Silverthread, Inc.
200 Portland St, Suite 500
Boston MA 02109

CodeMRI Usage Guide

Table of Contents

Vaults

Vaults are a collection of all projects at a client site. They are created by CMRI in the same system path as CMRI. They contain a full duplicate copy of scanned code, generated reports, and any other files produced by CMRI. The first step to starting CMRI in any directory is creating a vault.

1. In command line, navigate to a directory to place the vault.
2. Run the command “cmri” to start CMRI.
3. It should trigger the following output:

```
/PATHNAME does not seem to exist.
Would you like to create a new vault here? (y/n)
```

    a. Enter “n” to cancel the CMRI startup
    b. Enter “y” to continue the CMRI startup

4. After typing “y” to start CMRI, answer the questions
5. Contact information for vault creation does not need to match the license email.

```
/PATHNAME does not seem to exist.
Would you like to create a new vault here? (y/n)
y
What organization owns this vault? (Silverthread Client)
The Daystrom Institute
What is the group that owns this vault? (Engineering)
```

```
Software Management
Who is the primary contact for this vault? (John Doe)
Michael Okuda
Where is the primary contact located? (Nowhere, OK)
San Fransisco, CA
What is the primary contact's e-mail address? (unknown@silverthreadinc.com)
mokuda@ufop.gov
What is the primary contact's phone number? (000-000-0000)
477-477-4747
4 CPUs, 8903 MB of memory available. Defaulting to 2 workers.
Welcome to CMRI Development Platform v1.19.3.300.   Type help or ? to list
commands.


==================================
0 projects and 0 systems selected.
```

6. The vault is now created and CMRI is configured

Reports

Reports are the files produced by CMRI in Microsoft Excel format. They contain the diagnostic information regarding the health of the codebase including financial and timeframe metrics related to the impact of the codebase's existing architectural design.

1. Create & select a project
    a. Type "project add" and follow the prompts

```
==================================
0 projects and 0 systems selected.
project add

What would you like to name this project? (MyProject)
LCARS

==================================
0 projects and 0 systems selected.
```

    b. Select the project that was just created by typing "project select <project name>"

```
==================================
0 projects and 0 systems selected.
project select LCARS

vault:LCARS added to selection.

==================================
1 projects and 0 systems selected.
```

2. Create & select a system
    a. Type “system add” and follow the prompts

```
==================================
1 projects and 0 systems selected.
system add
Please provide the location of the source code or metadata for this system.
~/path/to/LCARS_SOURCE.zip
What would you like to name this system? (MySystem)
LCARS_database
What is the version of this system? (1.0)
47.0
==================================
1 projects and 0 systems selected.
```

    b. Select the system by typing “system select <system name>-<system version>”
        i. Wildcards are allowed

```
==================================
1 projects and 0 systems selected.
system select LCARS*

vault:LCARS:LCARS-47.0 added to selection.

==================================
1 projects and 1 systems selected.
```

3. Log into your Silverthread account using the command “login”
    a. Enter the information associated with the CodeMRI.com account

b. If the internet is inaccessible, obtain and enter an offline license

4. Scan the code by running “produce_reports”

```
run produce_reports
```

5. Several jobs will execute in order, processing the code and producing a set of Microsoft Excel Spreadsheets. To view the final reports, navigate to your vault root. The reports can be found in the vault root under the *vault/reports/<project name>/<system name>/* directory.

Offline License

It is still possible to use CMRI without an internet connection by using an offline license. An offline license is a license key that permits scans without logging into CodeMRI.com within CMRI. It can be obtained by sending a machine ID acquired within CMRI to Silverthread.

1. Run the command “license id.” This will create a unique machine ID

```
==================================
0 projects and 0 systems selected.
license id


Machine Identity
--------------------------------------------------------------------------------
Node: EC2AMAZ-GPONIPK#64bit#WindowsPE#AMD64#Intel64 Family 6 Model 63
Stepping 2, GenuineIntel#Windows#19916397486644
Node id: 57839eda3f7debc772ef1cabcb87b7b2
Vault id: dd8e9e82-a5dd-4372-9673-320257472d76


The machine identity has been written to:


C:\Users\Administrator\vault\licenses\machine_id.json


Please send this file to Silverthread for offline licensing.


==================================
0 projects and 0 systems selected.
```

2. Copy down the Node, Node ID, and Vault ID precisely in whatever manner is most convenient

3. Send the 3 values to a Silverthread representative
    a. The Silverthread representative will respond with an offline license file
4. Migrate the license file to the offline machine
5. Once back in CMRI, register the offline license with the command “license add -f path/to/<license name>.lic”

```
==================================
0 projects and 0 systems selected.
license add –f path/to/offlineLicense.lic

License added.

==================================
0 projects and 0 systems selected.
```

6. CMRI can now performs scans without an internet connection

Component Reports

A component is a manually defined file or directory of a codebase. Components allow the user to manually define relationships between different parts of the codebase. This allows CMRI to compare and analyze the similarities and differences of the actual codebase architecture with the theoretical architecture. The Component Reports visually depict the theoretical and actual component relationships using DSMs. This allows for broad analysis regarding where the least healthy regions of the codebase are.

1. Create & select the desired projects and systems
2. Run the command “system config components” to enable component creation

```
==================================
1 projects and 1 systems selected.
system config components

Listing keys for the current selection:

vault:CODEBASE:CODEBASE_database-1.0:
components:
```

```
==================================
1 projects and 1 systems selected.
```

3. To create components, use the command “system components add <component name> -e path/to/component/*
    a. When specifying the pathname to a component, CMRI is looking for the pathname beginning at the top level of the source code directory

```
==================================
1 projects and 1 systems selected.
system components add proxies –e src/Proxies/*

Added component `proxies` to `vault:CODEBASE:CODEBASE_database-1.0`.
Added expression(s) `src/Proxies/*` to component `proxies` of
`vault:CODEBASE:CODEBASE_database-1.0`.

==================================
1 projects and 1 systems selected.cmr
```

    b. This process can be repeated to add multiple files or directories to a single component

4. Once all the desired components are created, run “produce_silverthread_database” to fetch the component data

```
run produce_silverthread_database
```

5. Once complete, verify the components were generated properly with the command “system components list”

```
==================================
1 projects and 1 systems selected.
system components list

vault:CODEBASE:CODEBASE_database-1.0
Name     Expressions           File Count
-------  --------------------  ------------
tools    src/Tools/*           XX
proxies  src/Proxies/*         10
server   src/Server/*          123
```

```
design   src/Design/*          86


==================================
1 projects and 1 systems selected.
```

   a. If the file count for any directory is listed as “XX,” verify the pathname is spelt correctly, does not include special characters and that there are source code files in that directory

6. To declare expected component relationships, use the command “system components add-relationships <component A> <component B>,<component C>,…”
   a. This will assign a relationship from component A to B, C, etc.

```
==================================
1 projects and 1 systems selected.
system components add-relationships design server

Added relationship `design`->`server` to `vault:CODEBASE:CODEBASE_database-1.0`


==================================
1 projects and 1 systems selected.
```

   b. Use the command “system components list-relationships” to see all declared component relationships

7. Once all components and relationships are declared, produce reports
8. CMRI will create a new *Component Reports* folder in the *vault/reports/<project name>/<system name>/* directory
   a. This will include numerous diagrams detailing the hierarchical structure of the actual and theoretical component relationships

Portfolio Light

The Portfolio Light is an Excel file that contains brief summary of the top-level health of a codebase. It is a concise, basic description of the codebase diagnostics that is intended for non-technical and or executive members of the codebase team.

1. Create & select any projects and systems
2. Run the command “produce_portfolio_light”
   a. It is not required to produce reports beforehand

```
run produce_portfolio_light
```

3. The Portfolio Light will be placed in the *vault/reports/Portfolio_Light/* directory

Technical Health Improvement Plan

The Technical Health Improvement Plan (THIP) is an Excel document that details a series of steps determined by CMRI that provides the most efficient path to eliminate a core from the scanned codebase. This includes metrics including the impact of each broken relationship, in which files to locate the erroneous relationships, and more. It is intended to guide any struggling codebase back to a more healthy and manageable architecture. It aids Silverthread's goal to not only assess codebases, but to offer as much assistance as possible in the refactoring process.

1. Create & select a project and system
2. Produce reports for the system
3. After creating reports, run the command "produce_thip"

```
run produce_thip
```

    a. By default, the THIP will only consider critical cores (150+ files) and will pass over any smaller cores
    b. To customize the minimum core size for the THIP, run "produce_thip --size <num>"

```
run produce_thip ––size <num>
```

4. The THIP will be placed in the *vault/reports/<project name>/<system name>/* directory

Glossary

- **Benchmark**: the top 10% of all comparable codebases, defined as those which are between half and twice the size of the scanned codebase.
- **CMRI**: the CodeMRI executable
- **CodeMRI Scanner**: an application available for Windows. It can generate the metadata for a given codebase but does not provide any analytics.
- **CodeMRI Platform**: the full CMRI toolset.
- **Component**: a defined file or directory of a codebase. Components allow the user to manually define relationships between different parts of the codebase. This allows CMRI to compare and analyze the similarities and differences of the actual codebase architecture with the theoretical architecture.
- **Core**: a cyclical group of files. Theoretically, a core is any number of files $n \geq 2$ in which the relationship of the system could be mapped as a self-contained loop. Emerging cores are defined as cores containing 30-150 files and have been proven to cause some financial and developmental impact. Severe cores are defined as cores with 150+ files and have been proven to impose significant negative impacts on the codebase.
- **Design Structure Matrix (DSM)**: a visual representation of the hierarchical relationships of a codebase. Every file of the codebase is listed in order on both the x and y axes. The dots represent a direct relationship from the file on the y axis to the file on the x axis. A boxed group of dots indicates where a core is hierarchically within the codebase. For further reading, go here.
- **McCabe Complexity**: a measure of the efficiency of code itself; concise code will have low McCabe complexity whereas code with extraneous lines, functions, etc. will have high complexity. Files with a McCabe complexity of 20-50 are defined as problematic and have been proven to cause some financial and developmental impact. Files with values of 50+ are said to have high complexity and have been proven to impose significant negative impacts on the codebase.
- **Metadata**: the file produced by scanning a codebase with the CodeMRI Scanner. This file can then be scanned through the CodeMRI Platform to produce diagnostic analysis of the codebase.
- **Offline License**: a license key that permits scans without logging into CodeMRI.com within CMRI. It can be obtained by sending a machine ID acquired within CMRI to Silverthread.
- **Portfolio Light**: an Excel file that contains brief summary of the top-level health of a codebase.
- **Project**: a codebase or group of codebases that work together. Every vault must have at least one project.
- **Reports**: the files produced by CMRI in Microsoft Excel format. They contain the diagnostic information regarding the health of the codebase including financial and timeframe metrics related to the impact of the codebase's existing architectural design.
- **System/Codebase**: a snapshot of a project at a single point in time. Every project must have at least one system.

- **Technical Health Improvement Plan (THIP)**: an Excel document that details a series of steps determined by CMRI that provides the most efficient path to eliminate a core from the scanned codebase. This includes metrics including the impact of each broken relationship, which files to locate the erroneous relationships, and more.
- **Understand**: a program provided by SciTools that generates static analysis of a codebases required for CMRI to determine the codebase diagnostics.
- **Vault**: a collection of all projects at a client site. They are created by CMRI in the same system path as CMRI. They contain a full duplicate copy of scanned code, generated reports, and any other files produced by CMRI. The first step to starting CMRI in any directory is creating a vault.

Customer Support

Phone: 800-647-9366 (business hours Pacific Time)
Email: support@silverthreadinc.com