PNNL-SA-92579

STOMP Software Test Plan
Rev. 2.0

V. Freedman
M.D. White

December 2012

Prepared for
the U.S. Department of Energy
under Contract DE-AC06-76RLO 1830

Pacific Northwest National Laboratory
Richland, Washington 99352

This document electronically maintained on the STOMP SharePoint
(https://spcollab.pnnl.gov/sites/stompshare/Shared%20Documents/Forms/
AllItems.aspx)

Revision Log

Revision	Date	Authors/Reviewer	Summary of Changes
1.0	January 24, 2007	V. Freedman	First draft of Software Test Plan
2.0	December 11, 2012	V. Freedman	Updated to reflect current practices

Approvals

Approved: _____12/12/2012__
Mark D. White, STOMP Configuration Control Manager, Date

Approved: _____ 12/11/2012
Kary Cook, Software Quality Engineer, Date

Contents

1.0 Introduction 4
1.1 Purpose 4
1.2 Definitions 4
1.3 References 4
2.0 Limitations 4
2.1 Software Test Plan Maintenance 5
2.2 Software Test Environment 6
3.0 Test Identification 6
3.1.1 STOMP 6
3.1.2 STOMP Utilities 11
3.2 Acceptance Criteria 12
3.3 Test Schedule 12
4.0 Test Results Archival and Documentation 12

List of Tables

Table 2.1 STOMP Operational Modes 5
Table 3.1 STOMP Modes and Post Processors 8
Table 3.2 STOMP-W Tests/Directory Structure 11

1.0 Introduction

This document describes the software test plan (STP) for the Subsurface Transport Over Multiple Phase (STOMP) software used to support work under a number of different environmental assessments. This plan also includes pre- and post-processing utilities used in conjunction with STOMP.

1.1 Purpose

This document provides information on the requirements and procedures necessary for testing STOMP software in support of changes made in accordance with the STOMP Software Configuration Management Plan (SCMP).

This STP identifies specific tests to be run for two separate purposes. First, the tests will help ensure that the STOMP modules are correctly performing their intended function. Second, this plan identifies a standard suite of tests to be performed when software changes are made. Initially all tests in this test plan will be executed. Subsequently, only the specific modules being modified under an approved Software Change Request (SCR) will require testing to ensure the changes are implemented as desired. If changes made could affect other modules, then regression testing will be performed.

1.2 Definitions

Glossary

Term	Definition
DOE	Department of Energy
CVS	Concurrent Versions System (configuration management software)
NAPL	Non-Aqueous Phase Liquid
PNNL	Pacific Northwest National Laboratory
RL	Revision Log
SCM	Software Configuration Manager
SCMP	Software Configuration Management Plan
SCR	Software Change Request
STOMP	Subsurface Transport Over Multiple Phases
STP	Software Test Plan

1.3 References

- STOMP SharePoint
- STOMP Software Requirements Document (SRD)

2.0 Limitations

The STOMP simulator has been written with a variable source code that allows the user to choose the solved governing equations (e.g., water mass, air mass, dissolved-oil mass, oil mass, salt mass, thermal energy, CO_2 gas). Depending on the chosen operational mode, the governing

transport equations will be written over one to four phases (e.g., aqueous phase, gas phase, (nonaqueous phase liquid) NAPL phase, ice phase, solid phase).

In addition to the variable source code configuration, separate branches of STOMP source code are maintained. Production versions of STOMP are isolated from the research branch of the code. These versions are termed NQA-1 and CO2, where NQA-1 includes the modes –W, -WAE, -WO and –WOA. The CO2 branch includes the –CO2 and –CO2e modes. STOMP modes currently in production are further defined in Table 2.1.

Due to the complexity of the variable source code configuration, STOMP testing has been divided into three different stages, and is based on a thorough testing of the primary operational mode of water.

- In the first stage of testing, basic functionality of the operational mode is tested, using a limited number of options. In addition to testing for the operational mode's intended functions, unintended uses are also tested.
- In the second stage, options for the non-water modes are tested for both intended and unintended uses. For example, the second stage of STOMP-WO involves only testing the options describing oil transport.
- In the third stage of testing, water options for the operational mode are tested. Although these options have been tested in the STOMP-W mode, different subroutines are accessed in different operational modes. Hence, testing is required for governing equations that have already been tested in other operational modes.

Table 2.1 STOMP Operational Modes

Operational Mode Name	Operational Mode Identifier	Operational Mode
STOMP-W[a,b]	Water	1
STOMP-WAE[a,c]	Water-Air-Energy	3
STOMP-WO	Water-Oil	4
STOMP-WOA	Water-Oil-Air	5
STOMP-CO2[b,d]	Water-CO_2	32
STOMP-CO2e[a,b,d]	Water-CO_2-Energy	33

[a] Operational mode also exists for eSTOMP
[b] Can also be executed with ECKEChem Reactive Transport Module (-R)
[c] Can also be executed with Barrier (Plant) Module (-B)
[d] Can also be executed with Geomechanics Module (-EPMech)

2.1 Software Test Plan Maintenance

This plan will remain in effect as long as PNNL is supporting assessments with the STOMP software, or further use of the software is contemplated. The Software Configuration Manager (SCM) will have responsibility to monitor compliance, and ensure that changes and updates are reflected in the STP, as required. Reviews of this STP will occur periodically throughout the

project's life. Revisions to this STP will be managed manually, per the SCMP. This STP will be stored on the STOMP SharePoint site.

End use testing will be conducted before releasing source code to verify that any changes in the software, hardware and/or operating system have not impacted results. Test problems, are accessible on the STOMP SharePoint under the appropriate subdirectory as categorized by operational mode. This activity is documented in the appropriate test logs.

2.2 Software Test Environment

The software and scripts listed in Table 3.1 are executed in a UNIX Linux-based system. The development environment is also a Linux-based system, but is based on the Mac OS. Software is compiled using the Portland Group compiler for both development and unit testing. The Intel compiler is used for regression testing. Post-processors for STOMP are written in Perl.

Testing is performed by STOMP users, as assigned by the SCM. The SCM may also perform testing. STOMP developers may be used for testing, as long as the tests they perform are not code that they have developed. Although developers are responsible for testing, independent testing will be performed before the new source is accepted for production. A log file of results will be generated from each new test and its location recorded on the Change Implementation section of the SCR. If only component testing is required, then the comparison to previous results will be recorded on the SCR. However, for code that was checked into the repository before FY07 (before the current management plan was put into place), there are no SCRs that accompany source code updates and identify the locations of the test logs. Test logs will be generated and archived on the STOMP SharePoint site.

3.0 Test Identification

The software modes and scripts that will be tested as part of this test plan are identified by name in Table 3.1. A brief description of each software module and script is provided, as well as the tests required for configuration control. In the sections that follow, the tests planned for the software and scripts are described, and follow the requirements outlined in the STOMP SRD requirements document.

In general, most of the testing outlined in this document will be component testing, and will be structured to evaluate the correctness of the code. System-level (regression) testing will only be carried out on the source code for new source releases. The results of that testing will be recorded in a Revision Log. Complete automation of the testing process has been established. Analytical solutions will be used to verify test results when possible. In the absence of analytical solutions, experimental data may also be used to verify results.

3.1.1 STOMP

STOMP testing is carried out on an operational mode basis (as identified in the STOMP SRD). For each operational mode, a base case is established that tests its basic functionality. This base case is stored under the *base* subdirectory of the operational mode that is being tested. In the water mode, for example, a base case is stored under *stomp-w/base*. A second base case is stored under *stomp-w-t/base* that tests the functionality of solute transport. Because testing is structured

on operational modes that have already been tested, only one base case may exist for other operational modes.

Once a base case test has been established, different options are tested. These options are grouped by type, as organized by the input structure. For example, options in the water mode (stomp-w) include several different tests of boundary conditions, which are tested under the *bc* subdirectory. The different options that are tested exist in numbered subdirectories, and the correspondence of the numbered directory to the option being tested is recorded in an external database. Boundary condition options also exist for solute transport (stomp-w-t), and exist in a separate subdirectory under the stomp-w-t directory. Table 3.2 shows how testing is structured, and mimics the directory structure used to carry out both the component and system level tests.

For modes that consider other phases, testing is focused on the functionality of each phase in that mode. For example, in the water-oil mode (STOMP-WO), testing of the water phase is carried out in the *stomp-wo-w* directory, and testing of the oil phase is carried out in the *stomp-wo-o* directory.