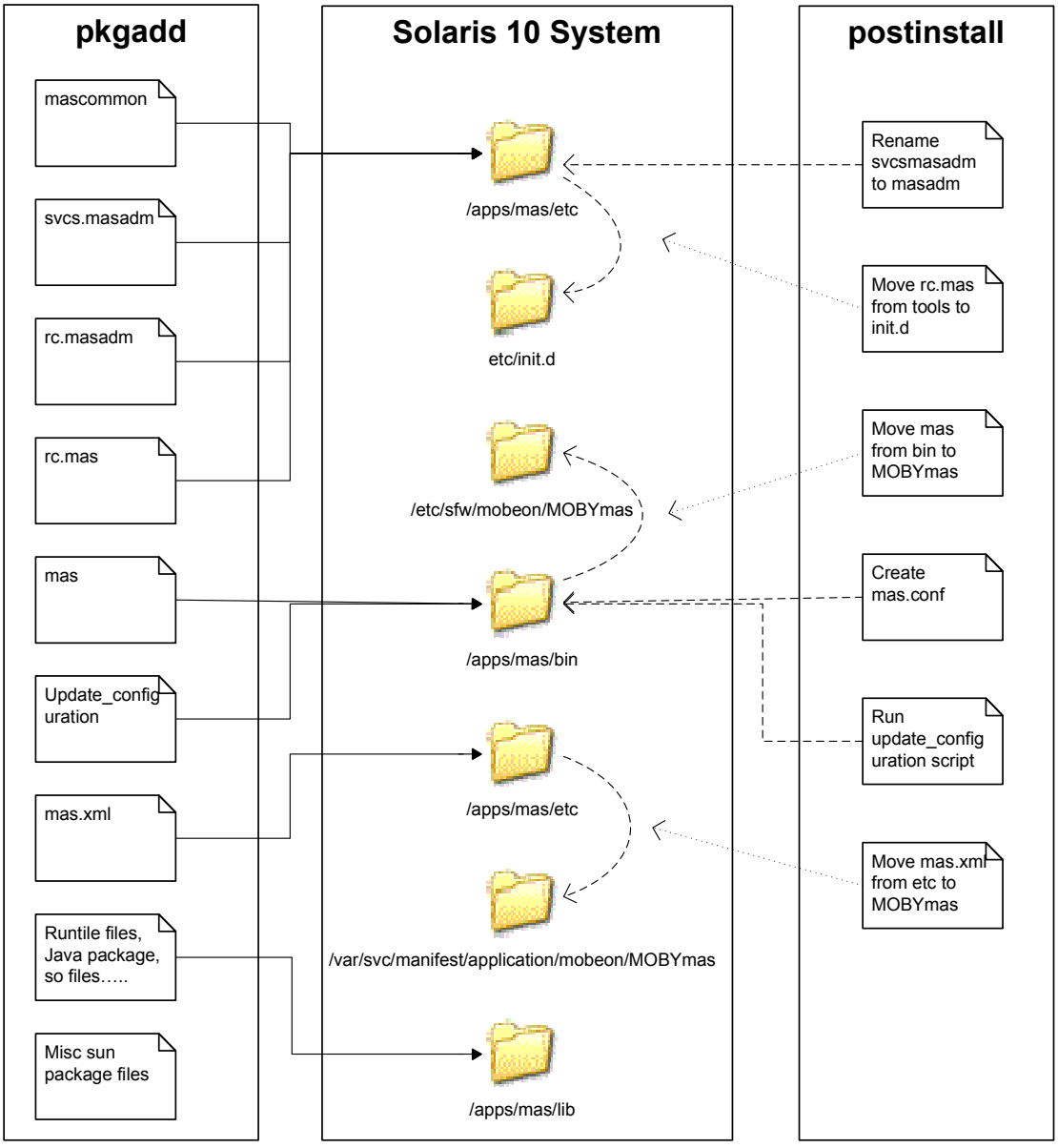

Figure 3

2.3.3 Solaris Package files

MAS package installation contains of several configuration files.

- pkginfo
  This file contains information about package name, install dir, default variable names, version of package.
  This file is needed by pkgadd to be able to install the package correct.
- request
  This file collects information from the installer and creates a responsefile.
  This file is used by pkgadd.
- mas
  This file is used when installing on a Solaris 10 system. This file contains start command used by svcs(watchdog). When svcadm enable is called, svsc looks in this script to determine what to do.
- mas.xml
  This file configure svcs to be able to find the mas script.
- mascommon
  This file contains functions that are independent on witch system MAS is installed on.
- rc.mas
  this file is installed in the /etc/init.d directory. This file calls the masadm script in the /apps/mas/tools directory.
- rc.masadm
  This file is renamed to masadm when installed on a Solaris 8 system.
  This file contains functions specific for Solaris 8 system.
- svcs.masadm
  This file is renamed to masadm when installed on a Solaris 10 system.
  This file contains functions specific for Solaris 10 system.
- Update_configuration
  This file update component xml configuration files with values specified during install or provided by the responsfile.
- Postinstall
  This script is executed before pkgadd is finished. This script determine target system version and install files.
- Preremove
  when pkgrm is executed, this script runs first, before removal of installed files. Move configuration files to a safe location.
- Postremove
  when pkgrm is executed, this script runs last, After removal of installed files. Clean up left files and directories.

3 Function Behavior

3.1 OMM Function Behavior

3.1.1 Register Service Enabler

To register a service enabler, the service enabler class that should be registered must implement the ServiceEnablerOperate interface.
Register the service enabler by calling createServiceEnablerEntry(ServiceEnablerOperate serviceEnabler);
This will return a ServiceEnablerStatistics interface to be able to operate on the registered service enabler in OMM.

If the service enabler is not registered, the service enabler is added and a new ServiceEnablerStatistic reference is returned.

If the service enabler is registered, the service enabler is not added and a reference to the existing ServiceEnabler

3.1.2 Register Provided Service

Register the provided service by calling createProvidedServiceEntry(String serviceName, String Host, Integer port, String serviceEnablerName);
This will return a ProvidedService to be able to set the status of the service.

If the service enabler is not registered, a new service enabler entry is created.

3.1.3 Register Consumed Service

Register the consumd service by calling createConsumedServiceEntry(String serviceName, String Host, Integer port);
This will return a ConsumedService to be able to set the status of the service and increment success and failed operations.

3.1.4 Register Connection Monitor

To be able to monitor connection statistic a connection monitor must be registered.

The class that contains the functionality to provide OMM with connection statistic must implement the ConnectionMonitor interface.
Then register the object by calling registerConnectionMonitor(ConnectionMonitor monitor)

When OMM receives a request to start monitor data, the startMonitor(ConnectionInfo statistic) in the registered object is called with the object to be filled with data.

When stopMonitor() is called, there is no need to fill the statistic object with data.

3.2 Operate Function Behavior

These operations are manual script operations called from command prompt.

3.2.1 Lock

Locks all provided services registered in OMM.

3.2.2 Unlock

Unlocks all provided services registered in OMM.

3.2.3 Shutdown

Locks all provided services registered in OMM.

Then the MAS are shutdown.

3.2.4 Start

Starting up MAS and read configuration.

3.2.5 Stop

Stop MAS without ending all calls before shutdown.

3.2.6 Restart

Stop MAS if running without ending all calls before shutdown.
Then start MAS again.

3.2.7 Status

Print the status of the MAS process.

3.2.8 Enableautostart

Enables restart of MAS process if process dies.

3.2.9 Disableautostart

Disables restart of MAS process if process dies.

3.2.10 Monitor

Start Monitor to be able to view connections and statistics.

3.2.11 Viewmib

Print Mib values on screen.

3.2.12 Installmcp

Install MCP package on this host.

3.2.13 Uninstallmcp

Uninstall MCP package installed on this host.

3.2.14 Viewmcp

Print MCP information on screen.

3.2.15 Installapp

Install application package on this host.

3.2.16 Uninstallapp

Uninstall application package installed on this host.

3.2.17 Viewapp

Prints application information on screen

3.2.18 Reoadconfig

Tell MAS to reload configuration parameters.

3.2.19 Register

Register MAS, MCP and application to MCR

3.2.20 Unregister

Unregister MAS, MCP and application to MCR

3.2.21 Version

Print MAS version on screen.

3.2.22 Testnumber

Prints number analyzer information on screen.

3.3 SNMP-Agent Function Behavior

The SNMP-Agent for MAS handles reading and writing values to the MIB and communicates with MAS. The SNMP-Agent also handles the diagnose service function.

The SNMP-Agent tries to communicate with all services registered in MAS and try to determine if the status of the service

The SNMP-Agent for MAS has no external command interface.

4 References

Intentionally left blank.

5 Terminology

OMM	Operate and Maintain Manager.
MCP	Media Content Packages.
MCR	Messaging Component Register.
XML-RPC	Remote Procedure Calling protocol.