PPI
Inventory Integrity Checking
Project Specification
June 23, 2008

1. Concept

To be able to check all or part of a data holding to ensure that the files in the holding are intact and that preservation information exists for all files.

2. Vision

Using a command line tool use can generate a list of checksums for a portion of file system. This list can stored in a separate location and reused to verify that the contents of the file system have not been changed. The list can also be used to detect changes to the file system which include altered, missing or added files. The output from the tool can viewed by the user as plain text. The user can choose to view a complete report which includes an accounting of intact, altered, missing or added files; or the user can select to view only one accounting class.

3. Requirements

1. **Generate Checksum List.** The tool shall generate a list consisting of a checksum and the relative path to the file.
2. **Refresh Checksum List.** The tool shall have the ability to update an existing list by:

    a. Replacing a checksum for a file in the list with a newly calculated checksum (if different).
    b. Adding an entry (checksum and file path) for a file which is not currently in the list.
    c. Removing an entry for a file in a checksum list which no longer exists.

3. **Verify Holdings.** The tool shall be able to compare the checksum in a checksum list to the calculated checksum for the corresponding file and indicate whether or not the checksums match.
4. **Determine Missing Items.** The tool shall be able to determine which entries in the checksum list do not have a corresponding file.
5. **Determine New Items.** The tool shall be able to determine which files exist in the file system, but do not have an entry in checksum list.
6. **Generate Reports.** The tool shall be able to generate reports of:

    a. Those files which have not been altered (have checksums that match entry in list).
    b. Those files which have been altered (have checksums that do not match entry in list)
    c. Those files which are missing (have a list entry, but do not exist)
    d. Those files which are new (files that exist, but do not have a list entry)

and allow the user to select which reports to display.

7. **Recursion.** The tool shall be able to perform tasks recursively on the file system.
8. **Exclude File**. The tool shall be able to exclude a given file name from inclusion in checksum operations.

4. Supplementary Requirements

1. Written in Java (if new development)
2. Use MD5 checksums.
3. Use a checksum list file format compatible with md5sum. The list is stored in a plain text file with one entry per line. Each entry will consist of an MD5 checksum, two spaces, and the path to the file.

5. Platform and Network Environment

1. A command line shell for command execution and viewing of results.
2. Java Virtual Machine for execution of Java applications.
3. RedHat Linux platform release 9 through Enterprise 5
4. Windows XP and Vista platform.

6. Existing Tool Assessment

Assessed tools:

Tool	MD5	Recursive	Generate	Verify	Missing	New	Update	Exclude
md5sum	Yes	No	Yes	Yes	Yes[1]	No	No	No
md5deep	Yes	Yes	Yes	Yes	Yes[1]	No	No	No

(1) Does not distinguish between missing files and invalid checksum

None of the assessed tools fulfill all the requirements. Most notably is the lack of determining whether files exist which do not have a corresponding entry in a checksum list. This is critical for detecting errant files or new additions to the holdings.