

Retours (biaisés) sur le langage Rust

François-Xavier Pineau¹

¹Centre de Données astronomiques de Strasbourg

23 mars 2021



□ Rappels sur Rust

- OV France 2019: “Le langage Rust: concepts et exemples”
- Rust combine:
 - contrôle de bas niveau: **performance**
 - fonctionnalités de haut niveau: **confort d'utilisation**
 - environnement **moderne**: Cargo (config toml), crates.io, doc markdown, tests
- **Performances de C/C++**, mais **sûr**
 - comme Java: **si ca compile, ca marche**, mais **sans GC**
 - compilé: temps de démarrage = lancement d'un process
- Le coût: contraintes du **RAII**
 - le compilateur ne laisse rien passer!
 - au début: on est obligé de se poser des questions, mais...
 - **avec un peu d'expérience**:
 - on apprend à mieux coder
 - **on se repose sur le compilato** (et on gagne du temps) au lieu de se battre contre

□ Quoi de neuf depuis?

- “Most loved language” sur Stack Overflow depuis 5 ans
- Plan de licenciement de [250 personnes](#) chez [Mozilla](#) en aout 2020, mais...
- Equipe dédiée chez Amazon pour [AWS](#)
- Equipe dédiée chez Microsoft, cf [ici](#) ou [ici](#)
- Transfert de propriétés de Mozilla à la récente [fondation Rust](#)
 - “Microsoft, Google, AWS, Huawei et Mozilla s’associent pour créer la Fondation Rust”
- Toujours en prod dans **Firefox**, **Dropbox**, [Facebook Libra](#) ...
 - rapide est sûr => populaire dans le monde la cryptomonaie
- Fait son entrée dans [le noyau linux](#) pour le dev. de [drivers](#)
- Fait son chemin pour entrer dans [curl](#)

□ Ecosystème

Petite sélection:

- [bindgen](#): “automatically generates Rust bindings to C libs”, ex [fitsio](#)
- [PyO3](#): “running and interacting with Python code from a Rust binary, as well as writing native Python modules”
- [Wasmbingen](#): “high-level interactions between Wasm modules and JavaScript”
- CLI: voir [clap](#) et/ou [structopts](#)
- Serveur Web: voir [arewewebyet](#): “no dominant framework at the level of Django or Rails”, “frameworks are smaller and modular, similar to Flask or Sinatra”
 - voir [actix-web](#), [rocket](#), ...
- GUI: voir [areweguiet](#) et [ce post](#). Bindings and wrappers for [GTK+](#), ...
- ...

Utilisation de Rust au CDS

- Public:
 - **MOCPy**: librairie MOC en python
 - origine 100% Python (Thomas Boch)
 - maintenant: 52% de Rust, 48% de Python (Matthieu Baumann, Thomas Boch)
 - intérêt: bien plus performant!
 - **Aladin Lite V3.0** basé sur WebGL, voir [démon](#) (Matthieu Baumann, Thomas Boch)
 - **cdshealpix rust** (F.-X. Pineau) et **python** (Matthieu Baumann)
- Interne:
 - code d'accès aux grands catalogues (+datawrapper PSQL)
 - prototype de cross-match
 - début de lib astrometrique (propagation des mvts propres)
 - ...

```
[pineau@cdsxnmatch2015 rust-indexfile]$ time ./xmatch_v2 gala_dr1 sdss8 allsky nn --dmax 3.0 count
Xmatch progress: 100%
99201152 match found in 4 seconds
99201152

real    0m4.352s
user   1m47.487s
sys    0m15.443s
```

Xmatch prototype

```
pineau@cds-dev-fxp:~/IdeaProjects/rust-indexfile$ export step=0.001
pineau@cds-dev-fxp:~/IdeaProjects/rust-indexfile$ time RUST_LOG=xmatch_master ./
/xmatch_master -c config/slaves.toml gaia_dr1 igsl3 allsky dist -r 3.0 histo -s ${ste
p} | ./gp_histo
[00:00:10] ##### 100% 10.30.2.96:8080 - success
[00:00:11] ##### 100% 10.30.2.98:8080 - success
[00:00:10] ##### 100% 10.30.2.99:8080 - success
[00:00:10] ##### 100% 10.30.2.100:8080 - success
[00:00:11] ##### 100% 10.30.2.101:8080 - success
[00:00:11] ##### 100% 10.30.2.102:8080 - success
[00:00:11] ##### 100% 10.30.2.103:8080 - success
[00:00:10] ##### 100% 10.30.6.25:8080 - success
Completed: 8; Failed: 0
Histogram contains 1058702383 matches.
```

```
real    0m11.452s
user    0m0.103s
sys     0m0.087s
pineau@cds-dev-fxp
```

```
[[slaves]]
addr = "10.30.2.96:8080"
partitions = [0]
replicas = []

[[slaves]]
addr = "10.30.2.98:8080"
partitions = [1]
replicas = []

[[slaves]]
addr = "10.30.2.99:8080"
partitions = [2]
replicas = []

[[slaves]]
addr = "10.30.2.100:8080"
partitions = [3]
replicas = []

[[slaves]]
addr = "10.30.2.101:8080"
partitions = [4]
replicas = []

[[slaves]]
addr = "10.30.2.102:8080"
partitions = [5]
replicas = []

[[slaves]]
addr = "10.30.2.103:8080"
partitions = [6]
replicas = []

[[slaves]]
addr = "10.30.6.25:8080"
partitions = [7, 8]
replicas = []
```

Fenêtre gnuplot 0

Xmatch separation histogram

separation histogram

Counts

Angular distance [arcsec]

Figure 1: Xmatch

□ Le cas de CDS HEALPix

- Un même coeur en Rust, cf [github](#)
 - Prototype de cross-match et d'accée aux grands catalogues en Rust (interne)
 - Des librairies Python (cdshealpix Python, MOCPy)
 - Fonctions dans PSQL (pour SIMBAD et Vizier? En cours...)
 - pg-sphere?
 - Aladin Lite WebGL (en WebAssembly)

□ Freins à l'adoption

- Apprendre un nouveau langage avec une courbe d'apprentissage raide au début
 - être convaincu et motivé
 - avoir un peu de temps (il faut 1 mois pour être productif)
 - (ne pas se battre mais s'aider du compilateur!)
- Écosystème moins riche que Python (pas d'Astropy)
 - mais wrapper Python: PyO3 (Rust <-> Python)
 - binding Rust existent pour l'API C de NumPy, ...
- Communique difficilement avec Java (ok via JNI, comme C, C++)
- Niche de C/C++. Ok pour:
 - code critique avec besoin de performances (et/ou de sécurité)
 - CLI (Command Line Interface)
 - bibliothèques utilisables depuis plusieurs langages
- ...