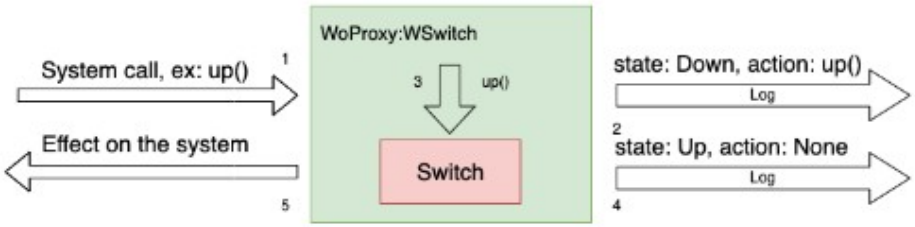
LISTIC

Offre de stage 2023-2024

Titre	**Virtualisation automatisée d'objet logiciel**
Niveau du stage	M2 M2 recherche Ingénieur année 3
Date de début et durée	Février/mars, 4 à 6 mois
Ville, Pays	Annecy, *France*
Laboratoire	LISTIC - Laboratoire d'Informatique, Systèmes, Traitement de l'Information et de la Connaissance http://www.polytech.univ-savoie.fr/LISTIC
Description du sujet de stage	**Contexte :** L"Objets Sage" ou Wise Object est un concept visant à rendre n'importe quel objet logiciel intelligent. Grâce à un proxy logiciel dynamique, le Wise Object intercepte tous les messages qui sont destinés à l'objet original, le rendant ainsi "Sage" (il se connaît lui-même). Cette interception a plusieurs buts : • **Monitorer** sans intrusion les états de l'objet afin de les analyser. • **Modifier** l'appel original si l'analyse révèle que l'appel entrant va mettre l'objet dans un état incohérent. • **Bloquer** l'appel si l'analyse révèle que l'appel entrant va mettre l'objet dans un état incohérent.  Voici un exemple du fonctionnement du Wise Object. Ici, le Wise Object crée un proxy sur l'objet métier Switch. Cela lui permet d'intercepter l'appelle système up() (1). A chaque changement d'état, le WO va envoyer des logs dans une mémoire. Le log (2) étant l'état de l'objet avant appel : down ainsi que la méthode appelée : up(). Il va ensuite demander à l'objet d'effectuer ce pour quoi il a été appelé pour voir ce qu'il se passe (3). Puisque l'objet change d'état, un nouveau log est généré (4). Enfin on retourne ou non le résultat de la méthode sur le système (5). On note que si une anomalie ou une erreur sont détectées (par analyse), ce dernier call peut être annulé ou modifié. Les logs seront ensuite analysés pour en apprendre plus sur l'objet ou détecter d'éventuelles anomalies. Cela constitue ce que l'on appelle un système auto-adaptatif. Le système s'analyse et se corrige lui-même si on détecte que quelque chose ne va pas. La philosophie WO permet de rendre n'importe quel objet logiciel sage avec très peu d'intrusion dans le code de l'application. **Objectif du stage :** Ce qui différencie les WO des autres systèmes auto-adaptatifs est sa capacité à rêver. Lorsque l'objet n'est pas occupé par une demande métier, il va entrer dans un état de rêve. Durant le rêve, il va effectuer des appels à ses propres

	méthodes afin de changer d'état. Ces appels n'ont aucun impact sur le monde extérieur puisqu'on ne renvoie pas les effets sur le système (5). Dès qu'un appel extérieur est reçu, l'objet se réveille, puis il se remet dans son état initial (avant le rêve) dans le but de ne pas impacter le système avec un état qu'il n'avait pas prévu. Cela nous permet d'en apprendre plus sur les différentes façons dont l'objet peut fonctionner (le système peut ne pas utiliser toutes les méthodes de l'objet, ou les utiliser avec uniquement les mêmes données). Nous avons cependant un problème pour le moment : l'objet peut accéder et modifier une donnée extérieure (fichier, bdd, etc.). Cela pose problème car cela modifierait la façon dont l'objet éveillé ou le système réel pourrait se comporter. Il faut que le système soit capable de modifier ces données extérieures pendant le rêve mais que la version pré-rêve soit restaurée dès que l'objet est appelé par le système (éveil). Le but de ce stage consiste à trouver une solution à ce problème. Pour l'instant les pistes qui ont été envisagées sont : conteneurisation avec docker, virtualisation avec des VM, modification importante dans le code du Framework (interception des IO, bouchons, etc.). **Travail demandé / livrable :** • Rédaction d'un état de l'art sur les différentes solutions possible • Prise en main du Framework WOF en Java • Modélisation et implémentation de solution de virtualisation • Contribution au Framework Java • Déploiement d'une architecture complète **Mots clés :** Modélisation logicielle, dev. Python/Java, virtualisation, IA, sagesse, apprentissage. **Candidature :** La candidature se fait via une première prise de contact par mail en nous fournissant un CV, une lettre de motivation et vos relevés de notes post-bac (licence et master). La sélection des candidats se fera sur l'excellence des dossiers en vue d'une candidature en thèse sur bourse au mérite. **Références :** • I. Alloui, F. Vernier, A Wise Object Framework for Distributed Intelligent Adaptive Systems, ICSOFT 2017. • I. Alloui, E. Benoit, S. Perrin, F. Vernier, WIOT: Interconnection between Wise Object and IOT, ICSOFT 2018. • S. Lejamble, I. Alloui, S. Monnet, F. Vernier, A New Software Architecture for the Wise Object Framework: Multidimensional Separation of Concerns.,ICSOFT 2022
Compétences requises	Conception et programmation Java et Python
Gratification	**4.05** euros/heure de travail
Tuteurs / Contacts	Hervé Verjus, Flavien Vernier Téléphone : 04 50 09 65 94 / 04 50 09 65 90 E-mail : {herve.verjus, flavien.vernier}@univ-smb.fr