

DESCRIPTION DU POSTE

Contexte

Les systèmes logiciels distribués de grande envergure (applications ou infrastructures) sont désormais omniprésents. Les systèmes basés sur des composants (par exemple, les architectures orientées services ou les microservices) offrent un moyen pratique de structurer les grands systèmes, en particulier les systèmes distribués déployés dans le Cloud, dans le cœur du réseau, ou à la périphérie (Edge) du réseau.

Cependant, les avantages des architectures distribuées s'accompagnent d'une complexité accrue et de défis techniques liés à l'observabilité, à la coordination, à la maintenance, etc. Un ensemble d'opérations appelées DevOps — c'est-à-dire les opérations gérées quelque part entre le développeur et l'administrateur de la machine — comprend notamment les configurations et reconfigurations du système qui sont nécessaires pour atteindre l'adaptabilité. Ces opérations englobent les changements initiaux et dynamiques qui peuvent survenir sur une architecture logicielle distribuée orientée services : ajouter ou supprimer des services, connecter ou déconnecter des services, et modifier certains paramètres ou le comportement interne des services. Elles peuvent être requises pour gérer divers types de scénarios dynamiques tels que la tolérance aux pannes, l'évolutivité (scalability), les mises à jour logicielles, diverses optimisations, etc.

De tels changements peuvent entraîner des défaillances. Une étude portant sur 597 pannes imprévues qui ont affecté des services Cloud populaires entre 2009 et 2015 a révélé que 16% d'entre elles étaient causées par une mise à niveau logicielle ou matérielle. L'étude conclut que « la complexité de l'écosystème matériel et logiciel du Cloud a dépassé les outils existants de test, de débogage et de vérification ». En effet, les méthodes de test et de débogage sont largement inadéquates dans le contexte des systèmes distribués, tandis que l'adoption de méthodes formelles plus appropriées reste marginale dans l'industrie. Ceci peut être attribué à la difficulté d'utiliser les méthodes et outils formels. Pourtant, les méthodes formelles peuvent alléger la charge des ingénieurs DevOps au lieu de l'alourdir.

Nous pouvons distinguer deux types d'approches DevOps : premièrement, les langages impératifs où l'utilisateur écrit un programme qui prescrit explicitement l'ensemble des actions à effectuer ; deuxièmement, les langages déclaratifs où l'utilisateur déclare la configuration souhaitée, et où l'outil en déduit ou compile le(s) programme(s) associé(s).

L'objectif du projet ANR FOR-COALA est double :

- Comprendre et combler le fossé entre un outil populaire de la communauté DevOps et un outil issu du milieu universitaire.
- Améliorer la compréhension de ces langages en se basant sur une sémantique formelle mécanisée et développer des transformations inter-langages vérifiées qui préservent la sémantique.

Les résultats du projet seront deux langages de configuration formellement certifiés, en open source, leur moteur d'exécution, et un outil de traduction vérifié.

Présentation

Depuis 2015, l'objectif général de l'équipe Languages, Modeling, Verification (LMV) est de faire progresser la fiabilité et la sécurité des logiciels, en particulier, mais pas exclusivement, dans le contexte de l'Internet of Things (IoT). Cet objectif s'inscrit dans le domaine général de la cybersécurité. Nous visons à garantir que le

logiciel impliqué satisfasse des propriétés critiques, soit par construction en tirant parti de la conception de bibliothèques et de langages de programmation, soit en utilisant des méthodes formelles.

Les programmes que nous considérons sont des logiciels embarqués, qu'il s'agisse d'applications, de composants de systèmes d'exploitation, de modules critiques pour la sécurité, de logiciels traitant du déploiement, de la configuration et de la reconfiguration d'applications distribuées sur l'Internet des Objets, ainsi que de programmes parallèles pour l'analyse de données massives provenant de l'IoT. Nous considérons des propriétés locales aux fonctions (contrats de fonctions) mais aussi des propriétés temporelles et d'autres propriétés de haut niveau.

Nous nous appuyons sur plusieurs branches des méthodes formelles, notamment la vérification déductive, les fondements théoriques de la sémantique des langages de programmation, mais aussi le typage, les systèmes de réécriture, l'analyse statique et la combinaison de techniques. Dans la plupart de nos travaux, nous utilisons l'assistant de preuve Coq pour générer du code certifié lorsque cela est pertinent ou, du moins, pour renforcer la confiance dans les résultats obtenus, ou pour compléter l'utilisation d'autres méthodes formelles.

L'équipe LMV fait partie du Laboratoire d'Informatique Fondamentale d'Orléans (LIFO) (<https://www.univ-orleans.fr/lifo/>).

Mission

L'ingénieur(e) contribuera à la conception et au développement de la sémantique formelle mécanisée, en utilisant notamment l'assistant de preuve Coq, bien qu'un travail avec l'environnement de preuve Why3 et le système Maude puisse être nécessaire.

L'ingénieur(e) fera partie de l'équipe For-CoaLa qui est actuellement composée d'un professeur et de deux maîtres de conférences, ainsi que d'un doctorant, à l'Université d'Orléans et à l'IMT Atlantique (Nantes)

PROFIL RECHERCHÉ

Formation et Compétences requises

Formation de niveau Licence (ou équivalent) en informatique

Compétences en méthodes formelles, en particulier la preuve de théorème interactive avec Coq

Familiarité avec au moins un outil d'Infrastructure-as-Code

Excellent niveau en Anglais (lecture et écriture)

Cours de logique, de programmation fonctionnelle et de méthodes formelles réussis avec mentions positives

Stage de recherche en méthodes formelles et/ou Iaas (Infrastructure as Code), avec rapport et artefact logiciel

L'UNIVERSITÉ D'ORLÉANS C'EST AUSSI :

Une université européenne, des valeurs partagées, un plan
égalité professionnelle femme/homme, diversité et non-
discrimination, inclusion, qualité de vie au travail et
engagement dans la transition écologique

Un cadre de travail respectueux de l'équilibre vie professionnelle et vie personnelle :

- Développement de ses compétences à travers des formations riches et variées
- Accompagnement des personnes en situation de handicap et adaptation des postes de travail
- 57 jours de congés annuels - dont 10 jours de RTT - pour un temps plein actif dès la prise de poste
- Télétravail possible selon nécessité et organisation du service (1 à 2 jours/ semaine)

Et des avantages ...

- Activités sportives et culturelles
- Restauration collective
- Forfait mobilité durable (vélo, covoiturage)
- Prise en charge partielle des transports
- Aides et prestations d'actions sociales : aide à la famille, aux vacances et séjours pour enfants,
- Aides aux parents d'enfants en situation d'handicap

Rejoignez-nous !

MODALITÉS ET PROCESS DE RECRUTEMENT

Lieu d'exercice : Université d'Orléans, Laboratoire d'Informatique Fondamentale d'Orléans (LIFO)

Prise de poste : Poste à pourvoir à partir de novembre 2025

Type de contrat/Statut : Catégorie A (IGE), Statut contractuel (CDD), temps incomplet (25%)

Adressez votre lettre de motivation et votre C.V. **jusqu'au 12/10/2025**

Lien de candidature : <https://jobaffinity.fr/apply/1vxs9bjhttcv4nse64>