
CADP : une boîte à outils pour la construction et l'analyse de processus distribués

H. Garavel, [F. Lang](#), R. Mateescu, W. Serwe

Inria - équipe Convecs

<http://convecs.inria.fr>



Qu'est-ce que CADP ?

- Une boîte à outils pour vérifier des systèmes asynchrones
- A l'intersection de plusieurs branches de la science informatique :
 - Théorie de la concurrence
 - Méthodes formelles
 - Vérification assistée par ordinateur
 - Construction de compilateurs
- Un travail au long cours
 - Développement de CADP démarré au milieu des années 80...
 - Initialement : seulement **deux outils** (Caesar et Aldebaran)
 - amélioration continue depuis près de 30 ans
 - Aujourd'hui (**CADP 2015**) : plus de **50 outils**



Principales fonctionnalités de CADP

Langages de spécification

- Sémantique formelle
- Basés sur les calculs de processus
- Syntaxe conviviale

Paradigmes de vérification

- **Model checking**
(μ -calculus modal étendu)
- **Equivalence checking**
(bisimulations)
- **Visual checking**
(dessin de graphes)

Techniques de vérification

- Analyse d'atteignabilité
- Vérification à la volée
- Vérification compositionnelle
- Vérification distribuée
- Analyse statique

Autres fonctionnalités

- Simulation pas à pas
- Prototypage rapide
- Génération de cas de test
- Evaluation de performance



Spécificités de CADP

- Programmes parallèles (plutôt que séquentiels)
- Passage de messages (plutôt que mémoire partagée)
- Langages à sémantique formelle (calculs de processus)
- Structures de données dynamiques (records, listes, arbres, etc.)
- Techniques énumératives (plutôt que symboliques)
- Basé sur les actions (plutôt que sur les états)
- Logiques arborescentes (plutôt que linéaires)



Domaines d'application

- Les études de cas traitées avec CADP couvrent les domaines suivants :
avionics, bioinformatics, business processes, cognitive systems, communication protocols, component-based systems, constraint programming, control systems, coordination architectures, critical infrastructures, cryptography, database protocols, distributed algorithms, distributed systems, e-commerce, e-democracy, embedded software, grid services, hardware design, hardware/software co-design, healthcare, human-computer interaction, industrial manufacturing systems, middleware, mobile agents, model-driven engineering, networks, object-oriented languages, performance evaluation, planning, radiotherapy equipments, real-time systems, security, sensor networks, service-oriented computing, software adaptation, software architectures, stochastic systems, systems on chip, telephony, transport safety, Web services
- Liste d'études de cas publiées :
<http://cadp.inria.fr/case-studies>



Accent mis sur la qualité du logiciel

- **But** : outils robustes
- **Moyens mis en œuvre** :
 - Processus de développement maîtrisé
 - Evolutions contrôlées (revue de code)
 - Attention particulière portée à la compatibilité ascendante
 - Test intensif (tests nocturnes)
 - Documentation précise



Quelques chiffres sur CADP

- **Dissémination internationale**
 - En 2012, la **10,000^e** licence de CADP a été accordée
 - **441** contrats de licence signés sur **5** continents
 - **171** études de cas effectuées avec CADP
 - **78** outils de recherche connectés à CADP
 - \geq **230** utilisateurs et **1400** messages sur le forum CADP depuis 2007
- **Plusieurs architectures supportées**
 - Processeurs : Itanium, PowerPC, Sparc, x86, x64
 - Systèmes d'exploitation : Linux, MacOS X, Solaris, Windows
 - Compilateurs C : gcc3, gcc4, Intel, Sun



Quelques recherches en cours

- Application au sein de projets en cours :
 - Protocoles (déploiement, reconfiguration, ...) pour le cloud computing (projet OpenCloudware)
 - Réseaux de calculateurs logiques programmables / systèmes GALS (projet Bluesky)
 - Protocoles de cohérence de cache (avec STMicroelectronics)
 - IHM pour les centrales nucléaires (projet Connexion)
- Développements en cours :
 - Génération automatique de code distribué
 - Vérification sur des modèles stochastiques et probabilistes
 - Amélioration des langages et des outils



Un futur prometteur

- **La concurrence est omniprésente**
 - Matériel : CPU multicœurs, clusters, grilles, nuages, ...
 - Logiciel : la concurrence devient nécessaire pour exploiter les nouveaux matériels
- **Impact sur la vérification**
 - Analyse formelle de systèmes concurrents
 - Vérification massivement parallèle
- **Prise de conscience dans l'industrie**
 - Besoin croissant de matériel et logiciel fiables
 - Les modèles (même non-formels) deviennent une pratique standard



Pour plus d'information...

- TP sur CADP à ETR'15
- Site Web de CADP : <http://cadp.inria.fr>

