

TP 02 : Création d'un outil de Transformation des BPMN Vers Rdp

L'objectif de ce TP est le développement d'un outil graphique pour la transformation des modèles BPMN vers les Réseaux de Petri (RDP) basé sur la transformation de graphes. Ce TP contient trois étapes :

- 1- Méta-modélisation des modèles source **BPMN** et cible **Rdp**
- 2- Proposition des Règles de la transformation (grammaire de graphe).
- 3- Exécution de la transformation sur un modèle BPMN.

1- Méta-modélisation des modèles BPMN :

- **BPMN.model**: c'est la syntaxe abstraite du modèle BPMN.

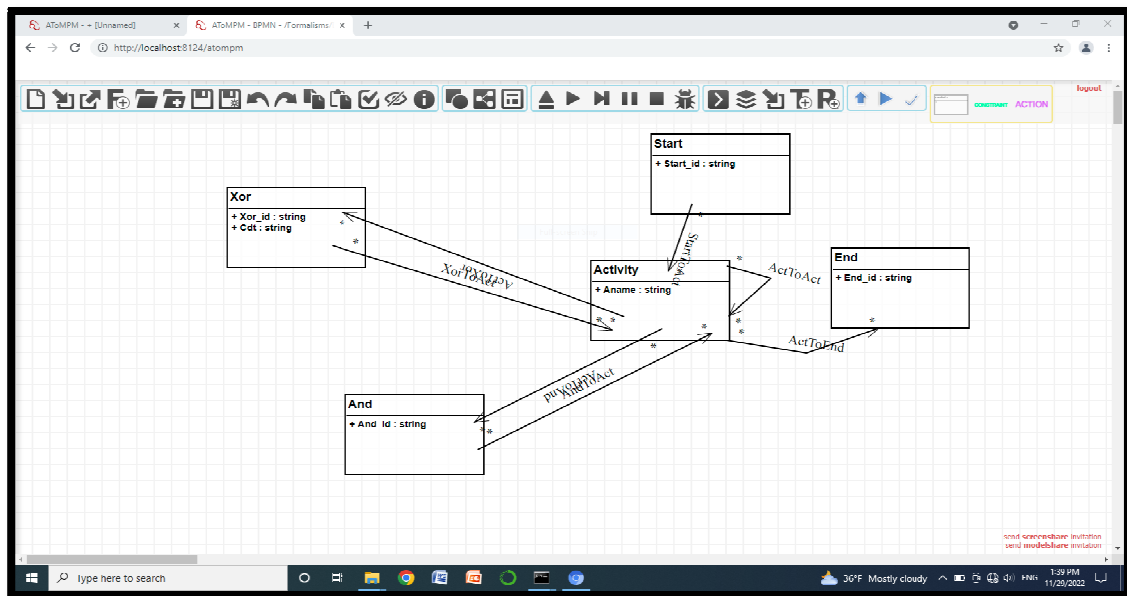


Figure1 : syntaxe abstraite du modèle BPMN.

- **BPMN.defaultIcons.model** : c'est la syntaxe concrète du modèle BPMN.

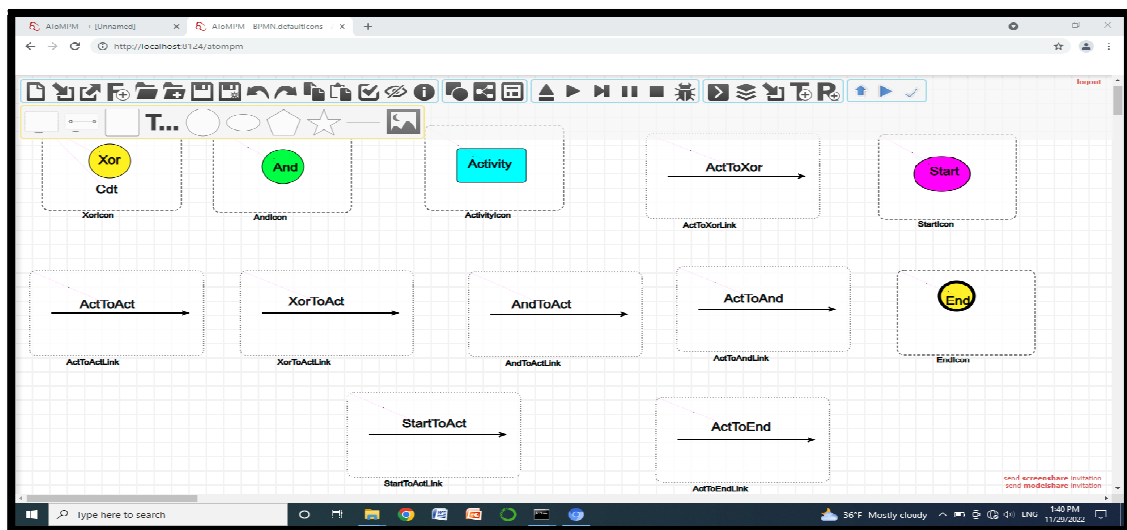


Figure2 : syntaxe concrète du modèle BPMN.

2- Règles de la transformation BPMN vers RdP :

Dans cette partie, vous utilisez la barre d'outils *TransformationEditor* pour créer une règle ou un processus de transformation. L'idée de la transformation BPMN vers RdP composée de trois catégories :

- **Les règles de création (5 règles)**
 - ❖ Les activités sont transformées en des transitions.
 - ❖ les connecteurs XOR sont transformés en des places.
 - ❖ les connecteurs AND sont transformés en des places.
 - ❖ Start et End sont transformés en des places.
- **Les règles de liaison (7 règles)**
- **Les règles de suppression (5 règles) : Supprimer** les activités, les connecteurs XOR, les connecteurs AND, Start et End.

Afin d'implémenter cette transformation vous utilisez les **17 règles** de transformation

2-1- Les règles de création

Activité vers transition

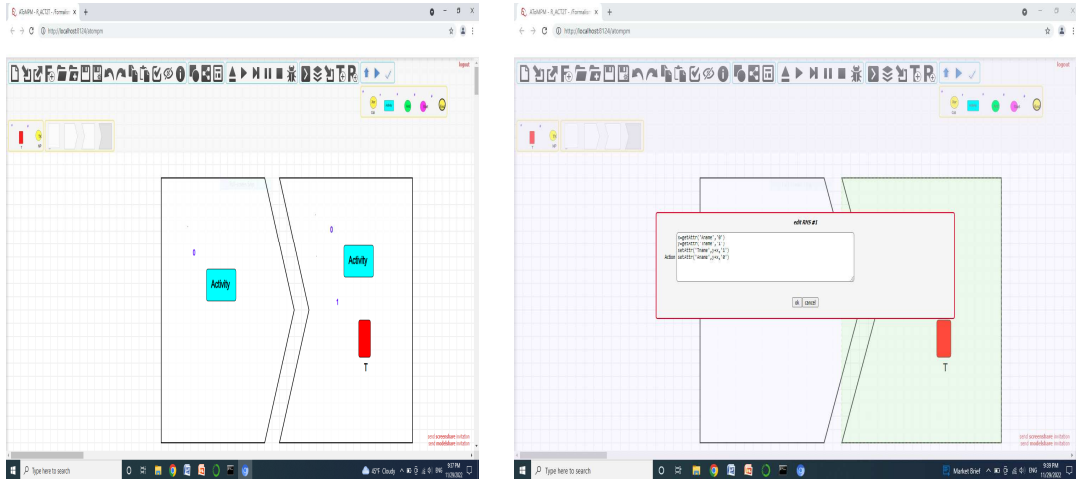


Figure3 : Activité vers transition

Start

* END

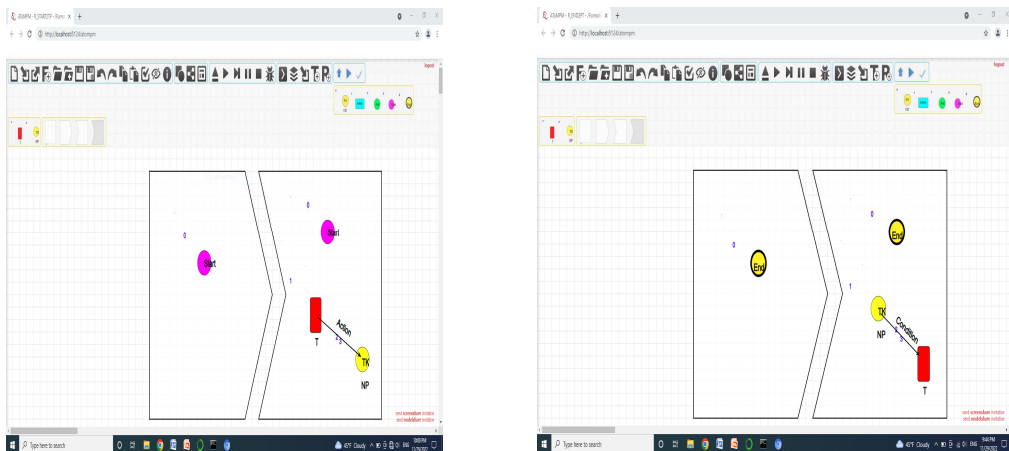


Figure4 : Start et End

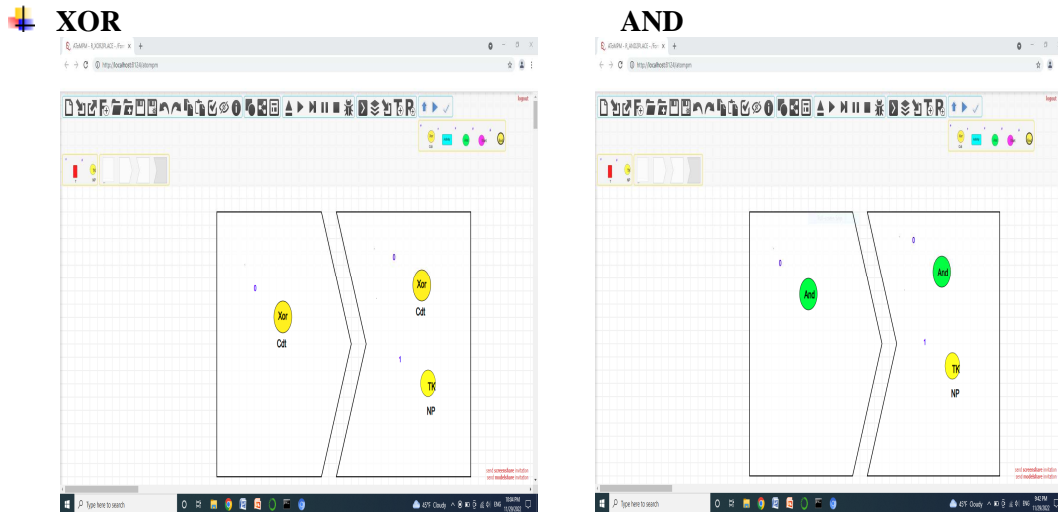


Figure5 : XOR et AND

2-2- Les règles de liaison

A- Act2Act

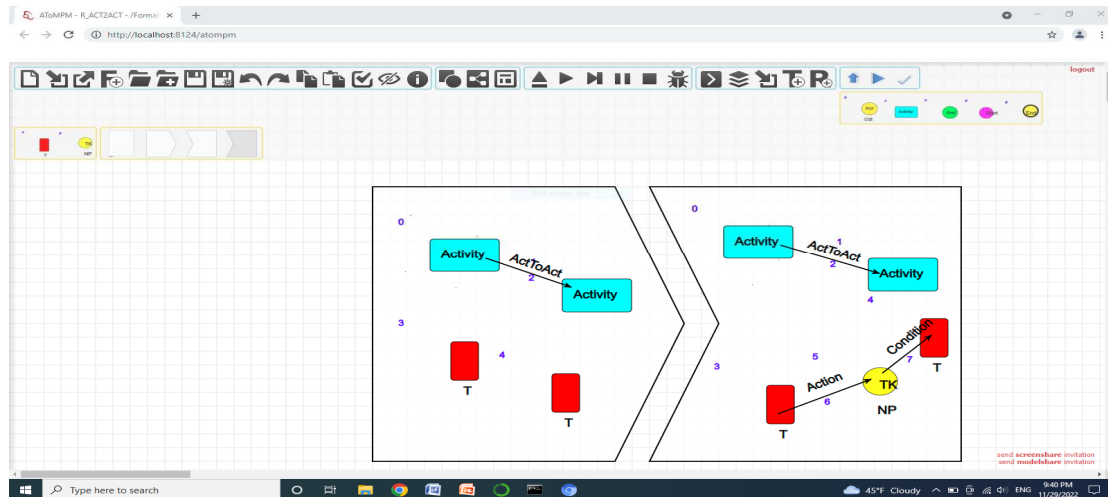


Figure06 : Règle Act2Act

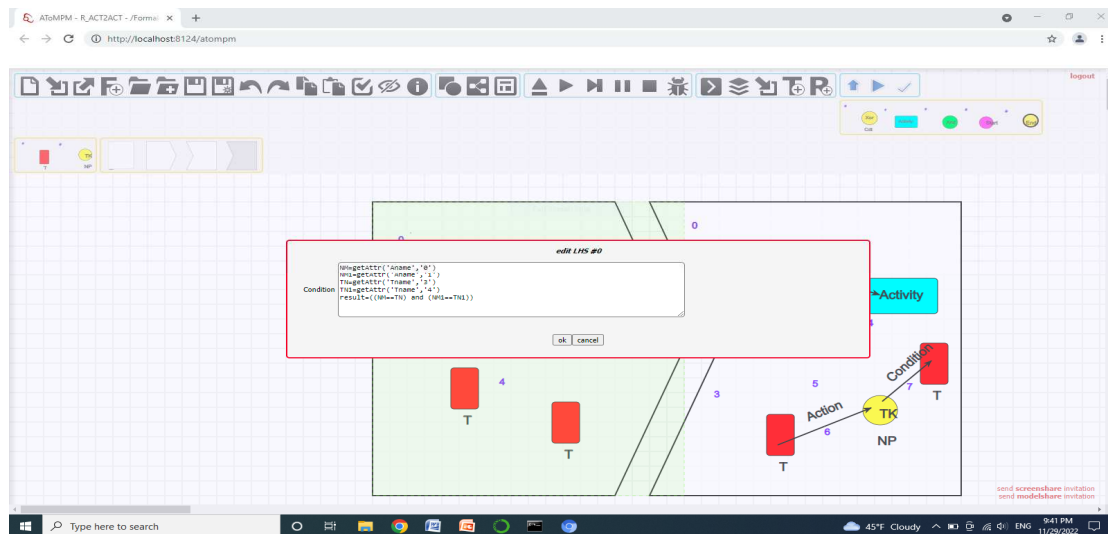


Figure07 : La partie condition de la règle Act2Act

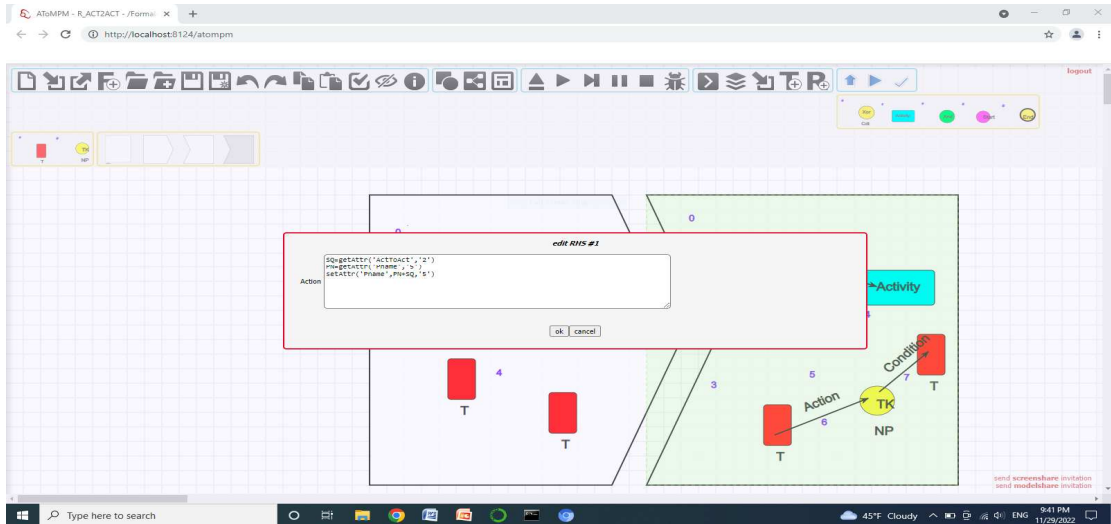


Figure08 : La partie action de la règle Act2Act

B- Act2And et And2Act

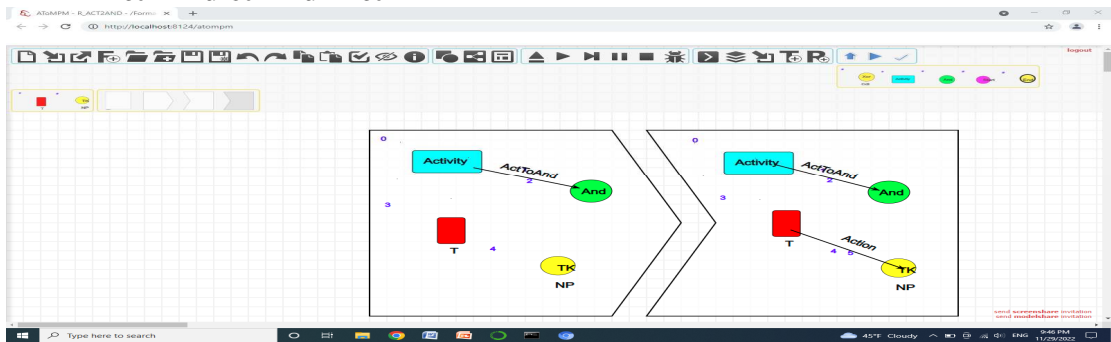


Figure09 : Règle Act2And

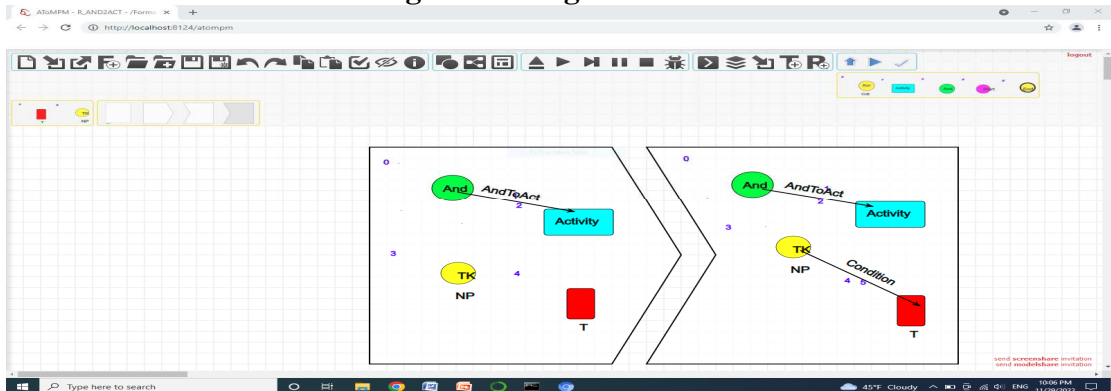


Figure10 : Règle And2Act

C- Act2Xor et Xor2Act

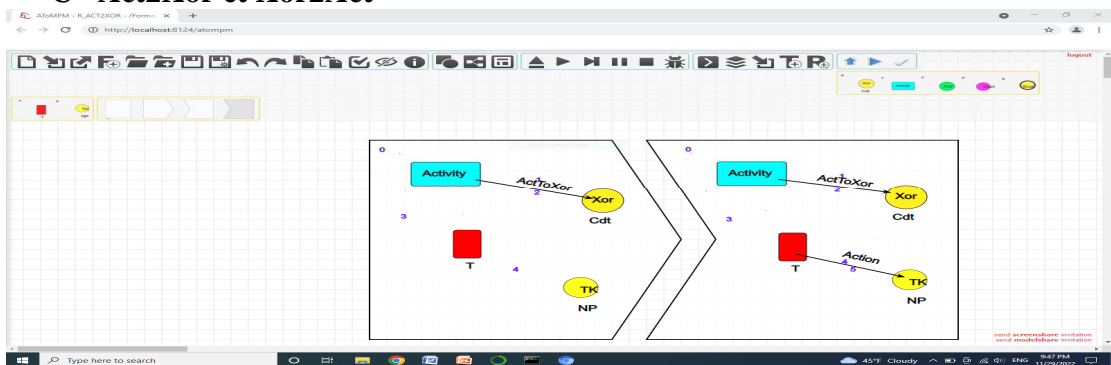


Figure11 : Règle Act2Xor

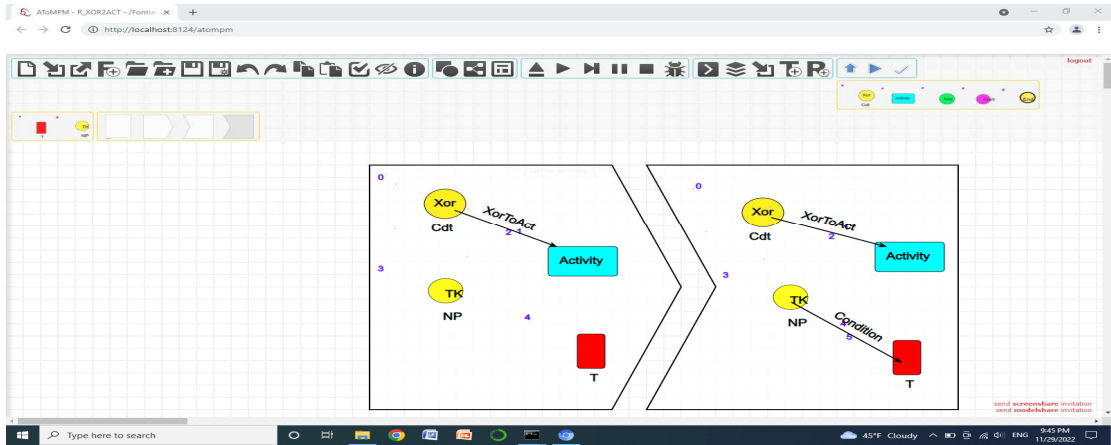


Figure12 : Règle Xor2Act

D- Start2Act et Act2End

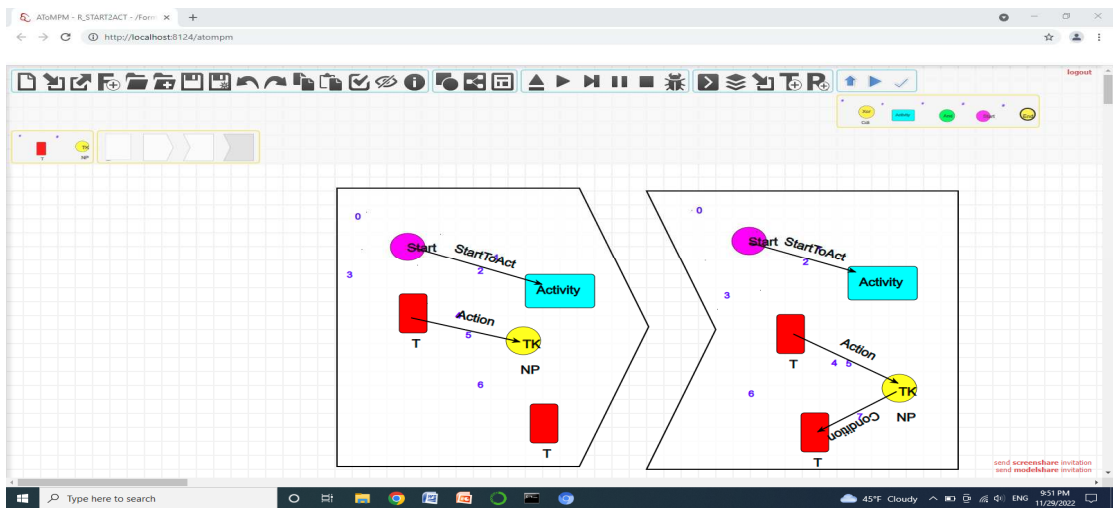


Figure13 : Règle Start2Act

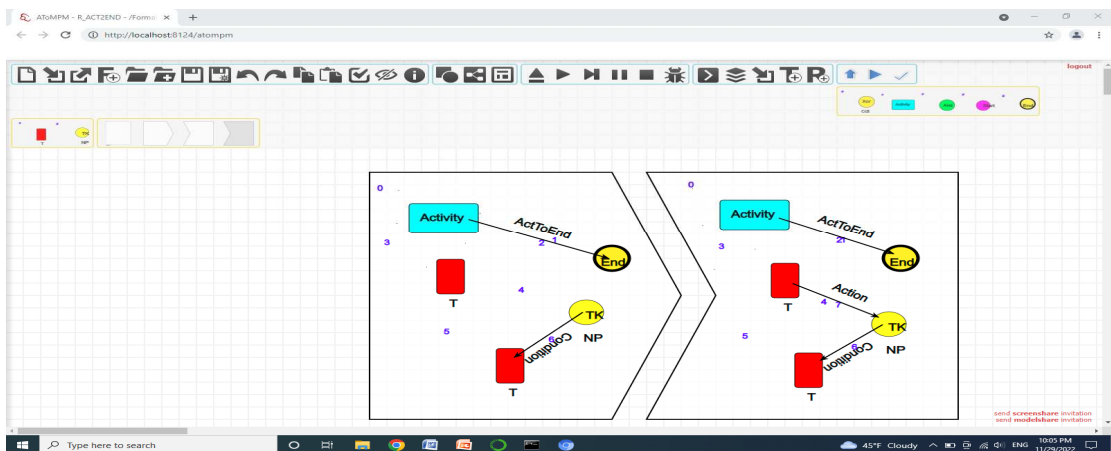


Figure14 : Règle Act2End.

2-3- Les règles de suppression

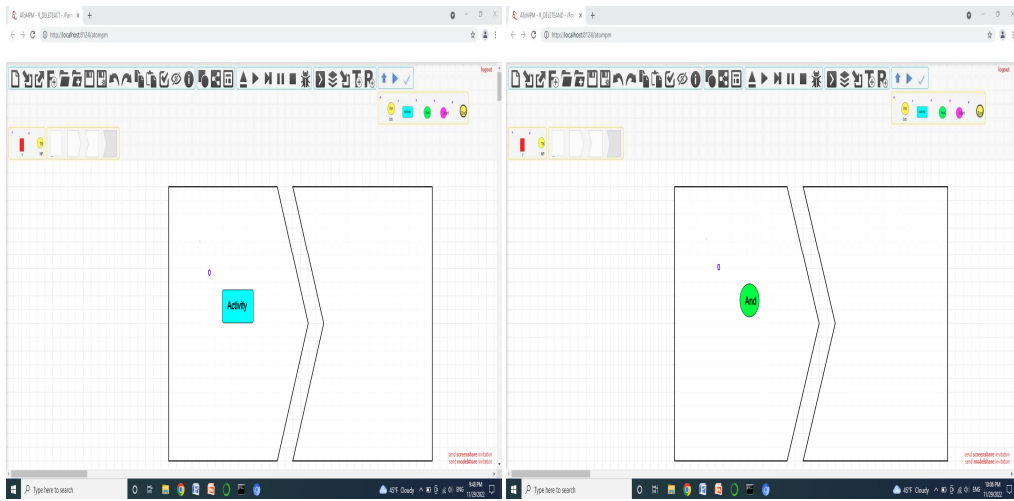


Figure15 : Règle supprimer Activité et And

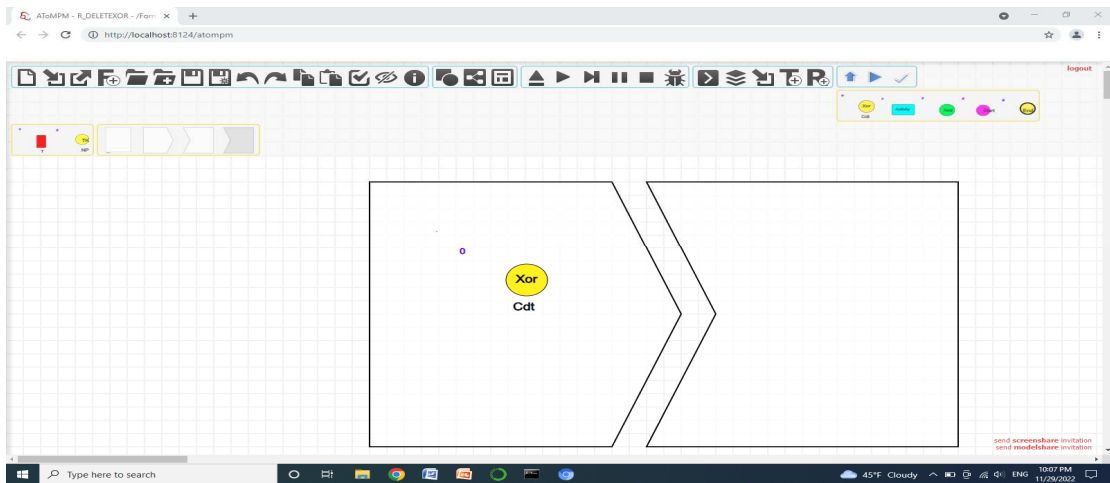


Figure16 : Règle supprimer XOR

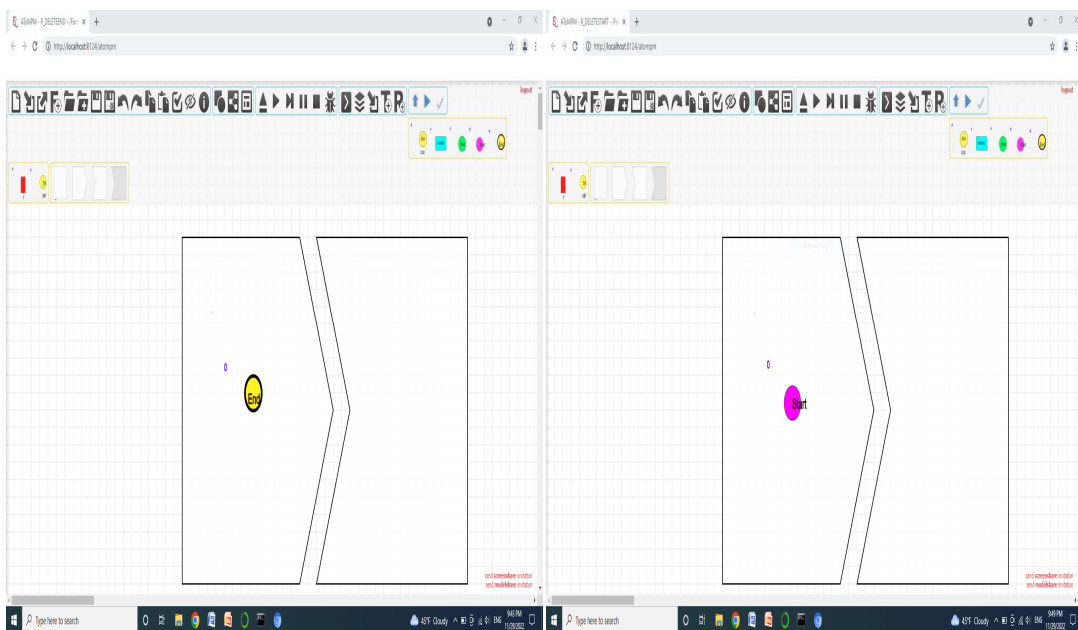


Figure17 : Règle supprimer End et Start

3- **Le langage d'ordonnement MoTif** : Les règles sont planifiées à l'aide d'un langage de planification (**MoTif**). Une planification se compose d'un certain nombre d'étapes commençant à la pseudo-étape *Start* et se terminant par une pseudo-étape *EndSuccess* ou *EndFail*. Chaque étape est reliée à deux autres étapes avec un lien de réussite ou d'échec. Selon le type d'étape, la ou les règles référencées sont exécutées d'une manière particulière.

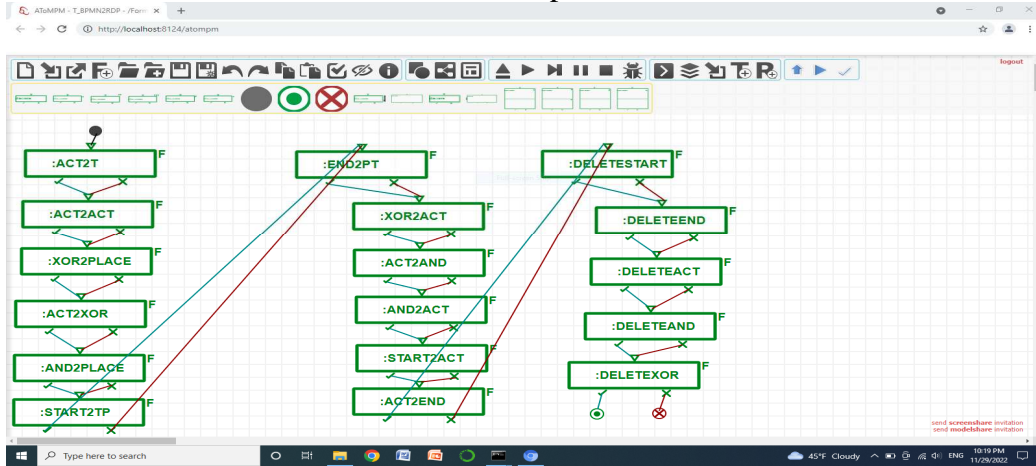


Figure18 : MoTif de planification des règles.

4- **Exécution de la transformation sur un modèle BPMN** :

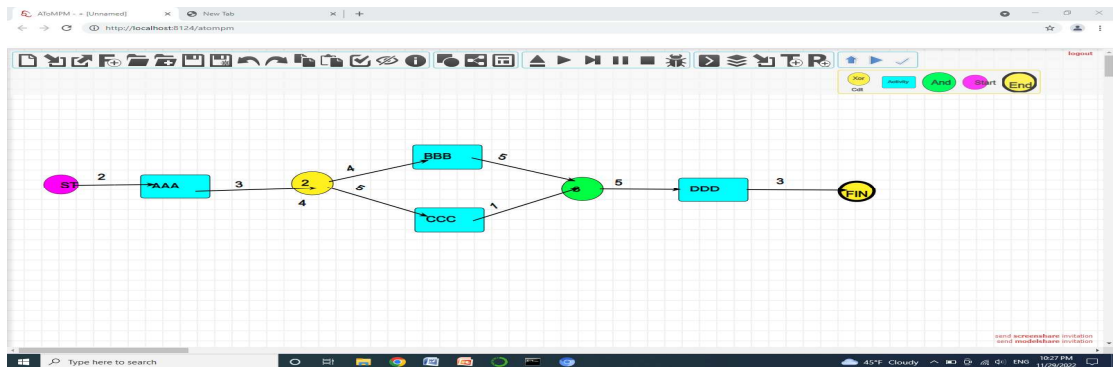


Figure19 : Exemple d'un BPMN.

✓ Utilisez la barre d'outils *TransformationController* pour exécuter la transformation *BPMN2RdP*.

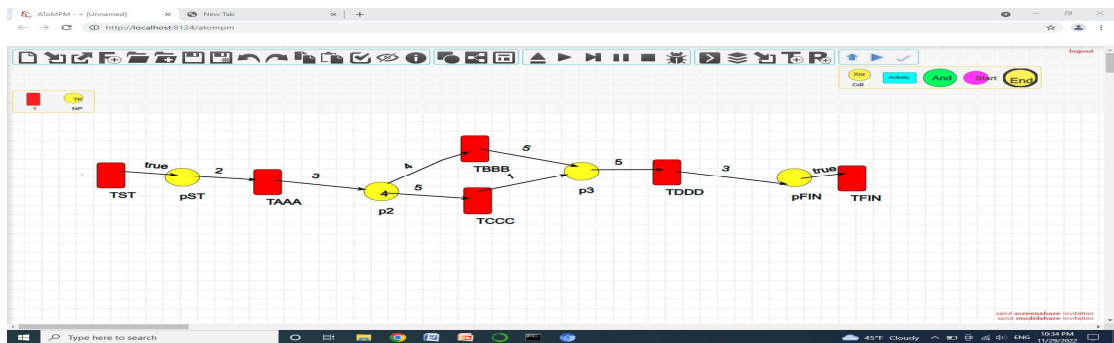


Figure20 : Résultat de la transformation des BPMN vers les RDP.