Retour de Philippe sur la revue de mi-parcours à l'ANR

- ANR positif => content
- pas de publication commune
- réfléchir à la valorisation industrielle

Bilan de ce qui a été fait ou est en cours :

- Cubicle avec des canaux
- Sémantique de BPMN des workflow codée en TLA+ :
  - soucis c'est la sémantique non paramétrée
  - discussion sur la ré-entrance et les multi-instances
  - sémantique à finir de discuter : visite de Sarah
  - encodage proche des réseau de Pétri -> se code aisément en TLA+
  - discussion sur l'atomicité des boites
  - traduction en Cubicle ?
  - exemple avec des soucis de sécurité pour avoir des soucis de safety
  - souci : dans la modélisation : le temps est abstrait avec du non-déterministe (pas réalise pour certain workflow) -> Lien avec les automates temporisés paramétrés
- Travaux d'Adam (un peu en périphérie) : prédicats HO (OPODIS)
- Non bloquant (un peu en périphérie)

interaction model checker / prover

- Cubicle -> WhyML : lance l'inférence d'invariant cubicle et les traduit en WhyML
- splitter en théorie des ensembles à la main
- model checker symbolique pour TLA+ de Igor Konnov : en cours de développement, pour l'instant pas paramétré (chaque chose en son temps)
- Mattias : lien Cubicle - Why3
- plus de structures de donnée dans Cubicle
- arithmétique dans Cubicle
- génération d'invariant dans Cubicle

Présentation de Mattias : "Cubicle rencontre Why3"

- Exemple du splitter
- Discussions sur les aller - retour Cubicle - Why3
- Invariant avec les canaux dans Cubicle ?

Parser Cubicle -> TLA+

Workflow

- Fini de coder la sémantique

- Traducteur workflow > TLA+
- format d'entrée : JSON - étudiants qui travaille sur passage BPMN -> JSON
- finir la partie communication
- quand Sarah est là : insérer le framework
- travailler sur le multi-instance
- dans la sémantique : des simplifications ont été faites car on savait qu'il n'y avait pas de multi-instance
- a priori dans le multi-instance, les instances ne communiquent pas ensembles -> peut-être une extension qu'on aura besoin de faire
- des instances qui partagent une ressource critique : cette ressource est un processus et ils communiquent avec.
- Barrière de synchronisation : marqueur spécifique (envoi en parallèle ou en séquence) et idem au niveau de la réception.
- Il reste des aspect d'ambiguïté quand il y a de la création dynamique d'instances
- multi-instance : pas la partie la plus formalisée dans la norme
- Il y a des points qui sont flous -> les moteurs font ce qu'ils veulent. Vision de Pascal, introduire autant d'annotations que de sémantiques possibles pour lever les ambiguïtés et ne pas faire de choix de sémantique par défaut.
- Laisser la possibilité aux utilisateurs de choisir le comportement (annotation) ou ne pas fixer et laisser le prouveur faire cela (annotation spéciale pour dire "je ne le donne pas").
- exemple le celui de la norme avec le Prix Nobel - peu d'exemple dans la norme => on va pouvoir faire un peu ce qu'on veut...
- changement dynamique de type d'un processus -> a priori non
- on va avoir besoin de traiter des données (un peu à la SDL) : possibilité de manipuler les identifiants des processus.
- bibliothèque de bench : souvent un seul processus (petits exemples industriels) - Sarah a réussi à récupérer un autre dépôt mais plein de formats différents (un peu de tout et n'importe quoi... même les étudiants pouvaient rajouter des choses)
- idée de Pascal : modèle de communication minimal
  - les modèles d'ordre sont sur les boites aux lettres et pas sur les canaux
  - notion d'erreur en BPMN : blocage de la communication
  - donc faudra s'intéresser à la question de quel est le bon modèle pour que ça marche.

- erreur : safety? -> a priori non
  - pouvoir terminer
  - terminer sans jeton qui traînent
  - toutes les tâches peuvent être activées
  - pas de raison qu'il n'y ai pas de safety, mais va regarder s'il a des exemples avec ça
- relancer l'entreprise (W4??) ils ont des modèles BPMN - d'ici la fin de l'année voir d'ici la fin de mois : ils développent un moteur BPMN, participent aux organismes de normalisation... Intéressé par la vérification et le paramétré. Soucis : on n'a pas le temps! (Sarah a une partie dans sa thèse sur les automates centralisés, il faudrait faire la liaison dans la thèse de Sarah via le temps).
- aller moins loin sur le paramétré pour mettre du temps (timeout). Sortie vers UPAL? Cubicle : pas du tout de temporisé. Temps discret avec des tick : se traduit sans souci.
- le soucis c'est quand les process sont synchronisés par le temps et pas par les messages. Sinon c'est facile on fait des transitions non déterministes sur des transitions de timeout. Qu'est ce qu'il y a vraiment dans les exemples? A priori, rien de compliqué...
- BPMN utilisé pour décrire un process d'installation d'un firmware avec un trou de sécurité : security of SoC firmware load protocol.