



# CosyVerif et l'enseignement des réseaux de Petri à l'Université Pierre & Marie Curie

Fabrice Kordon

LIP6/MoVe, UPMC

20 décembre 2013

# CosyVerif



# Le master informatique à l'UPMC

Université Pierre & Marie Curie - Télécom ParisTech

**UPMC** **Master Sciences et Technologies**  
Mention «Informatique»

**SYSTEMES ET APPLICATIONS RÉPARTIS**

<http://www-master.ufr-info-p6.jussieu.fr/md/specialite/sar/>  
[master.info.sar@upmc.fr](mailto:master.info.sar@upmc.fr)

Responsable  
F. Kordon

Responsable adjointe  
B. Bénard

Coordination  
A. Le Conte

Programmation pair-à-pair  
Conception et vérification formelle de systèmes critiques  
Systèmes d'exploitation pour processeurs multi-cœurs  
Programmation pour plate-formes mobiles  
Architecture et sécurité des systèmes  
Systèmes embarqués  
Cloud computing

Parcours:  
Conception et Réalisation d'Applications Réparties  
Systèmes Répartis et Middleware  
Systèmes Répartis Embarqués Temps-Réel  
Spécialisation en Ingénierie des Applications Réparties (seconde compétence)  
Informatique Appliquée à la Musique (ATIAM)



**1 minute de publicité!!!**



# Le master informatique à l'UPMC

## Spécialité «Systèmes et Applications Répartis»

- Créée en 2004 (suite du DEA SIR lui-même suite du DEA SI),
- $\simeq 35 + 10$  étudiants en M2,
- Enseignements de **modélisation** et de vérification.



# Le master informatique à l'UPMC

## Spécialité «Systèmes et Applications Répartis»

- Créée en 2004 (suite du DEA SIR lui-même suite du DEA SI),
- $\simeq 35 + 10$  étudiants en M2,
- Enseignements de **modélisation** et de vérification.

## Enseignements sur la modélisation des systèmes répartis

- Automates / **réseaux de Petri**,
- Pratique de la modélisation en réseaux de Petri,
- Un **TP** (prise en main) et un **projet** (avec CPN-AMI jusqu'en 2012).



# Le master informatique à l'UPMC

## Spécialité «Systèmes et Applications Répartis»

- Créée en 2004 (suite du DEA SIR lui-même suite du DEA SI),
- $\simeq 35 + 10$  étudiants en M2,
- Enseignements de **modélisation** et de vérification.

## Enseignements sur la modélisation des systèmes répartis

- Automates / **réseaux de Petri**,
- Pratique de la modélisation en réseaux de Petri,
- Un **TP** (prise en main) et un **projet** (avec CPN-AMI jusqu'en 2012).

# Défis pour CosyVerif

Déployer une salle de TP

Permettre un usage sur l'ordinateur personnel des étudiants



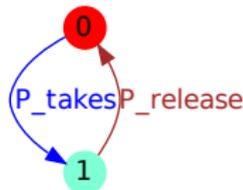
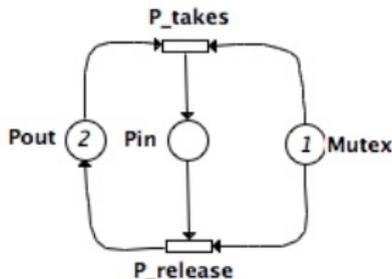
# Objectifs pédagogiques

## TP de prise en main des réseaux de Petri

- Relations entre le modèle et l'espace d'états,
- Notions sur les invariants,
- Bases du typage des données (couleurs),
- Requêtes simples sur l'espace d'états,
- **CosyVerif** utilisé en salle machine.

### Services utilisés :

- Prod,
- Bounds,
- Invariants (GreatSPN).





# Objectifs pédagogiques

## Le projet

- Modéliser un problème réel avec du parallélisme,
- Véhicules automatisés sur un pont à voie unique + notion de time-out,
- Requêtes CTL simples,
- Utilisation de prod (ancien mais très intéressant pédagogiquement).
- **CosyVerif utilisé en salle machine et sur leur machine.**

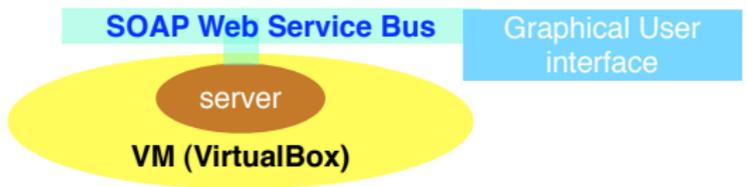




# Problématique du déploiement

## Architecture :

- Coloane (bundle Eclipse),
- Alligator+outils (VM Linux).





# Problématique du déploiement

Architecture :

- Coloane (bundle Eclipse),
- Alligator+outils (VM Linux).



Avant octobre 2013

- 1 Installer VirtualBox,
- 2 Installer Coloane,
- 3 Installer l'image disque,
- 4 Lancer la VM,
- 5 Lancer Coloane.

Difficile !

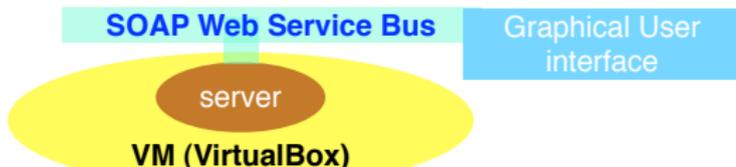
Administrateur – Utilisateur



# Problématique du déploiement

Architecture :

- Coloane (bundle Eclipse),
- Alligator+outils (VM Linux).



Avant octobre 2013

- 1 Installer VirtualBox,
- 2 Installer Coloane,
- 3 Installer l'image disque,
- 4 Lancer la VM,
- 5 Lancer Coloane.

Difficile !

Après octobre 2013

- 1 Installer VirtualBox,
- 2 Installer CosyVerif4<B>,
- 3 Lancer CosyVerif4<B>.

Facile ! (pour l'utilisateur)

Administrateur – Utilisateur



## Packager !



## Packager !

Coloane





## Packager !

Coloane  
+ Alligator (services Rdp)





## Packager !

Coloane  
+ Alligator (services Rdp)  
+ shell d'invocation





## Packager !

Coloane  
+ Alligator (services Rdp)  
+ shell d'invocation





## Packager !

Coloane  
+ Alligator (services RdP)  
+ shell d'invocation





## Packager !

Coloane  
+ Alligator (services Rdp)  
+ shell d'invocation



Linux → le script shell



## Packager !

Coloane  
+ Alligator (services Rdp)  
+ shell d'invocation



Linux → le script shell  
MacOS → «.app» (Platypus)





## Packager !

Coloane  
+ Alligator (tous services)  
+ shell d'invocation



Linux → le script shell  
MacOS → «.app» (Platypus)  
RdP/tous les outils



## Packager !

Coloane  
+ Alligator (tous services)  
+ shell d'invocation



Linux → le script shell  
MacOS → «.app» (Platypus)  
Autre sélection d'outils...





## Packager !

Coloane  
+ Alligator (tous services)  
+ shell d'invocation



Linux → le script shell  
MacOS → «.app» (Platypus)  
Autre sélection d'outils...



### Installation/Invocation d'un seul outil

(si VirtualBox installé)



## Bundle dédié aux réseaux de Petri

Embarque les outils stables du groupe **MeFoSyLoMa...**

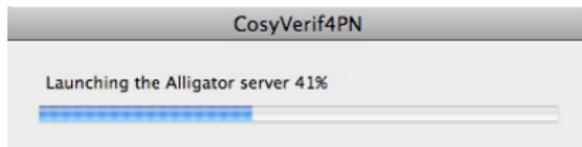
- 1 Behavioral Verification
  - ▶ **Prod**, **cunf**, **PNXDD**,  
**Crocodile**
- 2 Stochastic Analysis
  - ▶ **Cosmos**
- 3 Structural Analysis
  - ▶ **Invariants (greatSPN)**,  
**Bounds**
- 4 Import/Export
  - ▶ **unfold to P/T**, **PNML**  
**export**



Lors de l'invocation (Mac)



## Lors de l'invocation (Mac)





## Lors de l'invocation (Mac)

```
Bundle-PN [Running]
Debian GNU/Linux jessie/sid bundle-pn tty1

bundle-pn login: CONFIG : Setting up configuration... (org.cosyverif.Configuration:
updateConfiguration@org.cosyverif.Configuration)
CONFIG : Hostname is: 10.0.2.15 (org.cosyverif.Configuration:updateConfigura
tion@org.cosyverif.Configuration)
CONFIG : Port is: 9000 (org.cosyverif.Configuration:updateConfiguration@org.
cosyverif.Configuration)
CONFIG : Administration hostname is: 127.0.0.1 (org.cosyverif.Configuration:
updateConfiguration@org.cosyverif.Configuration)
CONFIG : Administration port is: 9001 (org.cosyverif.Configuration:updateCon
figuration@org.cosyverif.Configuration)
CONFIG : Debug is: off (org.cosyverif.Configuration:updateConfiguration@org.
cosyverif.Configuration)
-
```

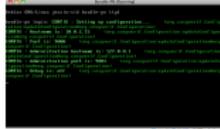


## Lors de l'invocation (Mac)

The screenshot displays the Coloane software interface for modeling and verifying Petri nets. The main window shows a Petri net diagram with the following components:

- Places:** 'outside' (12 tokens), 'keys' (4 tokens), and 'bags' (8 tokens).
- Transitions:** 'getBag', 'gotBag', 'getKey1', 'change1', 'returnKey', 'inPool', 'getKey2', and 'change2'.
- Connections:** Arcs connect 'outside' to 'getBag', 'gotBag' to 'outside', 'keys' to 'change1', 'change1' to 'keys', 'keys' to 'inPool', 'inPool' to 'keys', 'bags' to 'getKey1', 'getKey1' to 'bags', 'getKey2' to 'bags', and 'change2' to 'getKey2'.

The interface includes a 'Model...' sidebar with a tree view of models, a 'Palette' on the right with tools like 'Select', 'Marquee', 'Arc', 'Inhibitor', 'Nodes', 'Transition', 'Place', 'Note', and 'Link', and a bottom panel with 'Properties' (showing 'Properties are not available.') and 'Problems/Console' (showing 'No consoles to display at this time.').





## Identique sous Linux (mais directement via le script shell)

The screenshot displays two terminal windows. The left window, titled "Bundle-PN [Running] - Oracle VM VirtualBox", shows the configuration of the Alligator server. The right window, titled "fko@localhost:~/CosyVerif4PN — ssh — 87x28", shows the execution of the shell script.

```
Machine View Devices Help
CONFIG : Add org.cosyverif.service.greatspn.Translation as org
or.service.BinaryService. (org.cosyverif.alligator.Loader
verif.alligator.Loader
CONFIG : Add org.cosyverif.service.BinaryService.
.alligator.Loader)
CONFIG : Add org.cosyverif.alligator.Services.
.alligator.Loader)
INFO : Unloading mod
IF.alligator.Loader:re
INFO : Reloading all
(org.cosyverif.alli
INFO : Unloading mod
(org.cosyverif.alliga
INFO : Reloading all
services... (org.c
Loader)
CONFIG : Administratio
(org.cosyverif.All
CONFIG : Execution ser
syverif.Alligator:run@
CONFIG : Information s
rg.cosyverif.Alligator
```

```
fko@localhost ~]$ cd CosyVerif4PN/
fko@localhost CosyVerif4PN]$ bash startCosyVerif.sh
Waiting for VM "Bundle-PN" to power on...
VM "Bundle-PN" has been successfully started.

*****
Thanks for using CosyVerif4PN (v 1.0.2)
CosyVerif4PN is extracted from CosyVerif, a formal
verification environment, see http://cosyverif.org
for more details on this project.
*****

Launching the Alligator server
*****
/home/fko/CosyVerif4PN
fko@localhost CosyVerif4PN]$ [INFO] Computing menu of server: Local Alligator
(api.alligator.Connection.run)
[INFO] Trying to compute menu for new-style alligator. (api.a
lligator.Connection.run)
[INFO] Connecting to https://localhost:9000/services... (api.a
lligator.Connection.computeMenu)
[INFO] Computing menu... (api.a
lligator.Connection.computeMenu)
[INFO] Retrieving task identifiers... (api.a
lligator.Identifiers.getIdentifiers)
[INFO] Waiting 2 minutes before update of server: Local Alligator
lligator.Connection.run)
fko@localhost CosyVerif4PN]$
```



# Installation

MacOS

Drag and Drop!!!!

Linux

Copie de fichiers!!!!



MacOS

Drag and Drop!!!!

Linux

Copie de fichiers!!!!



**Attention aux droits d'accès**  
Il y a des fichiers dans le bundle



MacOS

Drag and Drop!!!!

Linux

Copie de fichiers!!!!



**Attention aux droits d'accès**

Il y a des fichiers dans le bundle

**Attention au «workspace»**

Le bundle-Eclipse embarque son propre «workspace»

Il faut utiliser les «import/export CAMI»



# Problèmes rencontrés (et corrigés)

## Installation en salle de TP

- Installation partagée par plusieurs comptes
  - Au gré des logins des étudiants,
  - Eclipse laissait des fichiers temporaires.
- Problème de sécurité
  - Les étudiants ne doivent pas pouvoir installer leur propre VM!!!
  - Unix : remplacement du script de lancement,
  - MacOS : utilisation d'un compte local (résoud le 1<sup>er</sup> problème).



# Problèmes rencontrés (et corrigés)

## Installation en salle de TP

- Installation partagée par plusieurs comptes
  - Au gré des logins des étudiants,
  - Eclipse laissait des fichiers temporaires.
- Problème de sécurité
  - Les étudiants ne doivent pas pouvoir installer leur propre VM!!!
  - Unix : remplacement du script de lancement,
  - MacOS : utilisation d'un compte local (résoud le 1<sup>er</sup> problème).

## Installation sur les machines personnelles

- Pas de soucis rapporté,
- Projets rendus (mais pas encore corrigés).



# Problèmes rencontrés (et corrigés)

## Installation en salle de TP

- Installation partagée par plusieurs comptes
  - Au gré des logins des étudiants,
  - Eclipse laissait des fichiers temporaires.
- Problème de sécurité
  - Les étudiants ne doivent pas pouvoir installer leur propre VM!!!
  - Unix : remplacement du script de lancement,
  - MacOS : utilisation d'un compte local (résoud le 1<sup>er</sup> problème).

## Installation sur les machines personnelles

- Pas de soucis rapporté **mais il peut y en avoir...**,
- Projets rendus (mais pas encore corrigés).



# Problèmes rencontrés (et corrigés)

## Installation en salle de TP

- Installation partagée par plusieurs comptes
  - Au gré des logins des étudiants,
  - Eclipse laissait des fichiers temporaires.
- Problème de sécurité
  - Les étudiants ne doivent pas pouvoir installer leur propre VM!!!
  - Unix : remplacement du script de lancement,
  - MacOS : utilisation d'un compte local (résoud le 1<sup>er</sup> problème).

## Installation sur les machines personnelles

- Pas de soucis rapporté,
- Projets rendus (mais pas encore corrigés).

**Avantage : approche répartie (OK pour la charge)**



## Ça fonctionne

- Nous y avons gagné une solution multi-plateforme
  - MacOS/Linux – pratique pour les étudiants,
- Et Windows? pas de packaging mais...
  - La VM du serveur est la même!
  - Coloane existe sous Windows
  - Bundle possible (mais problème de savoir faire + temps)
- **Gros travail de documentation!**



## Ça fonctionne

- Nous y avons gagné une solution multi-plateforme
  - MacOS/Linux – pratique pour les étudiants,
- Et Windows ? pas de packaging mais...
  - La VM du serveur est la même !
  - Coloane existe sous Windows
  - Bundle possible (mais problème de savoir faire + temps)
- **Gros travail de documentation !**

## Et après ?

Rentrée 2014, changement de maquette à l'UPMC...

- Le cours de modélisation est éclaté
  - une partie en M1, l'autre en M2,
- Donc des besoins plus complexes – Nous pensons être prêts
- **Et vous ?**



## Automatisation dans la «build farm»

- Production plus automatisée  
→ tout est déjà paraméré
- Permettre aux doctorants de faire leur propre bundle





## Quelques URLs

- <http://download.cosyverif.org> (les distributions)
- <http://cosyverif.org/index.php?N1=2&N2=3> (documentation)

**CosyVerif**  
The Petri Net Based Services

**Introduction**

A **Petri net** is a mathematical modeling language particularly adapted to the description of distributed systems. It is a directed bipartite graph, in which the nodes represent transitions (i.e. events that may occur, represented by rectangles or bars) and places (i.e. conditions, represented by circles or ellipses). The directed arcs describe which places are pre and/or postconditions for which transitions.

Petri Nets were first defined by the German mathematician **Cari Adam Petri**.

There are numerous Petri net dialects and **CosyVerif** handles several of them:

- P/T nets,
- Symmetric nets, a subclass of colored nets,
- Symmetric nets with bags, an extension of Symmetric nets.

The pages of this section present the formalism itself, some illustrating examples, as well as the services for Petri net-based Verification available in **CosyVerif**

**Available services for Petri nets**

So far, the following services are available in **CosyVerif** (full bundle or Petri Net Bundle):

- Behavioral Analysis:
  - Model Checking with PROD (Univ. Helsinki),
  - Model Checking with PNxDD (Univ. P. & M. Curie, LIP6),
  - Model Checking with Crocodile (Univ. P. & M. Curie, LIP6),
  - Curf (ENS de Cachan, LSV),
- Stochastic Analysis:

**Navigation**

- Petri net Home
- Grammars**
  - P/T Nets
  - Symmetric Nets
  - Symmetric Nets with Bags
- Examples**
  - P/T Nets
  - Symmetric Nets
  - Symmetric Nets with Bags
- Services**
  - P/T Nets
    - P/T invariant (GreatSPN)
    - P/T Bounds (Bounds)
  - Symmetric Nets
    - Model Checking (Prod)
  - Stochastic Nets
    - Probabilistic Model Checking (Cosmos)