

Le langage de programmation D

Software meeting 15/10/2014

Christophe Meessen

CPPM-IN2P3-CNRS



Pourquoi un nouveau langage ? (1/2)

- Le langage C est ancien
- Nombreux concepts et techniques inventés depuis
- C++ évolue par accrétion de nouvelles fonctionnalités
- Une refondation est nécessaire

Exemples:

- Tableaux dynamiques et associatifs
- Chaînes de caractères avec concaténation et Unicode
- ABI standard pour interopérabilité

Pourquoi un nouveau langage ? (2/2)

- Programmation fonctionnelle: closure, etc.
- Documentation automatique
- Paramètres In, Out, Ref et évaluation Lazy des args
- Garbage collector
- Distinguer classes et structs: agrégats légers
- Fiabilité maximale: boundary checking, contract prog.
- Tests unitaires, Try-catch-finally, synchronize,
- ...

Paradigmes de programmation (1/2)

Programmation impérative quasi identique au C

Programmation Orientée Objet comme Java

Tout dérive de la classe `Objet`, interfaces, pas d'héritage multiple, toute méthode est virtuelle sauf si finale

Métaprogrammation par templates,

exécution de fonctions à la compilation, `static if`.

Paradigmes de programmation (2/2)

Programmation fonctionnelle

function & delegate, closures, map, filter, reduce

Programmation parallèle & concourante

Communication et synchronisation par échange de messages entre fils d'exécution

Example

```
#!/usr/bin/env rdmd (Linux)
import std.stdio;
void main() {
    ulong lines = 0;
    double sumLength = 0;
    foreach (line; stdin.byLine()) {
        ++lines;
        sumLength += line.length;
    }
    writeln("Average line length: ",
           lines ? sumLength / lines : 0);
}
```

Références

Site web: **<http://dlang.org/>**

Livre en ligne : **<http://ddili.org/ders/d.en/>**

Spécifications de D : **<http://dlang.org/spec.html>**

Doc. Modules : **<http://dlang.org/phobos/>**