

Analyzing SnBench Components using Alloy

CS511 term paper proposal by

Vatche Ishakian [visahak@cs.bu.edu] and Michalis Potamias [mp@cs.bu.edu]

Abstract

The *SnBench* is an ongoing project at Boston University. The aim of this project is to provide a programming and runtime infrastructure, for a well-provisioned sensor network. As part of its infrastructure, SnBench utilizes some open source components. *Alloy* is a structural modeling language developed at MIT. The *Alloy Analyzer* is a tool that takes as input a model written in Alloy and performs verification tasks to it. We intend to use Alloy and its Analyzer to model and certify various components that make up SnBench.

1. SnBench

SnBench aims to provide a scalable and easily accessible sensor network comprising of building blocks. Blocks may be sensors, algorithms, services etc. In its current state, SnBench uses a functional language called SNAFU (SensorNet Applications as Functional Units) to program user functionalities. This language enables SnBench to abstract its network substance from high-level users. A program written in SNAFU is compiled into an execution plan expressed in a language called STEP (Sensorium Task Execution Plan). The latter is then processed in order to be dispatched into appropriate execution environments.

2. Alloy

Alloy is a structural modeling language based on first-order logic. It can be used to assist the procedure of software verification using formal methods. Alloy has been developed for expressing complex structural constraints and behaviour. A software system can be modeled in Alloy as a logical formula. The Alloy Analyzer attempts to find values that make this formula true. Alloy and its Analyzer have already been used in case studies and in some cases they have revealed bugs in critical applications.

3. Proposal

SnBench is written in Java following the Object Oriented paradigm. As mentioned before, the nature of the project is to comprise of pluggable blocks. Thus, as long as these specifications hold we have reasons to believe that Alloy is suited to model, test and verify SnBench. Systems to be analyzed will be provided by member of SnBench project Michael Ocean. We note that our first target is to analyze the http-server currently used. NanoHTTPD is open source software and has been plugged in the SnBench framework. We hope that this task will help us familiarize ourselves with general notions of formal methods as well as with strengths and weaknesses of Alloy. In brief, we propose to complete the following steps:

1. Study Alloy language
2. Study application implementation (NanoHTTPD)
3. Build a model for the application to be studied
4. Validate the application using the Alloy Analyzer
5. Report difficulties in modeling, results of automated analysis

Depending on the success of this task we intend to repeat steps 2-5 for various critical SnBench applications that will be provided to us by project members.

References

[1] Azer Bestavros, Adam Bradley, Assaf Kfoury, and Micheal Ocean. SNBENCH: A Development and Run-Time Platform for Rapid Deployment of Sensor Network Applications. In Proceedings of the IEEE International Workshop on Broadband Advanced Sensor Networks (Basenets 2005), Boston MA, October 2005.

[2] Daniel Jackson, Software Abstractions: Logic, Language, and Analysis. Massachusetts Institute of Technology, Cambridge, MA, USA, 2006.

[3] Alloy website. <http://alloy.mit.edu/>

[4] SnBench website. <http://csr.bu.edu/snbench/>

[5] NanoHTTPD website. <http://elonen.iki.fi/code/nanohttpd/>