



# *CosyVerif*: An Open Source Extensible Verification Environment

Étienne André, Lom Hillah, Francis Hulin-Hubard, Fabrice Kordon,  
Yousra Lembachar, Alban Linard, Laure Petrucci

ENS Cachan, Univ. Paris 6, Univ. Paris 13

17th July 2013

# *CosyVerif*



Many tools for distributed systems verification

- Relying on **different formalisms**
- Solving **different problems**
- Running on different OS
- Requiring some difficult installation procedures

⇒ Needs for:

- A **unified representation**
- **Interoperability** and tool integration
- For users: Easy installation and use
- For developers: Easy integration of tools



- 1 The CosyVerif Environment
- 2 Integrated Tools
- 3 Summary and Evolutions



- 1 The CosyVerif Environment
- 2 Integrated Tools
- 3 Summary and Evolutions



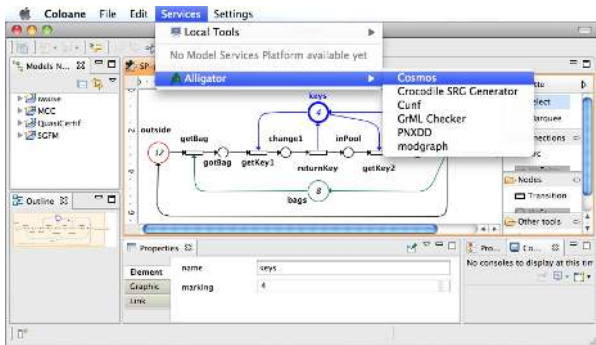
# A Common and Reusable Syntax

- *CosyVerif* relies on reusable and extensible formalisms  
[André et al., 2013]
- **FML** (Formalism Markup Language)
  - ▶ Describes formalisms (meta-models)
- **GrML** (Graph Markup Language)
  - ▶ Describes models
- Advantages
  - ▶ Unified model representation
  - ▶ Easy addition of new formalisms
- Applications: set of formalisms
  - ▶ Large family of (timed) automata and Petri nets



# Web-services Client-Server Architecture

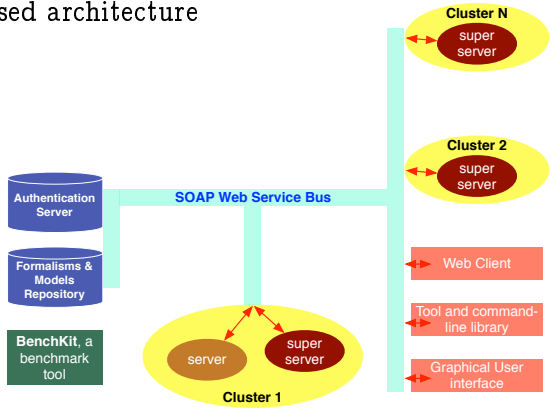
- **Easy to install:** simple and light multi-platform client connecting to servers





# Web-services Client-Server Architecture

- Tool invocation through Web services transparent to the end-user
- **Cloud**-based architecture





- 1 The CosyVerif Environment
- 2 Integrated Tools
- 3 Summary and Evolutions





# Tools Integration in CosyVerif

Numerous tools integrated to CosyVerif

- One official client: **Coloane** (platform-independent)
- 6 tools and 9 integrated services
  - ▶ Integration via Web services: easy to use and compose
- Multiple formalisms supported (Petri nets and extensions, hybrid automata, timed automata... and more!)





## State space generation & CTL verification [Colange et al., 2011]

- Symbolic/symbolic approach based on **Symmetric Nets with Bags** [Haddad et al., 2009]
- Two symbolic techniques to counter state space explosion
  - ① **symmetries** to reduce the reachability graph [Chiola et al., 1991]
  - ② **hierarchical Set Decision Diagrams** to store the reachability graph [Couvreur and Thierry-Mieg, 2005]



## Unfolding-based verification of Petri nets with read arcs (contextual nets) [Baldan et al., 2012]

- Features

- ▶ Unfolding construction tool [Rodríguez et al., 2011]
- ▶ Reachability and deadlock checking tool [Rodríguez and Schwoon, 2012]

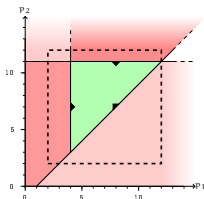
- Characteristics

- ▶ Unfoldings fully represent the state space of a c-net by a **partial order** rather than by a set of interleavings
  - ★ Often exponentially smaller than the state space, and never larger
- ▶ c-net unfoldings can be exponentially more **compact** than those of corresponding Petri nets [Baldan et al., 2012]



## Parameter synthesis for real-time systems [André et al., 2012]

- Quantitative robustness analysis
  - ▶ “Can we increase some of the timing delays such that the system still behaves well?”
- Schedulability analysis
- Hybrid system verification





## Construction and analysis of **modular state spaces**

[Lakos and Petrucci, 2004]

- Modular State Spaces for Synchronised Automata
  - ▶ synchronisation structure
  - ▶ only reachable parts of the automata
- Analysis
  - ▶ forward and backward reachability
  - ▶ deadlock-checking
  - ▶ liveness



## State space generation and CTL formulæ evaluation on P/T nets [Hong et al., 2012]

- Handles **Symmetric Nets** through their **unfolding** into an equivalent P/T net
- Exploits **hierarchy**: a state is seen as a tree, where the leaves correspond to place markings
- Relies on Set Decision Diagrams [Couvreur and Thierry-Mieg, 2005]



- 1 The CosyVerif Environment
- 2 Integrated Tools
- 3 Summary and Evolutions**





# An Open Environment

- Entirely **open source**
- **Open to contributions**
  - ▶ Tool integration
  - ▶ Alternative clients
  - ▶ New formalisms
- A **repository of models** using a common syntax
  - ▶ Coming from the integrated tools, and the model checking contests  
[\[Kordon et al., 2013\]](#)



# Recent and Ongoing Evolutions

- **Asynchronous tool invocation**
  - ▶ Get the result later (e.g., by email)
- **Federation of servers** and use of clusters
  - ▶ Enable load balancing
- **Repository** of formalisms and models
- **Command-line** version of the underlying platform



# Future Evolutions

- **Enhanced interaction** between tools
  - ▶ Output of a tool as input of another one
- Handling **semantics** (bridges between formalisms)
  - ▶ Also allows system simulation
- Handling **heterogeneous models** (mixing different formalisms)



# Future Evolutions

- **Enhanced interaction** between tools
  - ▶ Output of a tool as input of another one
- Handling **semantics** (bridges between formalisms)
  - ▶ Also allows system simulation
- Handling **heterogeneous models** (mixing different formalisms)

Try it!

<http://cosyverif.org/>





# Bibliography



# References I



André, É., Barbot, B., Démoulin, C., Hillah, L. M., Hulin-Hubard, F., Kordon, F., Linard, A., and Petrucci, L. (2013).  
**A modular approach for reusing formalisms in verification tools of concurrent systems.**  
In *ICFEM*, Lecture Notes in Computer Science. Springer.  
To appear.



André, É., Fribourg, L., Kühne, U., and Soulat, R. (2012).  
**IMITATOR 2.5: A tool for analyzing robustness in scheduling problems.**  
In *Formal Methods*, volume 7436 of *Lecture Notes in Computer Science*, pages 33–36.  
Springer.








Baldan, P., Bruni, A., Corradini, A., König, B., Rodríguez, C., and Schwoon, S. (2012).  
**Efficient unfolding of contextual Petri nets.**  
*Theoretical Computer Science*, 449:2–22.



Ballarini, P., Djafri, H., Duflot, M., Haddad, S., and Pekergin, N. (2011).  
**HASL: An expressive language for statistical verification of stochastic models.**  
In *VALUETOOLS*, pages 306–315.



## References II

-  Chiola, G., Dutheillet, C., Franceschinis, G., and Haddad, S. (1991).  
On well-formed coloured nets and their symbolic reachability graph.  
In *ICATPN*. Springer-Verlag.
-  Colange, M., Baarir, S., Kordon, F., and Thierry-Mieg, Y. (2011).  
Crocodile: A symbolic/symbolic tool for the analysis of symmetric nets with bags.  
In *ICATPN*, volume 6709 of *Lecture Notes in Computer Science*, pages 338–347.  
Springer.
-  Couvreur, J.-M. and Thierry-Mieg, Y. (2005).  
Hierarchical decision diagrams to exploit model structure.  
In *FORTE*, volume 3731 of *Lecture Notes in Computer Science*, pages 443–457.  
Springer.
-  Haddad, S., Kordon, F., Petrucci, L., Pradat-Peyre, J.-F., and Trèves, N. (2009).  
Efficient state-based analysis by introducing bags in Petri net color domains.  
In *ACC*, pages 5018–5025. Omnipress IEEE.
-  Hong, S., Kordon, F., Paviot-Adet, E., and Evangelista, S. (2012).  
Computing a hierarchical static order for decision diagram-based representation from P/T nets.  
*Transactions on Petri Nets and Other Models of Concurrency*, V:121–140.



## References III



Kordon, F., Linard, A., Becutti, M., Buchs, D., Fronc, L., Hulin-Hubard, F., Legond-Aubry, F., Lohmann, N., Marechal, A., Paviot-Adet, E., Pommereau, F., Rodrigues, C., Rohr, C., Thierry-Mieg, Y., Wimmel, H., and Wolf, K. (2013).  
**Web report on the model checking contest @ Petri Net 2013.**

Available at <http://mcc.lip6.fr>.



Lakos, C. and Petrucci, L. (2004).  
**Modular analysis of systems composed of semiautonomous subsystems.**  
In *ACSD*, pages 185–196. IEEE Computer Society.



Rodriguez, C. and Schwoon, S. (2012).  
**Verification of Petri nets with read arcs.**  
In *CONCUR*, volume 7454 of *Lecture Notes in Computer Science*, pages 471–485.



Rodriguez, C., Schwoon, S., and Baldan, P. (2011).  
**Efficient contextual unfolding.**  
In *CONCUR*, volume 6901 of *Lecture Notes in Computer Science*, pages 342–357.







# Source of the graphics used



Title: Scottish Kitten

Author: RN3DLL

Source: [https://commons.wikimedia.org/wiki/File:Scottish\\_Kitten.png](https://commons.wikimedia.org/wiki/File:Scottish_Kitten.png)

License: CC BY-SA 3.0



## License of this document

This presentation can be published, reused and modified under the terms of the Creative Commons license **Attribution-ShareAlike 3.0 Unported (CC BY-SA 3.0)**

Author: *The CosyVerif team*



<https://creativecommons.org/licenses/by-sa/3.0/>