

# Rapport du projet de Base de Données I

DE COOMAN Thibaut      STAQUET Gaëtan

19 décembre 2016

## Table des matières

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>2</b>	<b>Utilisation</b>	<b>2</b>
<b>3</b>	<b>Syntaxe</b>	<b>2</b>
3.1	SPJRUD . . . . .	2
3.2	Schémas . . . . .	2
<b>4</b>	<b>Construction de l'AST</b>	<b>3</b>
4.1	Décomposition . . . . .	3
4.2	Construction . . . . .	3
<b>5</b>	<b>Vérification de l'AST</b>	<b>3</b>
<b>6</b>	<b>Traduction de l'AST</b>	<b>3</b>

## 1 Introduction

Comme conseillé dans l'énoncé, nous avons employé le principe de l'AST pour implémenter ce vérificateur et traducteur.

Dans ce rapport, nous parlerons de l'utilisation de l'application, de la syntaxe à employer pour rentrer les commandes, de la façon dont l'AST est construit, de sa vérification et de sa traduction en SQL, ainsi que de la communication avec la base de données.

Nous supposons que nous travaillons dans la base de données suivante :

Name	Country	Population
Bruxelles	Belgique	184230
Paris	France	123456789

## 2 Utilisation

L'application permet de travailler sur une base de données déjà existante ou sur un schéma que l'utilisateur doit entrer. Une fois ceci fait, l'utilisateur peut entrer une requête SPJRUD. L'application vérifie cette requête et la traduit en SQL. Si l'utilisateur a demandé à employer une base de données, l'application exécute (via SQLite) la requête SQL et affiche le résultat. Dans tous les cas, l'application demande à l'utilisateur d'entrer une nouvelle requête (ou une ligne vide pour quitter l'application).

## 3 Syntaxe

### 3.1 SPJRUD

Les requêtes SPJRUD doivent respecter la syntaxe décrite dans la table suivante.  $E$  indique une relation algébrique et  $E'$  indique cette relation écrite dans la syntaxe demandée par l'application.

SPJRUD	Application
Relation R	Rel("R")
$\sigma_{A=a'}(E)$	Select(Eq("A", Cst('a')), $E'$ )
$\sigma_{A=B}(E)$	Select(Eq("A", Col("B")), $E'$ )
$\pi_X(E)$	Proj(["X <sub>1</sub> ", "X <sub>2</sub> ", "X <sub>3</sub> ", ..., "X <sub>n</sub> "], $E'$ )
$E_1 \bowtie E_2$	Join( $E'_1$ , $E'_2$ )
$\rho_{A \rightarrow C}(E)$	Rename("A", "C", $E'$ )
$E_1 \cup E_2$	Union( $E'_1$ , $E'_2$ )
$E_1 - E_2$	Diff( $E'_1$ , $E'_2$ )

### 3.2 Schémas

La syntaxe à employer pour définir des schémas de relation est la suivante :

"Nom de la relation", ("Nom de la colonne 1", "Type de la colonne (SQL types)", "Si la colonne peut contenir la valeur NULL ou non"), ("Nom de la colonne 2", ...), ...

Par exemple,

"Notes", ("Nom", "VARCHAR(25)", False), ("Points", "INTEGER", False)

définirait la table suivante :

Nom	Points
-----	--------

Nous n'avons pas permis de populer une table définie de cette façon.

## 4 Construction de l'AST

Admettons que l'utilisateur veuille traduire la requête en SPJRUD suivante :

**Requête SPJRUD 1.**  $\rho_{Name \rightarrow City}(\pi_{Name}(\sigma_{Country='France'}(Cities) \cup \sigma_{Country='Belgique'}(Cities)))$

Cette requête devrait être encodée comme :

**Requête 1.** *Rename("Name", "City", Proj(["Name"], Union(Select(Eq("Country", Cst("France")), Rel("Cities")), Select(Eq("Country", Cst("Belgique")), Rel("Cities")))))*

Pour pouvoir construire l'arbre correspondant à cette requête, nous avons décidé de procéder comme suivant :

1. Vérifier les parenthèses et crochets
2. Décomposer la requête
3. Construire l'arbre noeud par noeud

La vérification des parenthèses et crochets est suffisamment simple pour ne pas être expliqué ici.

### 4.1 Décomposition

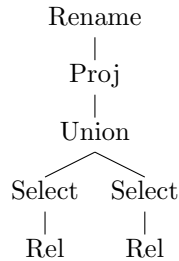
Cette étape crée une liste de listes et/ou de chaînes de caractères. Par exemple, *Select(Eq("A", Cst(a)), Rel("R"))* donne la décomposition *["Select", ["Eq", ["A", "Cst", ["a"]], "Rel", ["R"]]]*

La requête 1 donnerait ainsi la décomposition :

```
["Rename", ["Name", "City", "Proj", [{"Name"},
"Union", ["Select", ["Eq", ["Country", "Cst", ["France"]],
"Rel", ["Cities"]], "Select", ["Eq", ["Country", "Cst",
["Belgique"]], "Rel", ["Cities"]]]]]]
```

### 4.2 Construction

De cette liste, nous pouvons construire l'arbre suivant :



## 5 Vérification de l'AST

## 6 Traduction de l'AST