

# QA

## Product QA plan

*July 14, 2001. Document Revision 2*

---

Netclime Inc.

P.O.Box 251666, LA, CA-90025  
Tel (310) 571-3135 Fax (646) 514-0412

email: [imagen@netclime.com](mailto:imagen@netclime.com)

# Imagen QA Plan

## Interpreter (Scanner & Parser)

For each script statement, several scripts must be created (refer to Lang\_ref.doc), testing each construction. Each script can be either right or wrong. The script output must be saved in file, allowing automated testing. A program (or a batch file) **starting** all scripts must be provided. The series of test scripts must be run each time the interpreter is changed. A program (or a batch file) should **compare** the scripts output with the prerecorded (right) one, any difference should be reported.

Other test scripts containing:

- ◆ endless loops, testing the timeout & iteration restrictions
- ◆ endless recursion, testing the timeout & recursion restrictions
- ◆ very long expression, testing the stack used for the calculations
- ◆ lots of variables & functions, chains of include files
- ◆ combinations of sequences of symbols: if work as expected
- ◆ file/line numbers: if errors are displayed at appropriate places (check multiline comments too)
- ◆ correct error handling

## Graphics & Text

Scripts testing all graphic & test functions must be provided. The scripts should pass invalid values and the tester must check the results.

# Imagen Cache QA Plan

## Web Interface

- ◆◆ Imagen queries – cache hit, cache miss, too long
- ◆◆ Cache parameters – cleanup
- ◆◆ Error messages – invalid configuration, etc.

## Command-Line Interface

- ◆◆ Imagen queries
- ◆◆ Command-Line Cache arguments
- ◆◆ Error messages & user interface

## Apache Module Interface

- ◆◆ Test merging parent and child directory Apache configuration
- ◆◆ Perform the other tests for the web interface

## **IIS Extension Interface**

- ◆◆ Perform the tests for the web interface

## **Configuration**

It's better to perform the next tests as CGI and command-line application, Apache module and ISAPI extension.

- ◆◆ Test with no configuration file
- ◆◆ Test with incorrect syntax in the configuration file
- ◆◆ Check applying the values from the file
- ◆◆ Check applying default values
- ◆◆ Test with missing mandatory fields (i.e. fields with no default values)

## **Common Cases**

- ◆◆ Test the cleanup action
- ◆◆ Test the statistics action – updating when new cache entry added and showing current statistics
- ◆◆ Test the smart cleanup action
- ◆◆ Test the smart cleanup autostart behavior
- ◆◆ Test the trace log
- ◆◆ Test per cache entry synchronizations
- ◆◆ Test global synchronizations
- ◆ Max count of simultaneously running Imagen copies
- ◆ Performance tests (including time-eating comparison between the Cache as web server plugin, and disk-saved images)

# **Imagen Cache Plugin QA Plan**

## **Configuration**

- ◆ with no configuration file
- ◆ configuration parsing
- ◆ with no rings
- ◆ with just one ring with a few hosts on it:
- ◆ test priorities interpreting
- ◆ test when a part of the URLs are wrong
- ◆ test when a part of the URLs fail for a while (for example: turn off the web server on the remote host, and turn it on after a few tests; the same host should not be used for 50 requests)
- ◆ tests with simultaneous requests
- ◆ with more rings:
- ◆ test auto switching to the next ring, when all URLs on the current fail
- ◆ test using local Imagen when all rings are passed

- ◆ test resetting the rings by the **reset\_rings** dispatcher command
- ◆ log file processing
- ◆ statistics
- ◆ stress tests – many rings, many hosts per ring, many simultaneous requests

## Imagen Dispatcher QA Plan

### Configuration

- ◆ with no configuration file
- ◆ configuration parsing
- ◆ with no rings
- ◆ with just one ring with a few hosts on it:
- ◆ test priorities interpreting
- ◆ test when a part of the URLs are wrong

### Common cases

- ◆ test when a part of the URLs fail for a while (for example: turn off the web server on the remote host, and turn it on after a few tests; the same host should not be used for 50 requests)
- ◆ tests with simultaneous requests
- ◆ with more rings:
- ◆ test auto switching to the next ring, when all URLs on the current fail
- ◆ test using local Imagen when all rings are passed
- ◆ test resetting the rings by the **reset\_rings** dispatcher command
- ◆ log file processing
- ◆ statistics
- ◆ stress tests – many rings, many hosts per ring, many simultaneous requests