Wikiprint Book

Title: Semantic Analysis Framework (SAF)

Subject: ThirdEye - WikiStart

Version: 109

Date: 09/15/14 11:37:59

Table of Contents

Error: Failed to load processor `NewsFlash`

```
No macro or processor named 'NewsFlash' found
```

Semantic Analysis Framework (SAF)

SAF is a framework for offline data analysis of networked and distributed system data and represents the current stage of development in an ongoing research effort called the **ThirdEye** project. SAF enables users to analyse and reason over data using semantically meaningful abstractions and thus at a level closer to their understanding of system operation. Contrast this with the traditional analysis techniques that require users to operate over data at the low-level of attributes and values to infer semantic relationships from data.

Analysis using SAF can be broken down into two distinct phases.

Modeling phase

SAF enables system or domain experts to encode their high-level understanding of system behavior as **abstract models** over data. For example, the TCP connection setup model captures the abstract behavior of a TCP connection setup and the DNS Kaminsky Model captures behavior of the DNS Kaminsky experiment.

Models are written in a simple logic-based modeling language. The modeling language provides semantically relevant constructs to express relationships such as causality, ordering, concurrency, exclusions, combinations and dependency relationships between high-level system operations.

Analysis phase

Models drive analysis over data. Given timestamped raw-data, in the form of syslogs, packet dumps, alert logs, kernel logs, or application logs, the framework enables users to analyse the data by encoding their high-level questions directly as semantically meaningful models over data. Users can either writing their own models from scratch or build one by composing existing models from the knowledge-base.

- Read how the semantic analysis approach differs from other traditional approaches
- Read our NSDI'11 paper on the Semantic Analysis Framework
- Run a simple analysis example using SAF

Download and Installation

SAF is written in Python and uses a SQLite database as a backend for storage and processing. The current release is an alpha release, has been tested on Linux and is currently under active development and testing. This release is meant to test the waters and encourage feedback from the community.

Date	Release	Release Notes
07/05/11	saf-v0.2a.tar.gz (Linux)	This release fixes lots of bugs and is the latest recommended version. Please read the detailed release notes.

Please kindly report bugs and issues here.

Older releases available here.

- Dependencies
- Installation instructions
- Copyright notice and licence information

Another way to get the framework is from the anonymous read-only svn repository. This method is preferred if you wish to stay in sync with the latest bugfixes and features. All important fixes and features will be announced via the mailing list.

```
$ svn co http://thirdeye.deterlab.net/svn/trunk/SAF
```

Please send any feedback, suggestion or comments via email to Arun Viswanathan and report bugs, issues and feature requests by opening a ticket.

Quick start

- Run a simple analysis example using SAF
- Explore examples of real-world modeling examples using SAF
- Explore the current knowledge base of models.

SAF Components

There are four key components of SAF.

Modeling language

The modeling language is at the core of the semantic analysis process and provides semantically relevant abstractions to capture relationships like causality, ordering, concurrency, exclusions, combinations and dependencies between high-level system operations.

- Basic concepts and terminology - Explains the key ideas behind the modeling process and discusses the terminology.
- Language syntax and semantics - Explains the language syntax and semantics with examples for each feature.
- Anatomy of a model script - Discusses the basics behind writing a model.
- Language expressiveness - Discusses what can be modeled in the language.
- Current limitations of the language - Discusses what cannot be modeled today in the language.

Knowledge base

The knowledge base is intended to be a repository of common understanding in the form of models. Users can use existing models to compose higher-level models and can also contribute their models to the knowledge base. In the near future, we envision this knowledge base to encourage a more **share-and-reuse** approach to data analysis in networked systems.

- Building models by composing existing models - Discusses how new models can be built by composing existing models from the repository.

Data normalization plugins

Data normalization plugins convert raw data in any form to a uniform representation called events. Currently, there are plugins available for normalizing packet dumps, apache combined log files, syslog files and mysql files. The plugin framework provides easy mechanisms to add a new plugin to the framework.

- List of available plugins
- Normalizing raw packets using the p2db tool
- How to add a plugin to the framework? *Under construction*

Analysis and Presentation Framework

The analysis framework takes the user-specified models and applies them over the normalized events to extract events satisfying the model. The extracted events can be finally reported to user in various formats.

- Framework options - Discusses the various command-line options available.
- Current limitations of the analysis framework

Additional resources

- Architecture of the framework
- SAF features in the pipeline

Thank you for visiting. Stay tuned for more updates to the website. You are visitor number: VisitCounter(WikiStart)?