

# Deployment of contraval (Contract Validator): (User's Guide)

Carlos Molina-Jimenez<sup>1</sup>

School of Computing Science, Newcastle University, UK, [carlos.molina@ncl.ac.uk](mailto:carlos.molina@ncl.ac.uk)

**Abstract.** This document is a walk through description of the deployment and usage of version 1.0 of *contraval*. *Contraval* is a software tool implemented at Newcastle University UK, for validating the logical consistency of contracts conceived as a set of ECA (Event Condition Actions) rules.

The ECA rules that represent the contract are written in the EPROMELA language and validating with the SPIN model checker.

## 1 Introduction

The repository includes several examples located in folders. Each example can be run independently. Let us use the *BuyerStoreContract* folder which contains the *BuyerStoreContract.pml* example to explain how use *contraval*.

To run the examples you need to have Spin Version 6.1.0 or newer deployed on your local computer. Version of SPIN for Linux, Win and Mac users can be downloaded from the SPIN's home page [1].

We will assume a Linux machine, but the examples can be run in Win or Mac as well.

1. Copy one of the folders (say BuyerStoreContract) into a folder of your local computer.
2. The folder includes several files:
  - example independent macros: *setting.h* and *BizOperation.h* *vector.lpr* and *for.h*. These macros implement the business operations added to PROMELA and are used in all examples.
  - *BuyerStoreContract.pml* and *rule.h*. These files are example specific. In this example, they contain, respectively, the code for the events involved in the contract and the rules that are triggered by these events.
3. To run the code type:

```
% spin -a BuyerStoreContract.pml
% cc -o pan pan.c
% pan -a
```

4. The *BuyerStoreContract.pml* model is provided with several LTL formulae. You can edit the file to comment and uncomment them as needed. Keep in mind that SPIN can verify only a single LTL at a time.

A detailed explanation of the *BuyerStoreContract.pml* can be found in a technical report ([2]) which is also include in the *BuyerStoreContract* folder.

## 2 Implementation

Details about the implementation of *contraval* and additional case studies can be found in Abubkr Abdelsadiq's PhD dissertation that originated this work [3].

## 3 Licence

The *contraval* tool is released under the Apache License, Version 2.0[4], which is available from Apache's web pages. Also, you can find a *txt* copy from our home page [5].

## 4 Implementation History

**Table 1.** Contraval-implementation history.

Version	Date	Contributors	Key features
1.1	Sep 2012	Abubkr Abdelsadiq	Implementation of macros for the specification of business operations.

## Acknowledgment

The author was partially funded by EPSRC grant KTS-EP/H500332/1. Discussions with Gary Brown from Red Hat concerning the tool framework and SAVARA and constructive comments from Stuart Wheeler from Arjuna Technologies, have been useful.

## References

1. SPIN: On-the-fly, ltl model checking with spin. <http://spinroot.com> (visited in Jul 2012 2012)
2. Molina-Jimenez, C., Shrivastava, S.: Establishing conformance between contracts and choreographies. Technical Report CS-TR-1383, School of Computing Science, Newcastle Univ. UK (April 2013)
3. Abdelsadiq, A.A.: A Toolkit for model checking of electronic contracts. PhD thesis, School of Computing Science, Newcastle University, UK (September 2012)
4. Foundation, T.A.S.: Apache license version 2.0, january. <http://www.apache.org/licenses> (2004)
5. Molina-Jimenez, C.: Carlos molina-jimenez home page. <http://homepages.cs.ncl.ac.uk/carlos.molina> (2012)