



Ecole d'ingénieurs et d'architectes de Fribourg
Hochschule für Technik und Architektur Freiburg



BIOZENTRUM
UNIVERSITÄT
WÜRZBURG

POP-Java : Parallélisme et distribution orienté objet

Beat Wolf, Pierre Kuonen: *iCoSys, Haute École Spécialisée de Suisse occidentale, Fribourg*
Thomas Dandekar: *Biozentrum, Universität Würzburg*

POP
java



Présentation

- Le modèle de programmation POP
- Le langage POP-Java
- Implémentation
- Évaluation
- Conclusion

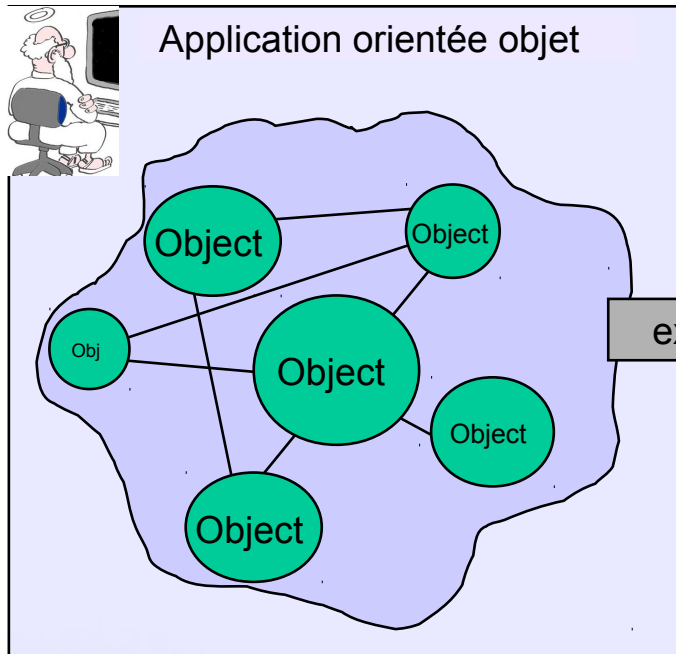


Programmation distribuée

- Très populaire
- Environnement hétérogènes, multi-machines et multicœurs
- Souvent compliqué à comprendre, différents modèles de programmation pour la distribution et le parallélisme local

Programmation distribuée

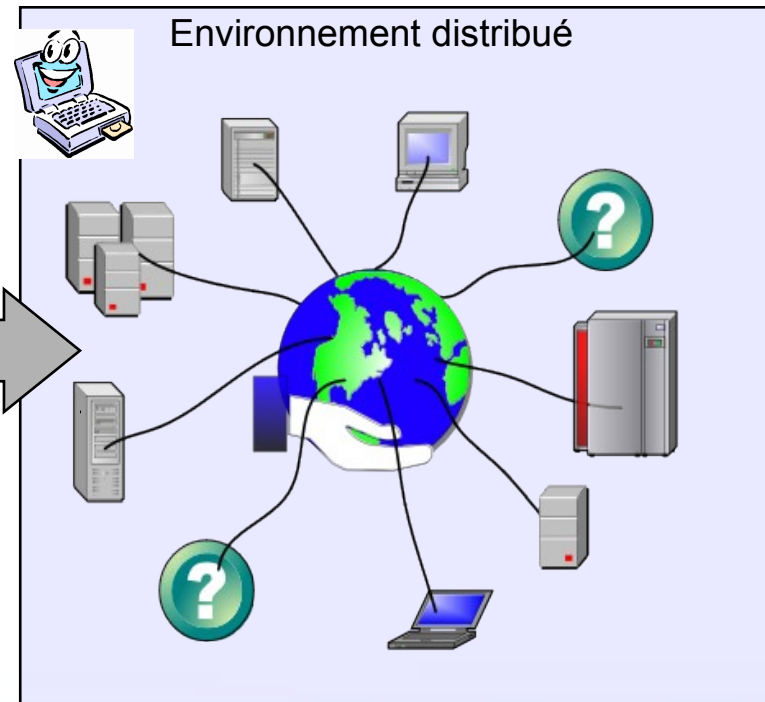
Software



- Hétérogènes
- Dynamique

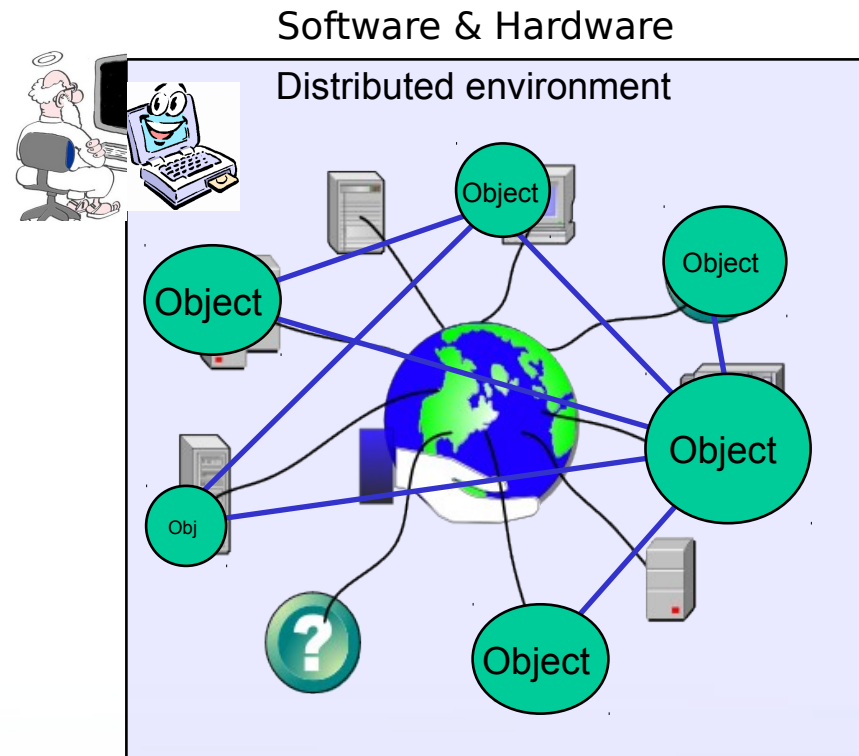
execute

Hardware



- Hétérogène
- Grande échelle
- Non structuré
- Dynamique, topologie inconnue

Programmation distribuée





Modèle POP

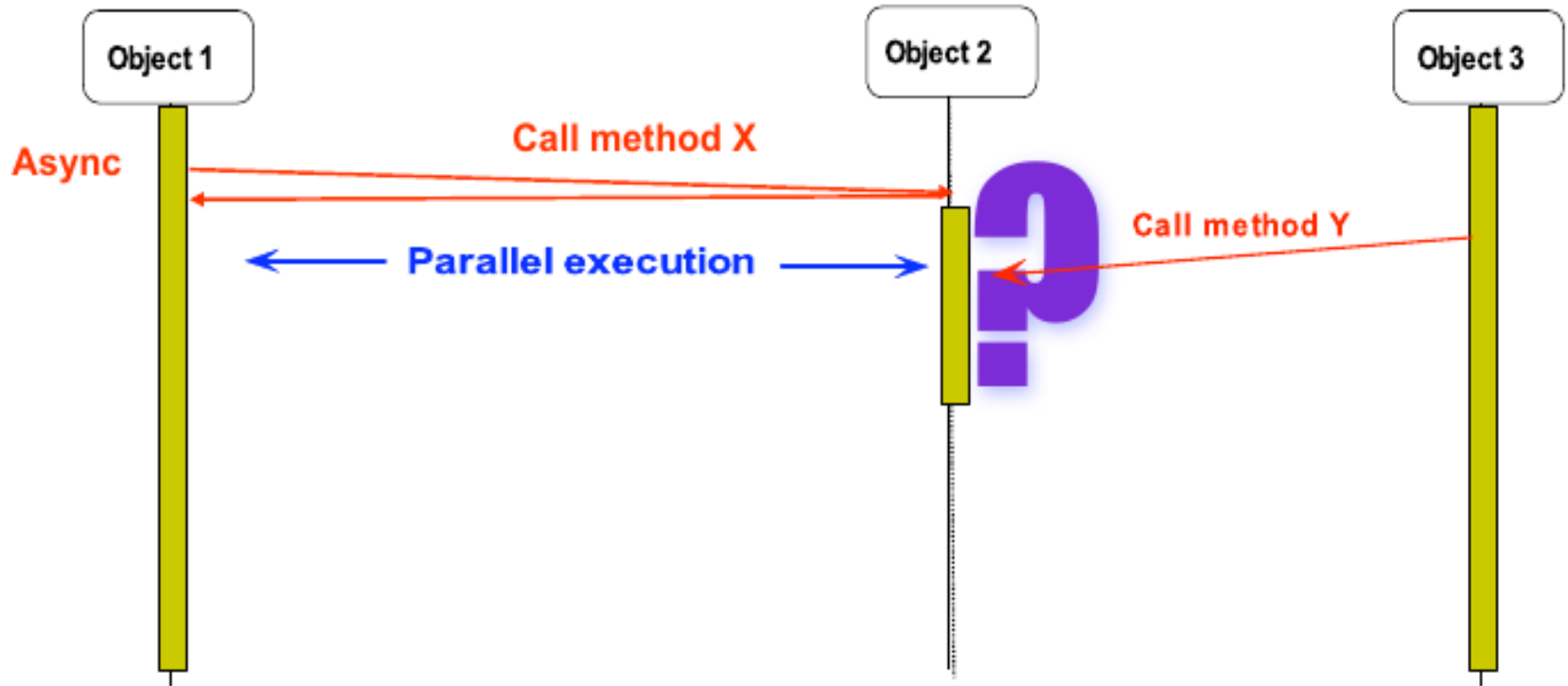
- Modèle de programmation distribué
- Objet = structure pour distribuer données et calculs
- Pas de mémoire partagé
- Programmation d'application distribués d'une manière orienté objets
- Même modèle pour la distribution et le parallélisme locale



Modèle POP

- Création transparente d'objets locaux et distants
- Sémantiques d'accès pour les méthodes publiques de chaque objet. Par exemple : appel synchrone ou asynchrone et leur comportement dans un environnement parallèle

Modèle POP

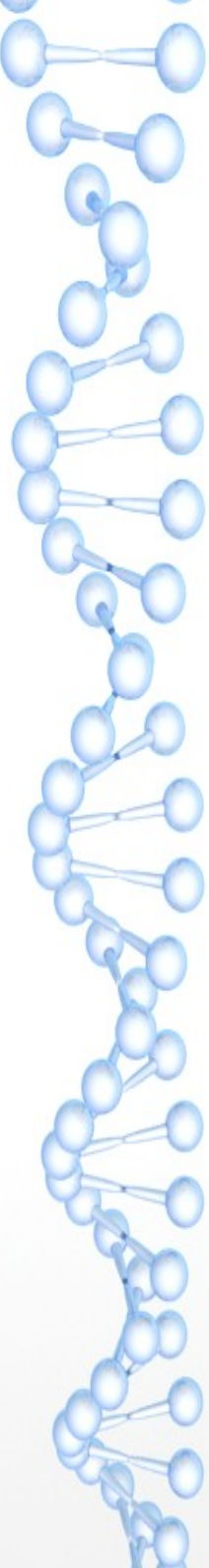




Modèle POP

- Sémantique d'accès
 - Synchronicité:
 - *Synchrone*
 - *Asynchrone*
 - Concurrence
 - *Concurrent*
 - *Séquentiel*
 - *Mutex*
- Méthode classique: *Synchrone Concurrent*
- Configuration objet : Machine distante a utiliser etc.

POP-C++



```
parclass Integer
{
    public:
        Integer() @ {od.url("localhost");};
        Integer(char* url) @ {od.url(url);};

        sync conc int Get();
        async seq void Set(int val);
        async mutex void Add(Integer &i);

    private:
        int value;
};
```



POP-Java

- Créations et accès aux objets distants
- Déclaration de la sémantique d'accès
- Intégration dans le langage
 - Extension du langage avec mots clés
 - **Annotations Java 5+**
- Pré compilateur JavaCC

Exemple de code

```
@POPClass
public class Integer {
    private int value;

    @POPObjectDescription(url = "localhost")
    public Integer(){}

    public Integer(@POPConfig(Type.URL) String url){}

    @POPSyncConc
    public int get(){
        return value;
    }

    @POPAsyncSeq
    public void set(Integer i){
        value = i.get();
    }

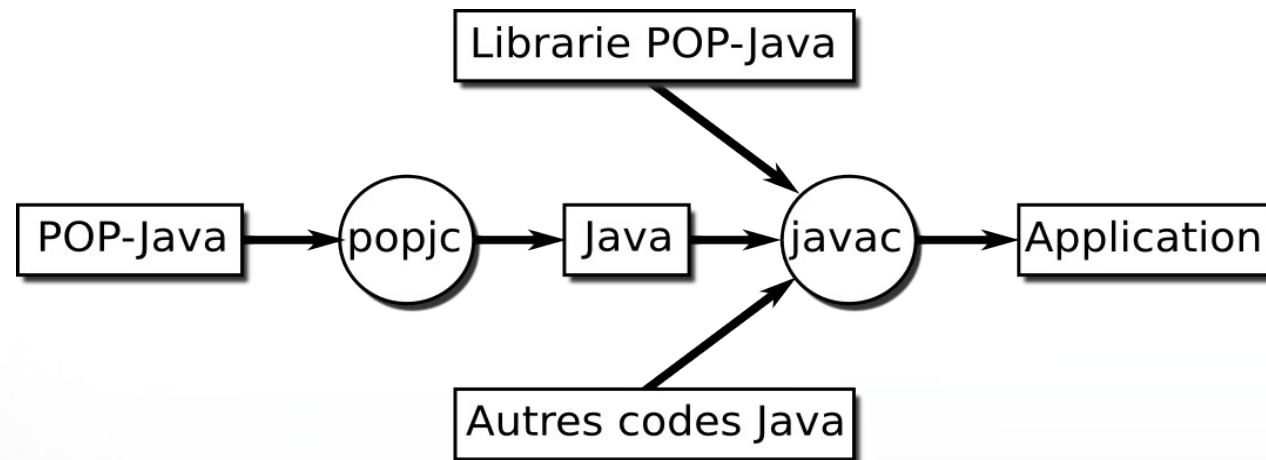
    @POPAsyncMutex
    public void add(Integer i){
        value += i.get();
    }
}
```

Integer a = new Integer();

Integer b = new Integer("1.2.3.4");

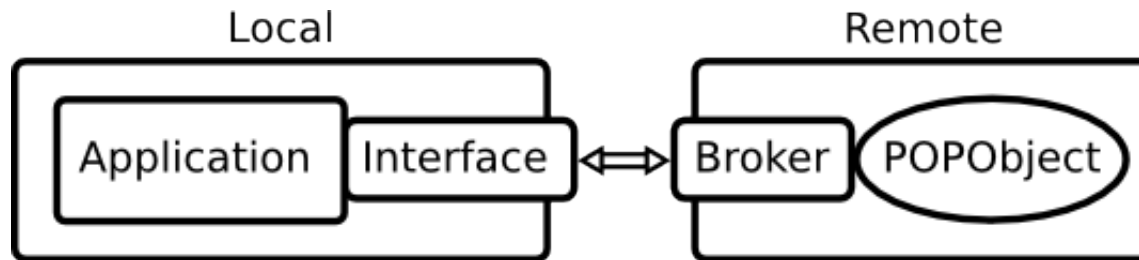
Compilation

- Extension “.pjava” pour les fichiers source
- Conversion en fichiers “.java”



Création d'objet

- Interception des constructeurs d'objets POP
 - `Integer a = new Integer();`
 - `Integer a = PopJava.newActive(Integer.class);`



- Connexion SSH
- Protocole documenté, interopérable avec POP-C++



Évaluation

- Alignement ADN
- Algorithme important en bio-informatique
- Algorithme de recherche de sous-chaîne
- Texte de référence: 3 GB (référence humaine)
- Plusieurs millions de chaînes a chercher, tailles des chaînes 50-1000 caractères

Algorithme

Reference

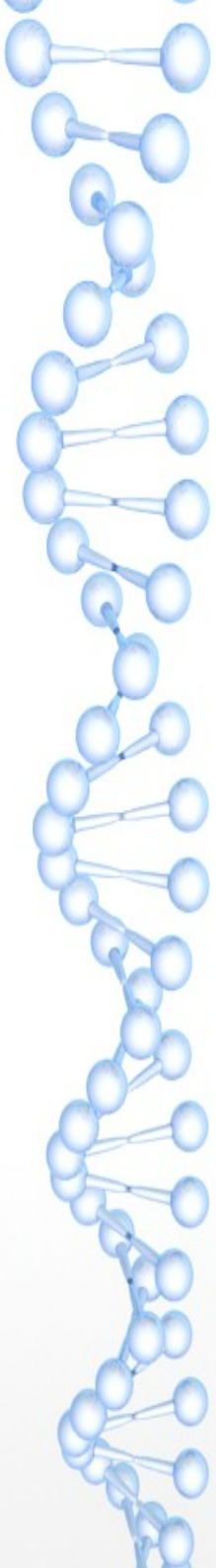
...ACTGTTACTACTTTGGTACGTTGTGATGCTAGCTAGTC...

1 ACTGTTACTACTT

2 ACTGTTACTACTT

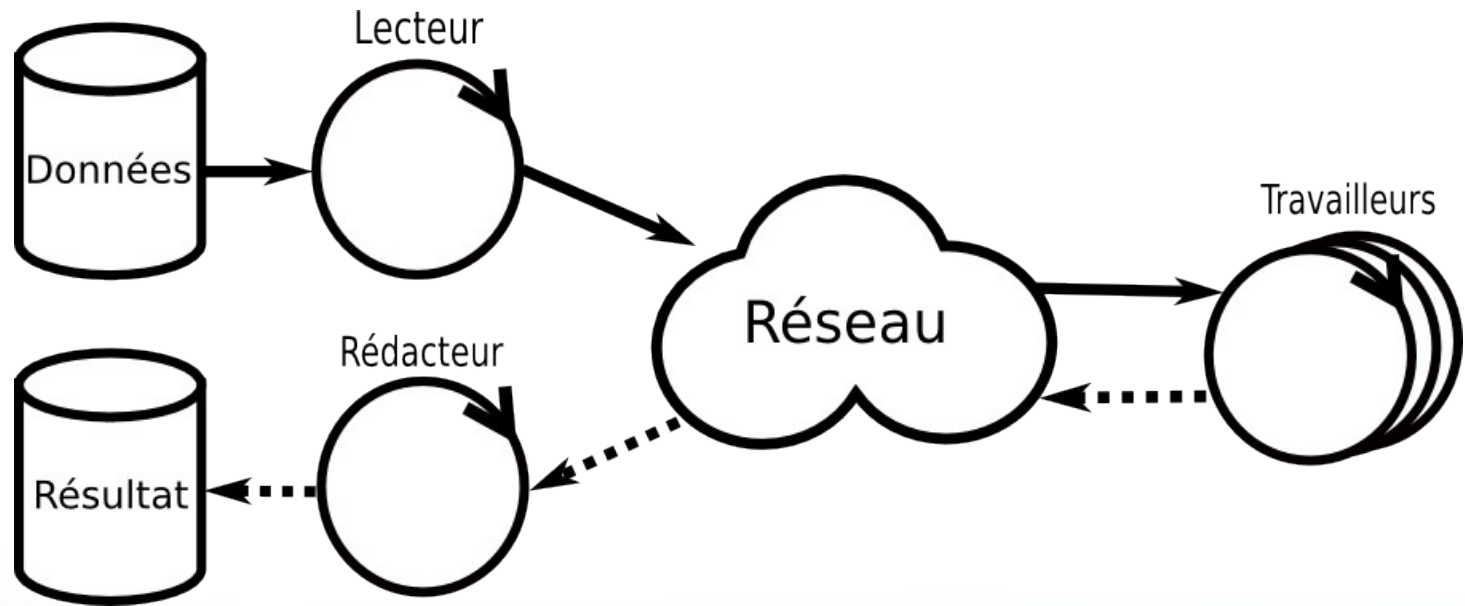
...

n ACTGTTACTACTT



Algorithme

- Producteurs/Consommateurs

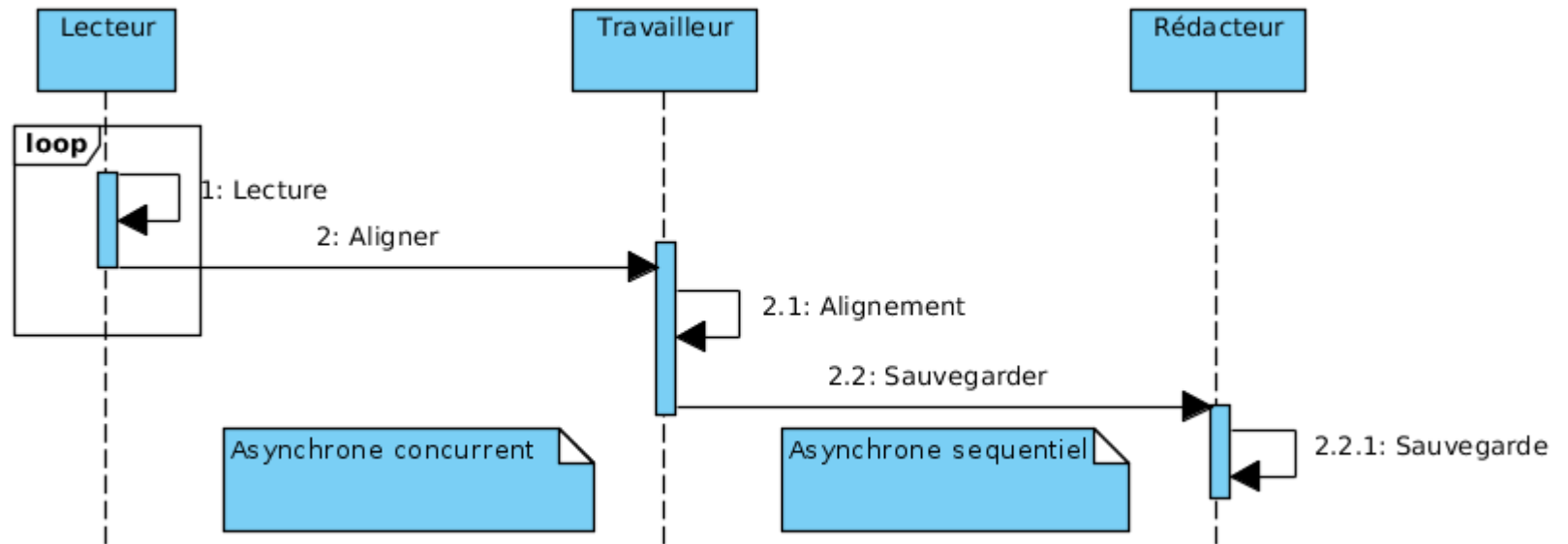




Évaluation

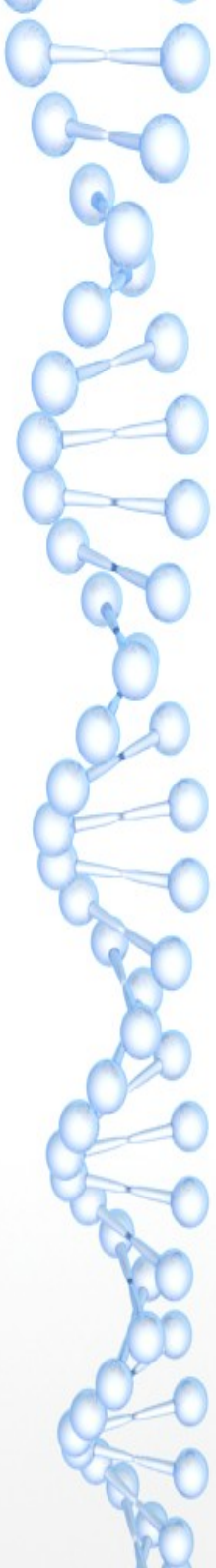
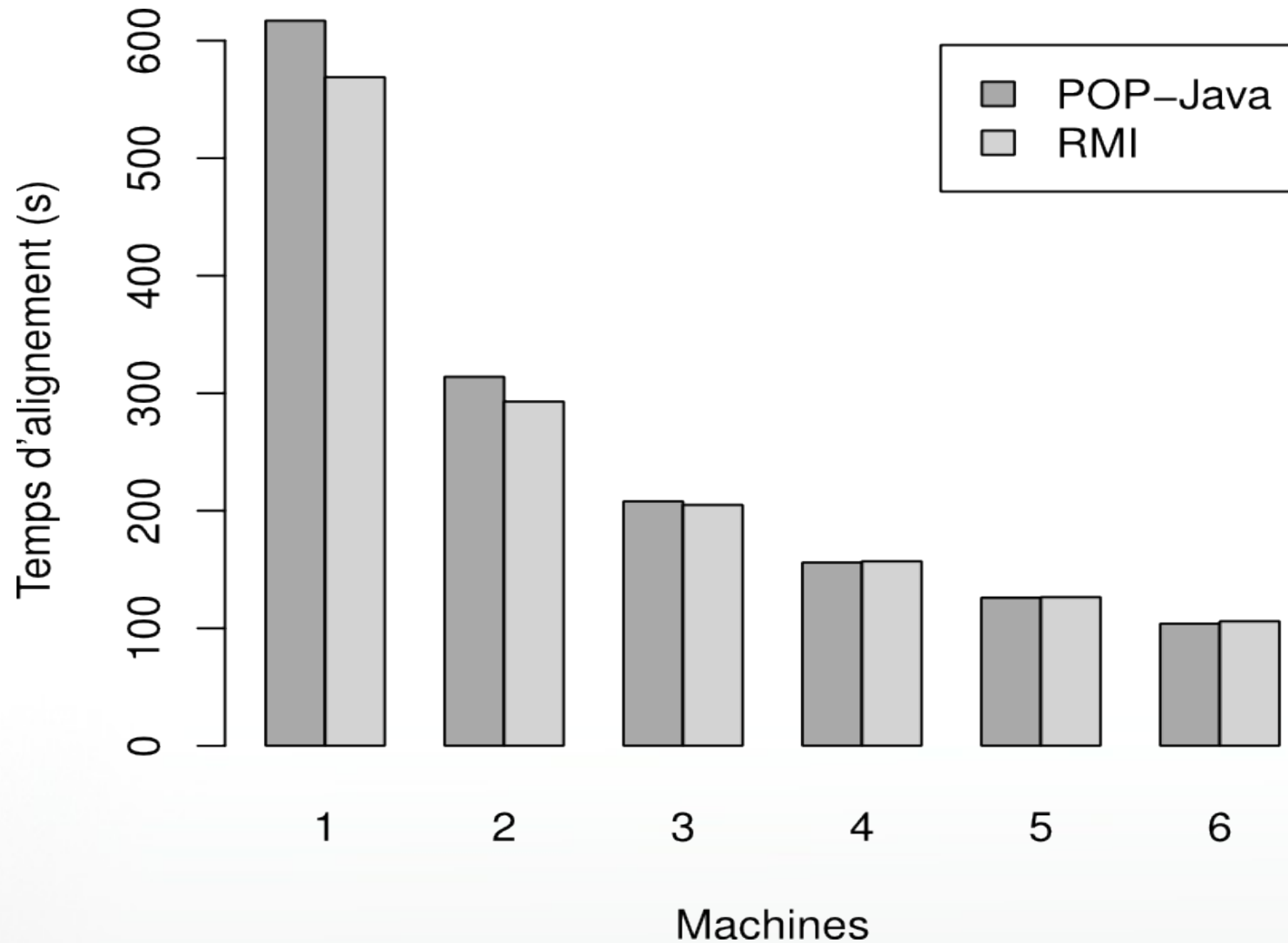
- Implémentation de l'algorithme en POP-Java et RMI
- Comparaison des performances
 - Cluster de 7 machines, 6 utilisées pour le calcul
- 16 millions de séquences aligné contre le chromosome humain 7 (158 MB)

Implémentation POP-Java

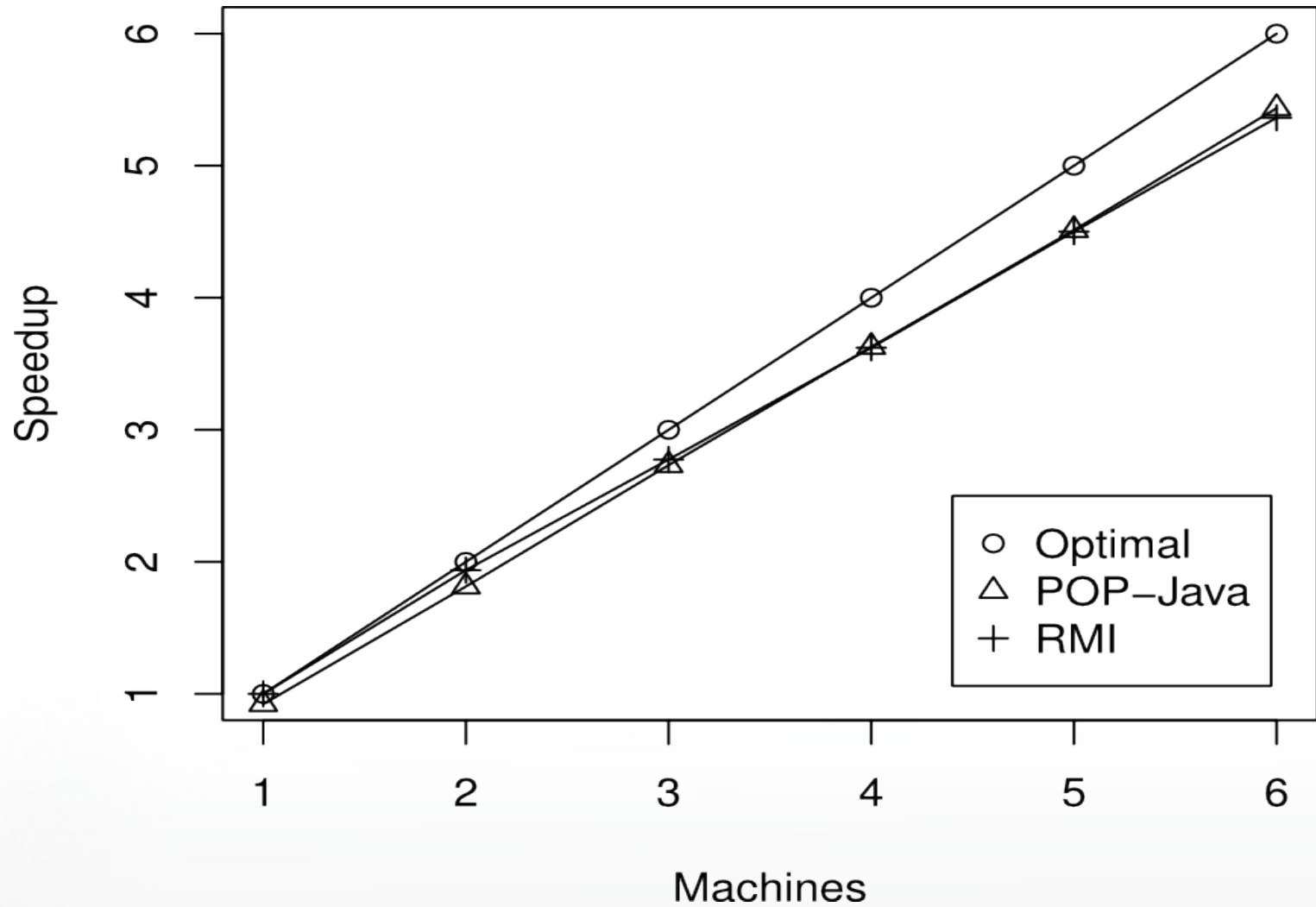


- 300 lignes de code POP-Java, 1200 RMI

Benchmark - Temps d'exécution



Benchmark - Accélération





Futur

- POP-Java Cloud
- Windows
- Fonctionnalités
- Intégration encore plus facile
- Version 1.0



Conclusions

- Facilité d'utilisation
- Performances compétitives
- Réduction de la complexité du code source

Questions



<http://gridgroup.hefr.ch/popjava/>

Création d'objet

