SIMVA-SoS Design Document

1. ABOUT SIMVA-SoS

1.1. Motivation

The increasing scale and complexity of systems have reached a point that imposes qualitatively new demands on the existing system technology. The emerging system is characterized by distributed, decentralized, and networked combinations of autonomous sub-systems under large-scale and complex environments. This new "system" is a general concept of "System of systems" (SoS). The term has been arisen in various domains such as social infrastructures, smart grids, military services, and air traffic.

With respect to SoS, there are various types of definition from the literatures that have been independently published by industry, government and academia. One definition suggests that SoS is composed of independent sub-systems, which is called as Constituent Systems (CSs), and CS has two unique properties: (1) operational independence, i.e., each system operates independently and it achieves its own goals by itself and (2) managerial independence, i.e., each system is managed in most parts for its own goals rather than the goals of SoS. These indicate that CS is operated, managed, and developed independently, and it may not focus on the whole SoS-level goals. The behavior of an SoS emerges from the interaction among its CSs.

A key aspect of SoS is that how we can achieve whole SoS-level goals by orchestrating the interconnection between otherwise independent CSs. Thus, naturally, it is necessary to study approaches to verifying whether the SoS-level goals are achieved or not. However, there is a lack of supporting tools for modeling and verifying SoS and SoS-level goal achievements.

1.2. Goals

SIMVA-SoS is a software for simulating and verifying the SoS. SoS models are converted to executable SoS models and those become the inputs of simulation engine. Another input for simulation engine is SoS scenario data which is based on a target SoS. Also from SoS goal, verification property can be derived. Verification properties can be used for statistical verification.

1.3. Feature List

- Modeling
- Simulation
- Verification
- Slicing

2. ARCHITECTURE DESIGN

2.1. Architecture Diagram
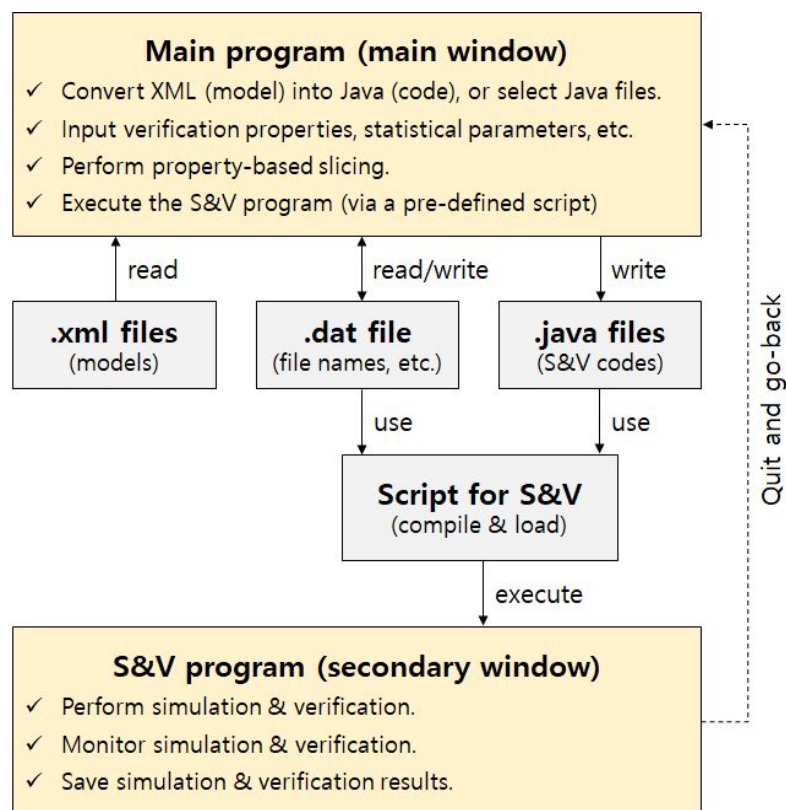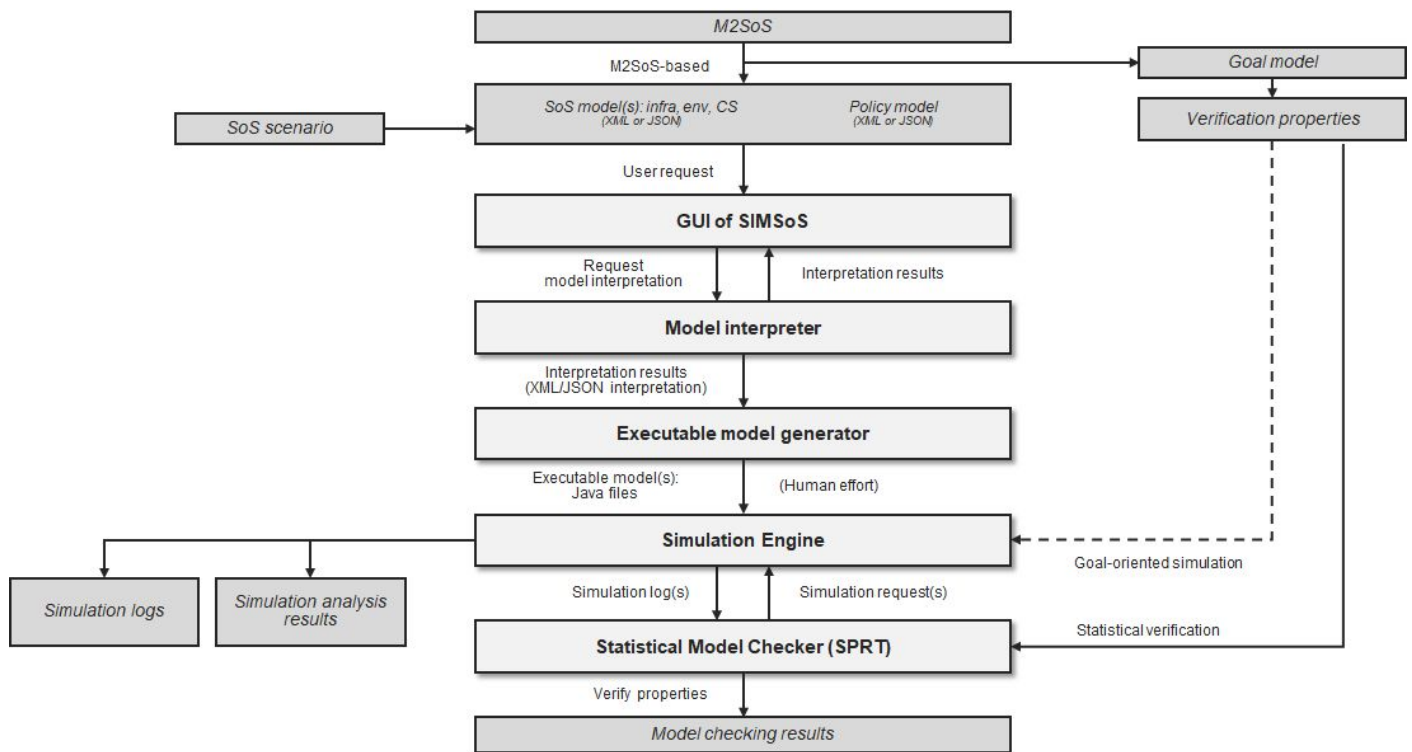
Update this program layer?

2.2.   Architecture Description

SIMVA-SoS is

3.   CLASS DESIGN

3.1.   Class Diagram

3.1.1.   Overview

3.1.2.   Executable Model Generation

(1) Controller

(2) Models

Overview

A. Organization & Constituent Systems

B. Environment

C. Simulation Scenario

3.1.3.   Simulation & Verification

3.1.4.   Slicing

3.2.   CLASS DESCRIPTION

The concepts of pattern come from MVC architectural pattern.

(We assumed that all attributes in classes have getter and setter method, so we omit all getting and setting method from class description.). Also we are going to use open-source program in order to use board.

4.   SEQUENCE DIAGRAM

5.   HUMAN INTERFACE DESIGN