

# DÉVELOPPEUR / INFORMATICIEN

— **Sciences mathématiques** (bachelor) — ... — **Sciences informatiques** (master) —

$$\sum_{i=0}^{n-1} f(i)$$

```
sum(f(i) for i in range(n))
```

**Olivier PIRSON**

Belge, né en 1975

 [www.linkedin.com/in/olivierpirson](http://www.linkedin.com/in/olivierpirson)

 <https://europa.eu/!CF64fw>

 [bitbucket.org/OPiMedia](https://bitbucket.org/OPiMedia)

 Bruxelles, Belgique

 [olivier.pirson.opi@gmail.com](mailto:olivier.pirson.opi@gmail.com)

 [www.opimedia.be](http://www.opimedia.be)

## ÉTUDES ET FORMATIONS (liste complète : [www.opimedia.be/CV/realisations.htm](http://www.opimedia.be/CV/realisations.htm))

2019 : Courte formation **Testing**, Treenity Solutions

Test d'APIs REST avec **Postman** et automatisation avec **Jenkins**

2019 :  **Master en sciences informatiques**, ULB, 74% **distinction** (suivi en **anglais**)

Matières **théoriques** et projets **pratiques**.

(Modules : optimisation, algorithmes. Options : langages de programmation, parallélisme.)

Mémoire : **Parallélisation d'un interpréteur abstrait, au travers du modèle acteur**

2015 : **Laboratoire de programmation Java**, Business Training via Bruxelles Formation

Apprentissage et développement d'une application web *RepairCenter* en méthodologie **Scrum**

2015 : **Cours de l'EPFC**

*Algorithmique – conception et analyse, Préparation à la Certification **Java**, Programmation **C++***

2013 – ... : **MOOC** Nombreux **MOOCs** (cours en ligne) **universitaires**, EPFL, Princeton, Stanford. . .

**Algorithmique, compilateur, langages, optimisation** mathématique, **paradigmes fonctionnel**. . .

**Bash, C++, Java, ML, Oz, Python, Scala**. . . ; **Git**. . . ;

**Anglais**. . .

2011, 2014 et 2016 : Courtes **formations** Evoliris

**Android**, gestion de projet, **GNU/Linux**, HTML5/CSS3, réseau, sécurité, **UML**

2010 : **Formation professionnelle de développeur web**, Business Training via Bruxelles Formation

**UML**, (X)HTML/CSS, MySQL, Apache HTTP Server, PHP, JavaScript, Flash/ActionScript

1998 :  $\pi$  **Bachelor en sciences mathématiques, orientation informatique**, UCL, satisfaction

Projets informatiques en C++, Pascal, Java, Tcl/Tk. . .

## CONNAISSANCES INFORMATIQUES

**Développement** : Ada, OpenCL, **C**, **C++**, **Java**, **Python**, **Scala**, Scheme, *etc.*

Développement web : JavaScript/jQuery/AJAX, PHP, (X)HTML/CSS/Sass. . . ; CMS

Bases de données : **MySQL**, QuickBase. . .

Systèmes d'exploitation : **GNU/Linux**, Debian, Android, Windows. . .

Outils : Apache HTTP Server, Doxygen, Eclipse, Emacs, **Git**/Mercurial, (L<sup>A</sup>)T<sub>E</sub>X, **Makefile**, regex, shell. . .

Divers : calcul **parallèle**, prog. **fonctionnelle** et **orientée objet (POO)**, UML. . .

CV du 2 novembre 2020.

Mise à jour et version en  anglais

en ligne : [www.opimedia.be/CV](http://www.opimedia.be/CV)



## LANGUES

- 🇫🇷 **Français** : langue maternelle
- 🇬🇧 **Anglais** : niveau CECR B1+  
(le master de l'ULB était en anglais)

## DIVERS

- 🚗 Permis de conduire B
- 📄 Détenteur d'une carte **activa.brussels** : **déduction de 350 à 800€** par mois

## EXPÉRIENCES PROFESSIONNELLES

2020 – ... : **Solution Building Engineer chez Sopra Steria** (consultance TIC)

2015 – 2016 : **Développement web pour ArtCoreWeb** (développement web)

Lancement d'une activité de développement web (via JobYourSelf, SMart. . . avec l'objectif de devenir indépendant), avec un partenaire s'occupant des parties commerciale et design.

- Conception et développement d'un site web pour l'activité : illustration récursive interactive de la démarche partant d'une demande pour aboutir à sa réalisation. HTML/Sass, JavaScript  
[www.opimedia.be/CV/2015\\_ArtCoreWeb/ArtCoreWeb](http://www.opimedia.be/CV/2015_ArtCoreWeb/ArtCoreWeb)
- Développement d'un site web statique : HTML/CSS, Bootstrap, JavaScript.
- Développement d'un prototype d'application **Android** en **Java** pour facilement prendre des photos d'articles en magasin et les ajouter avec descriptif sur un e-commerce.
- Analyses techniques, rédaction de rapports et devis.
- Apprentissage.

MySQL, PHP, HTML/CSS/Sass, Bootstrap, JavaScript, jQuery, CMS, e-commerce, Android, Java

2011 – 2013 : **Développeur web chez Umedia** (production cinéma)

Analyse, développement web **full stack** (MySQL, PHP, HTML/CSS, JavaScript), déploiement et suivi sur le **CMS MindTouch** utilisé en interne pour le partage de documents et la communication. Développement d'un template. Interaction avec l'API. Développement d'outils à intégrer : calendrier, "mur à la Facebook", réservation de lunch. . .

Licencié suite au remplacement du CMS MindTouch par un autre logiciel géré par une entreprise externe.

- Administration du CMS.
- Migration du CMS d'un serveur web Apache sous Ubuntu vers un serveur web IIS sous Windows.
- Croisement de la base de données des employés avec la liste des utilisateurs du CMS.
- Développement d'une application de ticketing en MySQL, PHP, HTML/CSS, JavaScript.
- Dépannages IT ou transfert à l'administrateur système responsable du parc informatique.
- Aide aux stagiaires business pour l'élaboration de base de données en ligne avec QuickBase.

**Ubuntu**, Apache HTTP Server, Windows, IIS, MySQL, **PostgreSQL**, PHP, HTML/CSS, JavaScript, jQuery, AJAX, MindTouch, QuickBase

décembre 2010 – février 2011 : **Stage de développeur web chez Vertige** (développement web)

Stage faisant suite à la formation professionnelle de développeur web.

- Analyse, développement web full stack (MySQL, PHP, HTML/CSS, JavaScript) et déploiement d'un "mur à la Facebook" pour le site web [Comedien.be](http://Comedien.be) utilisant le CMS **SPIP**.  
Screenshot du mur : [www.opimedia.be/CV/2010\\_Vertige/Comedien.be/Mur\\_de\\_Comedien.be.png](http://www.opimedia.be/CV/2010_Vertige/Comedien.be/Mur_de_Comedien.be.png)
- Refonte du système de connexion du site web [Comedien.be](http://Comedien.be). Factorisation du code existant.


MySQL, PHP, HTML/CSS, JavaScript, jQuery, AJAX, SPIP

## EXPÉRIENCES DE DÉVELOPPEUR, PAR LANGAGE

⚠ Cette dernière section liste des projets personnels ou didactiques classés par **langage de programmation** en commençant par les langages les plus proches de la machine : assembleur, OpenCL, **C**, **C++**, Bash, BASIC, Pascal, PHP, JavaScript, Java, Processing, **Python**, Julia, Scheme, Erlang, **Scala**, Haskell, Prolog,  $\text{\LaTeX}$ . La mention des plus anciens projets a pour but d'illustrer que je programme depuis très longtemps.

Vous pouvez accéder aux sous-sections à partir de la table  $\equiv$  des matières de votre visionneur de PDF. Une courte sélection par ordre antichronologique est visible dans la section *Projects* de mon profil *europass* : <https://europa.eu/!CF64fw>

Les langages que j'ai utilisés et utilise le plus sont : **C**, **C++**, **Python**.

Dépôts de mes projets **libres (free software, open source)** sur Bitbucket (🔗 **GitHub**-like) :  
 [bitbucket.org/OPiMedia](https://bitbucket.org/OPiMedia)

Bookmarks personnels sur les langages de programmation : [www.opimedia.be/DS/languages](http://www.opimedia.be/DS/languages)

**CodinGame** 🏆 niveau 21 🕒 mentor

🏆 Badges Gold et Certificats **HackerRank** : C, C++, Java, JavaScript, Python ; Problem Solving

✅ Badges **LinkedIn** : C (top 96%), C++ (top 99%), Java (top 79%), Python (top 70%) ;  
Bash (top 84%), Git (top 94%), HTML (top 96%)

### ASM **assembleur** — langage proche du langage machine

---

2017 projet du master (23/20) : **parallélisation** par instructions **SIMD x86**

Implémentation et analyse des performances d'un filtre d'image écrit en C puis en assembleur utilisant des instructions SIMD (single instruction, multiple data).

2017 projet du master (19,5/20) : architecture **RiSC-16**

Implémentation d'opérations arithmétiques avec plusieurs jeux d'instructions RiSC-16.  
Voir aussi le projet C++ *cpprisc16*.


2016 projet du master : assembleur **LLVM**, voir le projet Java *compilateur Fortran*

~1995 projets personnels : **assembleur x86**, voir bibliothèques C *sous MS-DOS* et *de grands entiers*



### **OpenCL** — dérivé de **C** pour programmation **parallèle** sur CPU multi-cœur et **GPU**

---

2018 projet personnel :  **assertOpenCL**

**Bibliothèque OpenCL** implémentant la macro **assert** manquante par rapport au C, avec des exemples de programmes hôtes en C, C++, Java, Scala et Python.

<https://bitbucket.org/OPiMedia/assertopencil>

2018 projet du master :  **évaluation et représentation graphique de L-systèmes**

Implémentation et analyse des performances de systèmes de LINDENMAYER en Python puis OpenCL (**PyOpenCL**) exécuté sur GPU. Il s'agit de systèmes de réécriture générant des structures récursives pouvant être interprétées comme des figures fractales, notamment des plantes. Génération d'images bitmaps, ou vectorielles avec le langage de programmation **PostScript**.

<https://bitbucket.org/OPiMedia/l-system-in-pyopencil>

2018 projet du master : voir le projet C++ *problème  $\sigma_{\text{impair}}$*

2017 projet du master (19/20) : **contrôleur de signal numérique dsPIC33** (architecture Harvard)  
Implémentation du calcul de transformée de FOURIER discrète, gérant dépassement et précision.

~1995 projet personnel : **bibliothèque et applications sous MS-DOS**

- Partie générale implémentant des **structures de données** et des fonctions d'aide.
- Partie gérant à partir des interruptions MS-DOS le **clavier, la souris et le mode texte**. Développement d'une interface avec menus déroulants telle que l'**IDE de Turbo Pascal**, gérée par une **boucle événementielle**.
- Partie gérant les **modes graphiques** : double buffering, affichage de backgrounds, automatisation d'éléments associés à des sprites et dynamique de mouvement dans une boucle événementielle.

Optimisation de fonctions critiques en **assembleur x86**.


Développement de plusieurs applications utilisant cette bibliothèque, dont un dictionnaire de citations : [www.opimedia.be/DS/grenier/online-DOS](http://www.opimedia.be/DS/grenier/online-DOS)

~1995 projet personnel : bibliothèque de calcul sur des **grands entiers**

Opérations arithmétiques sur des entiers sans limitations de tailles.  
Optimisation de fonctions critiques en assembleur x86.

2019 finaliste du **C/C++ Challenge Belgium & The Netherlands 2019** organisé par Dekimo

2018 projet personnel et du master (17/20) : **vérification numérique parallèle du problème  $\sigma_{\text{impair}}$**   
Développements mathématiques d'une conjecture arithmétique personnelle.

 Vérification numérique de cette conjecture avec les technologies **parallèles multithreading**, par **passage de messages (OpenMPI)** et pour **GPU (OpenCL)** :

[https://bitbucket.org/OPiMedia/parallel-sigma\\_odd-problem](https://bitbucket.org/OPiMedia/parallel-sigma_odd-problem)

Slides de présentation du problème et des résultats :

<https://speakerdeck.com/opimedia/parallel-numerical-verification-of-the-s-odd-problem>

**calcul distribué, cluster, GCC, clang, CxxTest, Cppcheck, Valgrind, Doxygen** ; multithreading, OpenMPI, OpenCL ; Makefile, gnuplot, Graphviz

2017 projet personnel :  **cprisc16**

Bibliothèque C++ réimplémentant les jeux d'instructions assembleurs RiSC-16, facilitant les expérimentations par rapport au simulateur RiSC-16 utilisé pour le projet du master *RiSC-16*.

<https://bitbucket.org/OPiMedia/cprisc16>

2017 projet du master (17,5/20) :  **problème d'optimisation PFSP**

Implémentation et analyse d'**heuristiques d'amélioration itérative** pour résoudre le problème d'optimisation *Permutation Flow-shop Scheduling Problem* (PFSP).

<https://bitbucket.org/OPiMedia/permutation-flow-shop-scheduling-problem>

2016 projet du master (18/20) : recherche de collisions partielles dans la fonction de hachage SHA-1  
Calcul **parallèle**/distribué par passage de messages avec la bibliothèque **OpenMPI**.

2016 projet du master (17/20) : simulateur d'**ordonnanceur**

Implémentation et analyse de méthodes (globale et partitionnée) d'ordonnancement de tâches.

1997 projet du bachelor : site web dynamique de gestion de stock

C++ côté serveur avec CGI en **Perl**. Côté client HTML et **Tcl/Tk** pour afficher des histogrammes.

1997 projet du bachelor : implémentation de structures de données

Implémentation et analyse des performances de plusieurs structures de dictionnaire : naïve, bibliothèque de l'époque, arbre binaire de recherche automatiquement équilibré AVL, trie.



## **Bash** — langage de script shell

---

2020 projet personnel :  *HackerRank [CodinGame...]/helpers*

Aide à la résolution de problèmes de type HackerRank, CodinGame... dans divers langages, en automatisant entre autre exécution et comparaison des résultats :

<https://bitbucket.org/OPiMedia/hackerrank-codingame-helpers>

2020 projet personnel :  *gocd*

Commande pour changer de répertoire, comme la traditionnelle commande `cd` mais avec un nom à la place d'un répertoire, à partir d'une liste d'association nom-répertoire :

<https://bitbucket.org/OPiMedia/gocd-bash>

## **B** **BASIC** — langage simpliste

---

~1993 projet personnel : dictionnaire de citations en AMOS BASIC sur Amiga

Voir aussi le projet PHP *OPiCitations*.

~1991 projets personnels : jeux *Tetris* et *Sokoban* en AmigaBASIC sur Amiga

~ 1985 – 1990 projets personnels : anciens jeux en BASIC sur TRS-80 puis GW-BASIC sous MS-DOS

## **P** **Pascal** — langage impératif

---

1992 projet scolaire de fin d'année : calculatrice de polynômes en Turbo Pascal, avec une TUI



## **PHP** — langage web côté serveur

---

Voir aussi [EXPÉRIENCES PROFESSIONNELLES DE DÉVELOPPEUR WEB](#).

2017 projet du master : voir le projet Python *IMDb*

2013 – ... projet personnel : **“...”** *OPiCitations*

Dictionnaire en ligne de citations : [www.opimedia.be/OPiCitations](http://www.opimedia.be/OPiCitations)

MySQL, PHP, HTML/Sass, JavaScript

2010 projet de formation : **WHM** – Wiki Hyper Media


Réalisation finale de la *formation professionnelle de développeur web* : [whm.opimedia.be](http://whm.opimedia.be)

rewriting Apache, MySQL, PHP, XHTML/CSS

## **JavaScript** — langage objet dynamique principalement utilisé dans les navigateurs web

---

Voir aussi [EXPÉRIENCES PROFESSIONNELLES DE DÉVELOPPEUR WEB](#).

2020 projet personnel :  *mixed radix*

Application HTML/Sass/JavaScript pour convertir les représentations de nombres d'une base mélangée à l'autre : [www.opimedia.be/mixed-radix](http://www.opimedia.be/mixed-radix)

2017 projet du master (17,5/20) :  *Disjoint Compatible Perfect Matchings*

Illustration d'un sujet d'articles scientifiques par une page web statique en HTML/Sass, avec une application JavaScript : [www.opimedia.be/CV/2016-2017-ULB/INFO-F420-Computational-geometry/Project-Disjoint-Compatible-Perfect-Matchings](http://www.opimedia.be/CV/2016-2017-ULB/INFO-F420-Computational-geometry/Project-Disjoint-Compatible-Perfect-Matchings)

2011 projet personnel : *txt2*


Application HTML/CSS/JavaScript pour appliquer diverses transformation sur des données textes : [www.opimedia.be/DS/online-tools/txt2](http://www.opimedia.be/DS/online-tools/txt2)



## **Java** — langage orienté objet


---

Voir aussi les langages [Scala](#) et [Processing](#) qui tournent sur la machine virtuelle Java, et [EXPÉRIENCES PROFESSIONNELLES DE DÉVELOPPEUR WEB](#).

2017 projet du master (20/20) :  *ChocoChess*

Résolution de **problèmes de satisfaction de contraintes** (CSP) avec la bibliothèque *Choco solver*.  
<https://bitbucket.org/OPiMedia/choco chess>

**Javadoc**, JUnit, Choco solver


2016 projet personnel et du master (14,5/20) :  **compilateur Fortran**

Développement en Java d'un compilateur pour un sous-ensemble de Fortran. Analyse lexicale avec **JFlex**. Génération automatique à partir de la grammaire Fortran d'un parseur écrit en Java, avec un programme développé en Python. Génération de code natif via l'assembleur LLVM.

Java, JFlex, Python, LLVM

2014 projet pour le MOOC *Creative, Serious and Playful Science of Android App* (20/20)

Adaptation pour Android du projet Processing *No Cross Circles*.

2015 projet de la formation *Laboratoire de programmation Java* :  **RepairCenter**

Développement d'une application web en méthodologie **Scrum**, dont le développement d'une bibliothèque Java pour automatiser la gestion des formulaires. Slides de la présentation finale :

[www.opimedia.be/CV/2015\\_Laboratoire\\_de\\_programmation\\_Java/RepairCenter.pdf](http://www.opimedia.be/CV/2015_Laboratoire_de_programmation_Java/RepairCenter.pdf)

UML, Eclipse, **SVN (Subversion)**, MySQL Workbench, Tomcat, **Java EE**, JSP, JPA, Hibernate **ORM**, Javadoc, Scrum

1998 projet du bachelor : implémentation de structures de données




## **Processing** — dérivé de **Java** pour des arts visuels

---

2013 projet pour le MOOC *Creative Programming for Digital Media & Mobile Apps* (20/20)

Application visuelle et sonore  **No Cross Circles** produisant des sons MIDI lorsque des disques se touchent : [www.opimedia.be/DS/Processing/#No-Cross-Circles](http://www.opimedia.be/DS/Processing/#No-Cross-Circles)




2020 – ... projet personnel en cours :  **slidesfromvideo**

Extraction automatique de slides à partir de vidéos, par l'utilisation des bibliothèques **OpenCV** et **Pillow** (PIL)

2020 – ... projet personnel en cours : **outcmp**

Pour un programme et des entrées, l'exécute et teste ses sorties en les comparant aux sorties donnés supposées correctes ; résume le tout dans un tableau

2018 – ... projet personnel en cours :  **tsv2htmltable**

Application et bibliothèque **Python 3** de conversion de données textes telles que TSV, CSV... en tables HTML interactives (ou  $\text{\LaTeX}$  ou autres données textes), avec possibilité de transformations par expressions rationnelles (**regex**) : <https://bitbucket.org/OPiMedia/tsv2htmltable>

Python 3, PyPy, **pytest** ; HTML, DataTables,  $\text{\LaTeX}$


2018 projet du master : voir le projet OpenCL *L-systèmes*

2017 projet du master (15/20) : problème d'optimisation BPP

Implémentation et analyse d'heuristiques pour résoudre le *Bin Packing Problem* (BPP).

2017 projet du master (18/20) : application web utilisant le **grand volume de données** de **IMDb**

Extraction des données du site web cinéma *IMDb* avec Python pour alimenter une base de données MySQL. Application web dynamique : Apache HTTP Server, PHP, HTML/Sass.

2016 projet du master (20/20) :  **p0ss1ble**

Application interactive avec visualisation HTML pour casser un ensemble de messages ayant été chiffrés avec la méthode sûre AES-128 CTR, mais dont le mésusage la rend fragile.

<https://bitbucket.org/OPiMedia/p0ss1ble>

2016 projet du master (11/20) : structure de données à partir d'un article scientifique

Implémentation d'un arbre binaire de recherche persistant.

Slides de présentation de la structure : <https://speakerdeck.com/opimedia/persistent-search-trees>

2016 – 2019 concours de programmation de l'EPFC dans la catégorie ancien étudiant

Successivement **premier prix**, second prix, premier prix puis premier prix ex æquo.



Solution pour 2019, avec quelques développements théoriques :

<https://bitbucket.org/OPiMedia/epfc-contest-15-2019-le-lacher-de-boules-de-petanque>

2016 projet du master : voir le projet Java *compilateur Fortran*

2013, 2014 projets pour le MOOC **Discrete Optimization** (18,5/20) :

Implémentation d'algorithmes pour résoudre des problèmes d'optimisation discrète : knapsack, graph coloring, traveling salesman, warehouse location, vehicle routing.


2013 – ... projet personnel :  **SimpleGUICS2Pygame**

Bibliothèque réimplémentant en **Python compatible 2 et 3** des APIs de modules spécifiques (*SimpleGUI*. . .) écrits en JavaScript dans l'implémentation de Python *CodeSkulptor*.

<https://simpleguics2pygame.readthedocs.io>

Des **utilisateurs de par le monde** m'ont contacté, directement ou indirectement par exemple par [stackoverflow.com/search?q=SimpleGUICS2Pygame](https://stackoverflow.com/search?q=SimpleGUICS2Pygame)

Python 2, Python 3, **matplotlib**, **Pillow** (PIL), **pygame**, **pycodestyle** (pep8), **Pylint**, **mypy**, **pytype**, **pydeps**, **Sphinx**

2013 projet pour le MOOC *An Introduction to Interactive Programming in ...* (20/20) :  RiceRocks

Petit jeux dans l'environnement en ligne *CodeSkulptor*, utilisant le module *SimpleGUI* :

[https://py3.codeskulptor.org/#user305\\_XNvcqTxIBngtHPu.py](https://py3.codeskulptor.org/#user305_XNvcqTxIBngtHPu.py)

Liste d'autres programmes :

[https://simpleguics2pygame.readthedocs.io/en/latest/\\_static/links/prog\\_links.html](https://simpleguics2pygame.readthedocs.io/en/latest/_static/links/prog_links.html)

~ 2005 – 2009 projet personnel :  *DS Python*

Bibliothèque Python 2 implémentant des fonctions arithmétiques, une évaluation des combinateurs (**logique combinatoire**), et une **machine à registres illimités (URM)**. Quelques applications Python/**Tkinter** l'utilisant. [www.opimedia.be/DS/DSPython](http://www.opimedia.be/DS/DSPython)

 **Julia** — langage dynamique pour le calcul scientifique

---

2017 projet du master (17,5/20) : problème d'optimisation linéaire *p-center location*

Implémentation et analyse de plusieurs formalisations de ce problème.

 **Scheme** — langage fonctionnel dynamique épuré, similaire à **Lisp**

---

~2003 projet personnel : bibliothèque implémentant des fonctions arithmétiques

 **Erlang** — langage fonctionnel **concurrent** basé sur le modèle d'**acteur**

---

2018 projet du master (pas terminé) :  implémentation **parallèle** d'un Twitter-like scalable

<https://bitbucket.org/OPiMedia/mini-twitter-in-erlang>

 **Scala** — langage mêlant paradigmes **objet** et **fonctionnel**, tournant sur la machine virtuelle **Java**

---

2019 **mémoire du master** (15,5/20) :  **Parallélisation d'un interpréteur abstrait**, au travers du modèle **acteur** — Application à l'interpréteur abstrait *Scala-AM*

Présentation théorique de l'**interprétation abstraite**, technique d'**analyse statique**, c'est-à-dire technique visant à automatiser la preuve de propriétés sur les programmes ou la mise en évidence de bogues, sans exécuter ces programmes.

Présentation théorique de la programmation parallèle et plus particulièrement du modèle d'acteur. Un programme **parallèle** est conçu pour exécuter simultanément plusieurs tâches réparties sur plusieurs processeurs/cœurs. C'est désormais le moyen d'accélérer les programmes qui de nos jours ne peuvent plus simplement compter sur l'accélération proprement dite des processeurs. Le modèle d'**acteur** encapsule ces tâches dans des entités isolées qui communiquent par envoi de messages asynchrones. Cela réduit la difficulté, les programmes parallèles étant de nature non déterministe et sujets à de nombreuses erreurs potentielles.

Implémentation pratique de plusieurs algorithmes parallèles dans l'interpréteur abstrait *Scala-AM* existant. Évaluation des performances sur l'analyse de programmes **Scheme**.

Mémoire :



<https://bitbucket.org/OPiMedia/efficient-parallel-abstract-interpreter-in-scala/raw/master/Parallelisation-d-un-interpreteur-abstrait-au-travers-du-modele-acteur--Olivier-Pirson-2019.pdf>

Slides de la présentation finale :


[speakerdeck.com/opimedia/parallelisation-dun-interpreteur-abstrait-au-travers-du-modele-acteur](http://speakerdeck.com/opimedia/parallelisation-dun-interpreteur-abstrait-au-travers-du-modele-acteur)

Implémentation *Scala-Par-AM* :  <https://bitbucket.org/OPiMedia/scala-par-am>

sbt, sbt-assembly, ScalaCheck, ScalaTest, **Akka**, **Scaladoc** ;  $\text{\LaTeX}$ , TikZ, gnuplot, Zotero






2018 projet du master (19/20) :  *Explore the Desert*

Développement d'un jeu en mode console et en mode graphique, avec le modèle de **parallélisme** mémoire transactionnelle logicielle (**STM**).

<https://bitbucket.org/OPiMedia/explore-the-desert-in-haskell>

GHC, Gloss, HUnit, Haddock



2018 projet du master (16/20) :  *English Linear System Solver*

Implémentation d'un solveur de système linéaire en langue naturelle.

<https://bitbucket.org/OPiMedia/english-linear-system-solver-in-prolog>

$\LaTeX$   $\LaTeX$  — langage de composition de documents

---

2019 – ... projet personnel en cours : ensemble de paquetages  $\LaTeX$

~2005 – ... projet personnel : rédaction de documents logico-mathématiques

[www.opimedia.be/docs-math](http://www.opimedia.be/docs-math)

~ 2004 – 2014 projet personnel : ensemble de paquetages et d'outils pour  $\LaTeX$

$\LaTeX$ , PostScript, PSUtils, Makefile, Perl, Python

*“le but de l'abstraction n'est pas d'être vague, mais de créer un nouveau niveau sémantique dans lequel on peut être absolument précis.”*

(Edsger W. DIJKSTRA, *Le programmeur modeste*)

*“La maîtrise de la complexité est l'essence même de la programmation informatique.”*

(Brian KERNIGHAN et P. J. PLAUGER, *Software Tools*)

**Olivier PIRSON**

Belge, né en 1975


 [www.linkedin.com/in/olivierpirson](http://www.linkedin.com/in/olivierpirson)

 <https://europa.eu/!CF64fw>

 [bitbucket.org/OPiMedia](https://bitbucket.org/OPiMedia)

 Bruxelles, Belgique

 [olivier.pirson.opi@gmail.com](mailto:olivier.pirson.opi@gmail.com)

 [www.opimedia.be](http://www.opimedia.be)

$$\sum_{i=0}^n f(i)$$

`sum(f(i) for i in range(n))`

— **Sciences mathématiques** (bachelor) — ... — **Sciences informatiques** (master) —

**DÉVELOPPEUR / INFORMATICIEN**