Grant Agreement No.: 604656

Project acronym: NanoSim

Project title: A Multiscale Simulation-Based Design Platform for Cost-Effective CO_2 Capture Processes using Nano-Structured Materials (NanoSim)

Funding scheme: Collaborative Project

Thematic Priority: NMP

THEME: [NMP.2013.1.4-1] Development of an integrated multi-scale modelling environment for nanomaterials and systems by design

Starting date of project: 1st of January , 2014

Duration: 48 months

Document ID: WP1.TESTPLAN.MASTER

WP N°	Del. N°	Title	Contributors	Version	Lead beneficiary	Nature	Dissemin. level	Delivery date from Annex I	Actual delivery date dd/mm/yyyy
1	D1.3	Test Plan (based on IEEE 829)	Author: Thomas F. Hagelien Checked by: Stefan Radl (TUG) and Christoph Kloss (DCS)	1	SINTEF	Other	PU	31/06/2014	15/08/2014

1 Introduction

This document is the Master Test Plan (MTP) for the NanoSim/Porto project, and provides a high-level test planning and management overview. The MTP follows the IEEE 829-2008 format.

1.1 Document identification

Document Identification	PORTO-MTP (Porto Master Test Plan)
Author(s)	Thomas F. Hagelien
Reviewers	DCS, TUG
Manager	Dr. Shariar Amini (SINTEF)
Version of the Product	0.1
Version of this Plan	1

1.2 Scope

One of the challenges in the NanoSim project is integrating the different simulation tools involved in building a multi-scale simulation platform. As a common framework for offline coupling of the simulators, Porto is begin developed as part of the NanoSim project. In order to achieve the goals of developing an efficient and cost effective multi-scale platform, testing is an essential ingredient. As testing is an embedded part of the development work, planning and following up on the key testing activities is essential. In the different work-packages where software is being developed, it is assumed that adequate testing activities are being scheduled and executed. During the integration of the different models, and the testing of different offline coupling scenarios, however, the testing activities need to be planned and performed on a higher level. The goals for the system testing efforts are therefore to plan for quality for the overall system.

The requirements engineering process in NanoSim is largely ***Use Case***-driven. In the definition of the fully dressed use cases, the preconditions, success guarantee, main success scenario, extensions and other relevant factors are defined, and largely define the acceptance criteria for the required functionality.

1.3 References

Acronym	Name
FRD	Functional Requirements Document (D1.1, WP1)
WD	Detailed Workflow Document (D1.2, WP1)
PRP	Project Proposal (NanoSim EU Proposal)
PRP-1.3	Project Plan (NanoSim EU Proposal, Chapter 1.3)
EUC	External Use Cases (GitHub: https://github.com/NanoSim/Porto/blob/master/doc/specification/umlExternalCode/)

1.4 System overview and key features

See "*PRP 1.3.4 Table 1.3d: Work Package Description – Common Environment Software Platform*"

1.5 Test Overview

The consortium partners responsible for the planning, execution and reporting of the testing activities in WP1 are SINTEF with support from DCS and TUG. The integration and system tests will be performed incrementally and reported regularly as part of the half-year progress reports, limited to the involved modules and scenarios that are mature enough to be tested.

1.6 Organization and responsibilities

SINTEF is responsible for the administration and definition of the testing activities. DCS and TUG are supporting the execution and technical assistance of the testing activities. The NanoSim consortium is collectively responsible for defining the acceptance criteria.

1.7 Tools, techniques, methods and metric

Testing will be performed on various hardware and software configurations (Windows/Linux on desktops and clusters). The main testing methods for acceptance testing will be black-box testing, running of preconfigured workflow scripts, along with manual testing. Detailed white-box tests (Unit tests and integration tests); manual, semi-manual and fully automated unit tests will be conducted. Metrics for success will be based on pass/fail in acceptance testing, along with deviation reporting for regressions tests.

2 Details of the Master Test Plan

This section contains defines the high-level test processes. The tasks are defined in the FRD and EUC.

Task	Test Design
Method	Interface PaScal with OpenFOAM data format thermophysical data
Subsystem	PaScal
Inputs	Thermophysical data in OpenFOAM format available
Outputs	PaScal reports thermophysical data for various temperatures
Responsible	TUG

Task	Test Design
Method	Storing PaScal simulation data to vtk or hdf5 format
Subsystem	PaScal
Inputs	PaScal simulation result
Outputs	VTK or hdf5 format
Responsible	TUG

Task	Test Design
Method	Chemkin-II data format interfacing
Subsystem	PaScal
Inputs	Chemical reaction data in CHEMKIN-II format
Outputs	PaScal report on chemical reaction data
Responsible	TUG

Task	Test Design

Method	Drying of a Wet, Porous Particle
Subsystem	PaScal
Inputs	Relevant physical and chemical data. Specification of drying experiment.
Outputs	Simulation results
Responsible	TUG

Task	**Test Design**
Method	Import of ANSYS Fluent Mesh
Subsystem	CFDEM
Inputs	Fluent .msh or .cas file
Outputs	mesh OpenFOAM data format
Responsible	DCS

Task	**Test Design**
Method	C3PO library simulation test in ANSYS Fluent and NEPTUNE CFD
Subsystem	PaScal, C3PO, ANSYS Fluent and NEPTUNE CFD
Inputs	Physical and chemical data. BCs, ANSYS Fluent/NEPTUNE setup
Outputs	Simulation data
Responsible	TUG

Task	**Test Design**
Method	Storing large data files in MongoDB
Subsystem	Porto
Inputs	> 100 GB of Simulation data
Outputs	MongoDB document
Responsible	SINTEF

Task	**Test Design**
Method	Searching for data in MongoDB
Subsystem	Porto
Inputs	Data available in database, search criteria
Outputs	JSON document
Responsible	SINTEF

Task	**Test Design**
Method	Running jobs from the scripting shell
Subsystem	Porto
Inputs	Executable process to be run
Outputs	Result from process execution
Responsible	SINTEF

Task	**Test Design**
Method	Files and standard IO routines
Subsystem	Porto

Inputs	Data to be stored
Outputs	File on file system
Responsible	SINTEF

Task	**Test Design**
Method	Script Application Communication
Subsystem	Porto
Inputs	Data Message
Outputs	Data Message
Responsible	SINTEF

Task	**Test Design**
Method	Register Entities in the metadata-database
Subsystem	Porto
Inputs	Entity Schema
Outputs	Database Record
Responsible	SINTEF

Task	**Test Design**
Method	Defining workflows
Subsystem	Porto, NanoSim
Inputs	Simulation tools and setup data
Outputs	Simulation result
Responsible	SINTEF

Task	**Test Design**
Method	External file format driver
Subsystem	Porto
Inputs	Data file in external file format
Outputs	Porto entity
Responsible	SINTEF

Task	**Test Design**
Method	Data plotting
Subsystem	Porto
Inputs	Entity with array data
Outputs	Plot
Responsible	SINTEF

Task	**Test Design**
Method	MongoDB data storage
Subsystem	Porto
Inputs	Porto Entity, database, and collection name
Outputs	MongoDB entry containing data from the entity

Responsible	SINTEF

Task	**Test Design**
Method	Importing external data
Subsystem	Porto
Inputs	External File Format, driver plugin, entity definition
Outputs	MongoDB entry
Responsible	SINTEF

Task	**Test Design**
Method	Exporting data to proprietary data format
Subsystem	Porto
Inputs	Collection of data
Outputs	External file format with data
Responsible	SINTEF

Task	**Test Design**
Method	Source code generation from JSON
Subsystem	Porto
Inputs	JSON document, code template
Outputs	Compilable source code
Responsible	SINTEF

Task	**Test Design**
Method	Source code generation from MongoDB data
Subsystem	Porto
Inputs	Data available, code template
Outputs	Compilable source code
Responsible	SINTEF

2.1 Test Documentation requirements

The ***Level Test Plans*** that will outline the specific methods and details to achieve the indicated tests will follow the IEEE 829 Test Plan Structure.

2.2 Test administration requirements

Anomalies and issues discovered during testing will be submitted to the NanoSim/Porto issue tracker on GitHub.

2.3 Test reporting requirements

Appropriate test reports will be a part of the regular status updates.

3 General

3.1 Glossary

See List of definitions and abbreviations in FRD section 1.3

3.2 Document Change Log

Date	Description	Author(s)	Comments
15.08.2014	Initial version	Thomas F. Hagelien	
18.08.2014	RevRadl	Stefan Radl	Completed table in Ch. 2