

Larged Scaled Distributed System

Johnny Nguyen¹

Abstract—This electronic document permits me to synthesis the third course of Larged Scaled Distributing System from the fourth week.

I. INTRODUCTION

Il y a un pattern général qui équivaut à Single Processus Multiple Data. Qu'est-ce qu'un Workflow ? La représentation concrète contient une multitude de processus et pour chaque processus nous avons une seule évaluation. Le runtime est bien géré pour distribuer les ressources aux différents utilisateurs. Le workflow du manager consiste à avoir un workflow abstrait, qui donne un workflow concret, ce dernier nous donne une liste de tâche utilisant les ressources pour exécuter les tâches qui seront tracées.

II. WORKFLOW REPRESENTATIONS

Un workflow est un graphe contenant des dépendances. Il nous permet de lier toutes les parties d'une application. Nous pouvons avoir plusieurs représentations. Le *workflow* peut nous montrer l'algorithme utilisé. Alors que le *dataflow* permet d'organiser les tâches pour gagner du temps. N'importe quel langage peut-être utilisé pour le représenter. Il y a trois niveaux de parallélisation :

- Workflow,
- Données,
- Service.

A. Languages

Nous allons en voir trois types :

- **Directed Tasks Graphs**, nous pouvons utiliser des DAG. C'est une façon très naïve de représenter les workflows,
- **Structured Workflow Graph**, BPEL est un workflow pour les web services. Les parties d'un programme doivent communiquer entre elles. Contrairement à la partie d'un programme, un service doit être indépendant mais peut communiquer avec d'autres services. Ce service est dynamique. En BPEL, un fichier XML contient le plan détaillé du workflow. Nous découvrons ce que chaque balise signifie,
- **Data-driven workflow**, plus connu sous le nom de scientist workflow. Nous pouvons utiliser comme exemple Simple Concept Unified Flow Workflow (Scufl). En clair, cet exemple porte sur une stratégie sur l'itération.

B. Scripting approach

Le but est d'utiliser des fonctions telles que Foreach, fork, join et dopar. Ces fonctions nous permettent d'établir un graphe plus logique et nous obtenons les mêmes informations que dans le graphe original. Nous pouvons alors plus facilement analyser ce graphe. Un spécialiste peut consacrer tout son temps à établir ce type de script. De plus, la plupart des langages bloquent lors de la déclaration de variable. Nous pouvons constater que lors d'un control plus de détails sont présents alors que lors du data plus d'abstraction est utilisé.

Plusieurs techniques sont utilisées pour gérer ces données telles que les tableaux, les activités et le port. Ces techniques nous permettent donc de définir les différents besoins sur les données. **Iteration strategies** permettent de garder ce qu'il a d'important. **Data handling inside workflow** se focalise seulement sur les processeurs dont nous voulons distribuer les tâches. **Les conditions et les boucles** fonctionnent normalement.

III. THE WORKFLOW MANAGER

Le but est de finalement déléguer la tâche au développeur. Il existe plusieurs types de manager, le businessman ou le scientifique. L'organisation du temps est fréquemment rencontrée. C'est un problème dure à résoudre surtout lorsqu'il y a quelques limites. Une liste d'heuristique en découle:

- Connaître le chemin critique. (prendre en compte les bloquants),
- Prioriser les tâches (en prenant en compte les coûts élevés),
- L'abstraction est un point important.

IV. RESOURCES MAPPING

L'exemple du Condor matchmaker développé pour créer une infrastructure. Le processus consiste à prendre en compte les caractéristiques de l'infrastructure, nous en décelons trois classe : les jobs, les scheduler et les groupes. C'est un langage très simple à utiliser. Le worker utilise un job qui est retracé par une ombre. Le **pilot job** est une stratégie pour organiser l'exécution des tâches. Il permet de répondre à la question -Comment minimiser l'impact de la file ? - Pour résumer, c'est un job permettant d'organiser l'application. Un pilote n'ayant rien à faire disparaît.

V. CONCLUSION

Le partiel s'effectuera le 12 février 2019, tout document sera autorisé, même le pc, aucun algorithme sera demandé, seulement comprendre la logique de comment ça marche. La prochaine séance se consacrera sur un travail sur Spark qui est un processus Hadoop. L'utilisation de RDD, map reduce

¹This work was not supported by any organization

task. Et le workflow processing. Le mieux serait de jeter un coup sur le lab concerné.

ACKNOWLEDGMENT

Thanks to Johan Montagnat for his work.

REFERENCES

- [1] <http://www.i3s.unice.fr/~johan/>