

Comptes-rendus

Romain PRETET¹ and Camille VEDANI²

1 Université III Paul Sabatier, France, romain.pretet@univ-tlse3.fr

2 Université III Paul Sabatier, France, camille.vedani@univ-tlse3.fr



Résumé

Compte rendu des réunions du TER 2025. Lien du projet :
<https://fr.overleaf.com/3768471444ktsnghtzntdj#0fe84e>

1 Le 15/05/25

Classe ELM dans Ottawa

- **Localisation** : Dans les documents sur le site.
- **Fichiers** : `.elf` (exécutables pour différentes machines, format binaire).

Outils Graphviz

- Intégration directe avec Ottawa, utilisé dans XDD.
- Utilisation recommandée avec `xdot` (nécessite installation).
- À éviter pour les gros fichiers.

Fonctionnement d'Ottawa

- **Classe First** : Gère la classe application.
- **Fonction work** : Exécutée par le programme lors de l'utilisation.
- **Objets** : Chaque objet est une liste de propriétés, permettant d'accrocher des propriétés pour des analyses successives.
- **Parcours de CFG** : Utilisation de marqueurs pour éviter les boucles infinies.
 - Recommandation : Bien refaire la version 2.4 et ajouter le marquage dans le programme `.cpp`.
- **CFG Collection** : Ottawa fournit les CFG, pas besoin de les créer manuellement.
 - Un programme est une collection de CFG.

Objectifs

- Considérer les XDD comme des fonctions.



© Romain PRETET, Camille VEDANI;
sous licence Creative Commons CC-BY

Hashtable et Multithreading

- **Hashtable** : Utilisable en multithread.
- **XDD** : Stocké dans une hashtable partagée.
- **Politique de Partage** : Nécessité de développer une politique de partage pour éviter les conflits.
- **NodeManager** : Liste des XDD retenus.
- **OpsManager** : Enregistre les opérations déjà calculées.

Recommandations Techniques

- Reprendre la classe **HashMap** ou une surclasse gérant les accès parallèles.
- Mettre à jour **Ops** et **Node** pour assurer le bon fonctionnement.
- Commande à mettre dans **bashrc** : `export PATH=$PATH:~$HOME/otawa/bin`