

# Application BASTRI

## Fiches Equipes

### PI.R2 (SR0419MR)

Conception, étude et implémentation de langages pour les preuves et les programmes

PI.R2 □ PI.R2 (SR0368UR)

**Statut:** Terminée

**Responsable :** Pierre-louis Curien

**Mots-clés de "A - Thèmes de recherche en Sciences du numérique - 2023" :** *Aucun mot-clé.*

**Mots-clés de "B - Autres sciences et domaines d'application - 2023" :** *Aucun mot-clé.*

**Domaine :** Algorithmique, programmation, logiciels et architectures  
**Thème :** Programmation, vérification et preuves

**Période :** 01/01/2009 -> 01/01/2011

**Dates d'évaluation :**

**Etablissement(s) de rattachement :** <sans>

**Laboratoire(s) partenaire(s) :** <sans UMR>

**CRI :** Centre Inria de Paris

**Localisation :** Rocquencourt

**Code structure Inria :** 021092-0

**Numéro RNSR :** 200920939A

**N° de structure Inria:** SR0419MR

### Présentation

L'équipe  $\pi r^2$  constituée au sein du laboratoire Preuves, Programmes et Systèmes (UMR 7126), couvre trois champs de recherche :

- Un champ de recherche fondamentale autour de la correspondance entre preuves et programmes
- Un champ de recherche théorique autour du formalisme sur lequel repose l'environnement de développement de preuves et de programmes **Coq**
- Un champ implémentatoire avec le développement du logiciel **Coq**, notamment dans ses aspects langage de programmation à types « riches »

### Axes de recherche

- Fertilisation croisée entre la théorie de la démonstration et la théorie des programmes  
La correspondance syntaxique entre preuves et programmes, dite correspondance de Curry-Howard a été au cours des 30 dernières années un formidable moteur de la recherche tant pour la logique (la plupart des axiomes logiques sont calculatoires : par exemple, la preuve d'un énoncé existentiel est un procédé de calcul qui ultimement exhibe un témoin de cette existence) que pour la programmation (les types de données sont des formules spécifiant les propriétés des programmes et ces types peuvent être des formules arbitrairement complexes). Dans les 15 dernières années, de nouvelles étapes ont été franchies (mise en évidence d'une connexion entre raisonnement par l'absurde et opérateurs de contrôle, entre le formalisme logique dit « du calcul des séquents » et les structures des machines d'évaluation abstraites, rôle des effets de bord dans l'interprétation calculatoire de l'axiome du choix, ...). L'un des objectifs de  $\pi r^2$  est alors d'explorer les conséquences et fondements de ces dernières avancées, notamment en étudiant le contenu logique des délimiteurs de continuations issus de la programmation fonctionnelle, et en étudiant les relations entre les mécanismes de réflexion/réification en programmation et de correction/complétude en logique.
- Étude du Calcul des Constructions Inductives et de la théorie des types dans leurs aspects « langage de programmation »  
Le *Calcul des Constructions Inductives* est le nom du formalisme logique sur lequel repose Coq. Ce formalisme, qui dérive de la théorie des types de Martin-Löf, est à la fois un formalisme logique expressif (comparable en force à la théorie des ensembles) et un langage de programmation

### Contact

- **Responsable :** Pierre-louis Curien
- **Tél :** 01.44.27.53.05
- **Secrétariat Tél :** 01.39.63.57.89

### En savoir plus

- Site sur [inria.fr](http://inria.fr)
- Site du [responsable](#)
- Derniers Rapports d'Activité : [2015](#), [2016](#), [2017](#), [2018](#), [2019](#), [2020](#), [2021](#), [2022](#)

### Documents sur la structure

- [Intranet](#)
- [Privés](#)

### Décisions

- [6522](#) (13/01/2009) : création
- [7645](#) (15/12/2010) : prolongation

### Localisation

- **Adresse postale :** *Non renseignée*
- **Coordonnées GPS :** 48.83703, 2.103342

fonctionnelle fortement typé (type de formalisme que l'on range dans la famille des « théories des types »). Le Calcul des Constructions Inductives est un formalisme relativement jeune au sujet duquel plusieurs questions restent ouvertes. L'un des objectifs de  $\pi\lambda r^2$  est d'explorer quelques unes de ces questions, certaines étant d'un abord assez technique : traitement plus primitif des contraintes dites d'« inversion » lors des constructions par cas avec application au typage de la programmation par filtrage en présence de types « riches », prise en compte d'une forme syntaxique d'extensionnalité des fonctions (« éta-conversion ») avec applications à la conception d'algorithmes d'unification et d'inférence de type en présence de types riches.

- Développement du logiciel Coq, notamment dans ses aspects langage de programmation

Le logiciel Coq est un outil de développement semi-interactif de preuves dans une logique expressive bâtie autour d'un langage de programmation fonctionnelle fortement typé. Développé conjointement par plusieurs équipes INRIA et hors INRIA, Coq est utilisé tant pour la formalisation des mathématiques que pour la certification de propriétés de programmes. Naturellement doté de types « dépendants », Coq a une carte à jouer comme langage de programmation à type « riches ». Un des objectifs de  $\pi\lambda r^2$  dans ce cadre est le développement de bibliothèques de programmation certifiées, la certification du processus d'« extraction » de programme Coq vers des langages fonctionnels comme Objective Caml, ainsi que le développement de nouvelles méthodes de preuves (des « tactiques ») pour Coq. L'équipe  $\pi\lambda r^2$  développe aussi Pangolin qui est un outil spécifique de certification de programmes fonctionnels.

## Logiciels

- [Coq](#)
- [Pangolin](#)

## Relations industrielles et internationales

- ANR Déduction et Certification (DeCert)
- Action de Développement Technologique Coq ([ADT Coq](#))