



Hermes Webmin

Technical Reference

Version 1.0

Prepared by

**Center for E-Commerce Infrastructure Development (CECID)
The University of Hong Kong**

Maintained by Simon Lee (kylee@cecid.hku.hk)

Table of contents

| | |
|--|----------|
| 1. Status of this document | 3 |
| 2. Architecture | 3 |
| 3. Webmin System Operations Modules | 3 |
| 4. MSH Administration Operation Modules | 4 |
| 3.1 MSH Information | 4 |
| 3.2 MSH Maintenance | 5 |
| 3.3 MSH Database Information | 5 |
| 3.4 Message Tracking | 5 |
| 3.5 Message Archive | 6 |
| 3.6 Diagnosis Dump | 6 |

1 Status of this Document

The purpose of this document is to show the Hermes MSH API calls that have been used by the Hermes Webmin System.

2 Architecture

Hermes Webmin is a web-based admin front-end for Hermes MSH. It employs an agent-console architecture, which includes a Webmin servlet as the agent and some JSP pages as the monitoring console. The following diagram gives a pictorial view of the architecture.

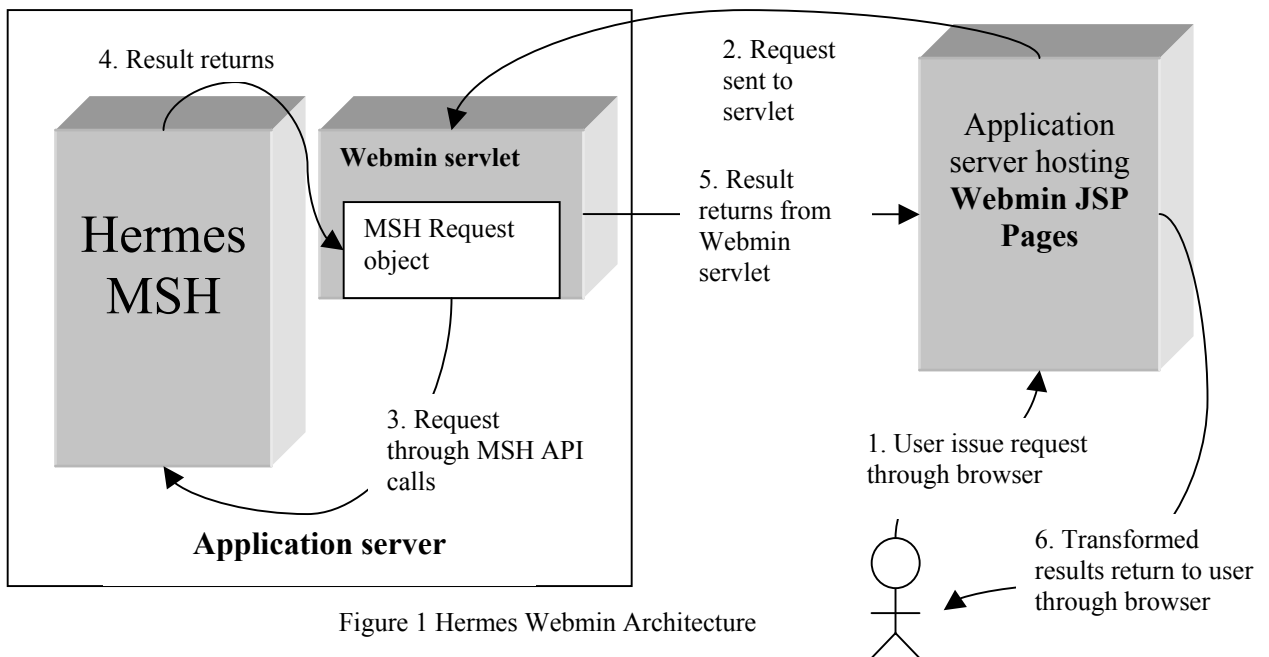


Figure 1 Hermes Webmin Architecture

The Webmin servlet acts as the interface between the MSH servlet and the administration JSP pages. All functions related to the MSH API call are implemented in the Webmin servlet. The JSP pages are for input and the results display. In the coming sessions, we will give the mapping between the JSP function modules and the APIs that they have called.

3 Webmin System Operations Module

This module allows you to perform Webmin system administrations, which includes adding user, removing user, editing user access right, change user login password and change the current monitoring MSH. These functions are all handled by the JSP

application, but not MSH. The following table gives the mapping between the function and its function call name.

| Function | API call |
|-----------------------|--|
| Add user | hk.hku.cecid.phoenix.hermes.webmin.app.UserManager.addUser(newUserID, newPasswd, newRight) |
| Edit user | hk.hku.cecid.phoenix.hermes.webmin.app.UserManager.editUser(editingUser, newRight) |
| Remove user | hk.hku.cecid.phoenix.hermes.webmin.app.UserManager.removeUser(removingUser) |
| Change login password | hk.hku.cecid.phoenix.hermes.webmin.app.UserManager.editUser(editingUser, newPassword, newRight) |
| Change Monitoring MSH | hk.hku.cecid.phoenix.hermes.webmin.app.WebminRequestHandler.setMshUrl(mshURLString, authName, authPwd) |

4 MSH Administration Operations Modules

This module allows you to perform MSH Administration operations, which includes MSH information checking, messages tracking and message archiving etc. Most of these functions are handled by MSH, while the others are handled by the JSP application. The following sessions describe the API call that we have used.

4.1 MSH Information

This module allows you to take a view on the MSH systems information. The following table gives you the mapping between the function and its API call name.

| Function | Handler | API call |
|---------------------------------|-----------------|---|
| Get Monitoring MSH URL | JSP application | hk.hku.cecid.phoenix.hermes.webmin.app.WebminRequestHandler.getMshUrl() |
| Get MSH Connection Availability | JSP application | hk.hku.cecid.phoenix.hermes.webmin.app.WebminRequestHandler.isMshConnectionAlive() |
| Get MSH Version | MSH | hk.hku.cecid.phoenix.message.handler.MessageServiceHandler.getMetaData().getRelease() |
| Get MSH Trusted Repositories | MSH | hk.hku.cecid.phoenix.message.handler.Request.getTrustedRepository() |
| Get Java Environment | MSH | hk.hku.cecid.phoenix.message.handler.Request.reportEnvironment() |

4.2 MSH Maintenance

This module allows you to maintain the running status of MSH. The following table gives you the mapping between the function and its API call name.

| Function | Handler | API call |
|-----------------------------|---------|--|
| Halt Msh | MSH | hk.hku.cecid.phoenix.message.handler.Request.haltMSH(Request.HALT_SUSPEND) |
| Terminate Msh | MSH | hk.hku.cecid.phoenix.message.handler.Request.haltMSH(Request.HALT_TERMINATE) |
| Resume MSH | MSH | hk.hku.cecid.phoenix.message.handler.Request.resumeMSH() |
| Check whether MSH is halted | MSH | hk.hku.cecid.phoenix.message.handler.Request.getIsHalted() |

4.3 MSH Database Information

This module allows you to take a view on the MSH database information. The following table gives you the mapping between the function and its API call name.

| Function | Handler | API call |
|--|---------|--|
| Check database connection | MSH | hk.hku.cecid.phoenix.message.handler.Request.checkDatabase() |
| Reset database connection pool | MSH | hk.hku.cecid.phoenix.message.handler.Request.resetConnectionPool() |
| Get database connection pool information | MSH | hk.hku.cecid.phoenix.message.handler.Request.getDBConnectionPoolInfo() |
| Check number of records in database | MSH | hk.hku.cecid.phoenix.message.handler.Request.getNumRecordsInDB() |

4.4 Message Tracking

This module allows you to track the status of messages processed by MSH. The following table gives you the mapping between the function and its API call name.

| Function | Handler | API call |
|-------------------------|---------|--|
| Get pending messages | MSH | <code>hk.hku.cecid.phoenix.message.handler.Request.getPendingMessages()</code> |
| Delete pending messages | MSH | <code>hk.hku.cecid.phoenix.message.handler.Request.deletePendingMessage(pendingMsgList)</code> |
| Get messages status | MSH | <code>hk.hku.cecid.phoenix.message.handler.Request.getMessageStatus(msgList)</code> |

4.5 Message Archive

This module allows you to archive the messages that have been processed by MSH. The following table gives you the mapping between the function and its API call name.

| Function | Handler | API call |
|--|---------|---|
| Archive message by time | MSH | <code>hk.hku.cecid.phoenix.message.handler.Request.archiveByDate(startTime, endTime)</code> |
| Archive message by application context | MSH | <code>hk.hku.cecid.phoenix.message.handler.Request.archiveByAppContext(appContextsArray)</code> |

4.6 Diagnosis Dump

This module allows you to perform a diagnosis operation on MSH. The following table gives you the mapping between the function and its API call name.

| Function | Handler | API call |
|--|---------|---|
| Diagnosis Dump (dump database) | MSH | <code>hk.hku.cecid.phoenix.hermes.webmin.servlet.DiagnosisDumpHandler.dump(request)</code> <code>hk.hku.cecid.phoenix.message.handler.DiagTool.dumpDatabase()</code> |
| Diagnosis Dump (dump MSH log) | MSH | <code>hk.hku.cecid.phoenix.hermes.webmin.servlet.DiagnosisDumpHandler.dump(request)</code> <code>hk.hku.cecid.phoenix.message.handler.DiagTool.dumpLog()</code> |
| Diagnosis Dump (dump MSH configuration file) | MSH | <code>hk.hku.cecid.phoenix.hermes.webmin.servlet.DiagnosisDumpHandler.dump(request)</code> <code>hk.hku.cecid.phoenix.message.handler.DiagTool.dumpFiles()</code> |