


# Stéphane Fermigier, Systematic : « le succès de Python s'est construit en 20 ans »

Le groupe thématique logiciel libre (GTLL) du pôle de compétitivité Systematic Paris-Région organise, avec le soutien de sponsors (Axa, Microsoft, Nexedi...), la **première édition française de la conférence PyData** dans les locaux de Telecom ParisTech, rue Barrault (13e), ce vendredi 3 avril 2015.

À l'occasion de [PyData Paris](#), 250 développeurs et utilisateurs d'outils d'analyse écrits en Python sont attendus. Stéphane Fermigier, président du GTLL et fondateur de l'éditeur Abilian, précise les objectifs de cette journée et met en lumière les atouts du langage de programmation dans le Big Data.

## **Silicon.fr : Pourquoi Python s'est imposé dans le traitement de données et le calcul scientifique ?**

**Stéphane Fermigier** : Python est un langage de haut niveau, similaire à bien des égards à ce  que proposent les plateformes propriétaires présentes dans ce domaine, comme par exemple Matlab pour le calcul scientifique ou SAS pour les statistiques. En ce sens, Python permet d'exprimer, avec une syntaxe familière, les concepts mathématiques que manipulent les pratiquants du calcul scientifique et de l'**ingénierie des données**. Mais c'est aussi, contrairement à ces langages spécialisés, un langage généraliste, utilisé largement – Python est cité parmi les cinq langages les plus populaires du moment – dans différents domaines, dont le Web et les applications d'entreprise. De ce fait, Python dispose de capacités d'intégration supérieures, notamment avec les bases de données (SQL ou NoSQL) et les services web, y compris le web crawling et le web scraping ou encore les frameworks de calcul distribué.

Notons également que le succès de Python s'est construit en vingt ans, de manière organique, selon une démarche globale d'innovation ouverte et incrémentale. Et en trois phases. Dans un premier temps, c'est le côté « langage de script » qui a été utilisé principalement : l'idée de départ était d'utiliser Python comme une interface interactive au-dessus de **bibliothèques scientifiques** écrites en Fortran, en C ou en C++. C'est le type d'utilisation que j'avais personnellement à la fin de ma thèse en 1996-1999. Dans un deuxième temps, l'introduction de la bibliothèque NumPy en 2006 (qui elle-même reprenait et harmonisait des développements réalisés au cours de la décade précédente) a permis de fournir un socle commun de types de données et de primitives numériques et matricielles, à un écosystème dont la croissance s'est accélérée. Et ce avec des projets comme SciPy, un ensemble de bibliothèques scientifiques, **scikit-learn** pour le machine learning, scikit-image pour le traitement d'images, Pandas pour les statistiques, matplotlib pour la visualisation, etc. À ces projets viennent s'ajouter des bibliothèques comme SymPy pour le calcul symbolique, SimPy pour la simulation, et des projets d'intégration comme Sage ou IPython, le « cahier d'expériences interactif » qui est devenu un standard pour la publication de résultats scientifiques sous une forme répétable par les autres chercheurs, ou pour les travaux dirigés des étudiants.

Enfin, dans une dernière phase, on a vu l'émergence et la maturation d'efforts pour permettre aux

personnes qui implémentent les algorithmes de **calcul à haute performance**, et pas seulement à ceux qui les emploient, d'utiliser directement Python pour cela. Et ce en proposant des extensions compilées du langage Python (Cython, Pythran...) ou des techniques de compilation dynamique (Numba, Theano...) qui permettent d'atteindre, en Python ou une variante proche et interoperable, des performances similaires à du code écrit en C ou en Fortran. Ces outils prennent aussi parfois en charge les GPU, permettant des gains de performance encore plus spectaculaires sur certains types de calculs, notamment les réseaux de neurones (deep learning).

### **Que peuvent attendre les développeurs et les data scientists de la conférence PyData Paris ?**

Cette première édition de PyData Paris s'inscrit dans la continuité des conférences PyData à travers le monde. Elles ont commencé en 2012 aux États-Unis. On retrouve donc à PyData Paris les mêmes fondamentaux, à savoir : rapprocher les différents membres de la **communauté PyData**, notamment les créateurs des outils et leurs utilisateurs. Comme il s'agit d'une première en France, nous avons aussi souhaité proposer des présentations accessibles au plus grand nombre, notamment aux développeurs Python qui ne pratiquent pas encore la data science et aux data scientists qui ne sont pas familiers de Python et de l'écosystème PyData. Nous mettons également l'accent sur les **retours d'expérience** relatifs à l'utilisation de ces outils dans l'industrie, le commerce et les services. Globalement, on peut dire que nous proposons un tiers de présentations introductives, un tiers de présentations techniques plus avancées et un tiers de retours d'expérience. Ce qui devrait satisfaire aussi bien les débutants que les utilisateurs confirmés des outils PyData.

### **100 euros le ticket d'entrée pour un événement organisé avec le soutien de généreux sponsors, c'est beaucoup, non ?**

La question des modèles économiques et de la fixation des prix est passionnante en général, et encore plus dans le monde **Open Source**. Il y avait pas mal d'inconnues au départ pour cette première édition de PyData à Paris, et plusieurs critères ont guidé notre budget. Nous avons d'abord constaté que le chiffre de 100 euros par jour est assez courant pour des conférences similaires, qu'elles soient monotrack ou bitrack, à Paris et avec des intervenants internationaux qu'il convient de défrayer. De plus, nous appliquons des réductions substantielles pour les jeunes pousses, les académiques, et à plus forte raison les étudiants. Par ailleurs, il nous a semblé sain d'équilibrer le budget, et donc de distribuer les risques entre le sponsoring et la billetterie. Enfin, PyData Paris est organisé par une association à but non lucratif, le pôle de compétitivité Systematic. Les excédents éventuels dégagés par l'événement iront au soutien ou à la promotion des projets PyData.

À l'arrivée, nous serons 250 participants ce vendredi 3 avril 2015, ce qui est proche de la capacité maximale d'accueil du lieu de la conférence, et constitue un gros succès pour nous. Nous pouvons donc considérer qu'il y a un véritable engouement autour de la formule que nous proposons – comme pour les autres conférences PyData dans le monde –, ce qui valide notre approche et nous incite à recommencer l'an prochain, probablement sur deux ou trois jours au lieu d'un.

### **Quels projets industriels de développement « PyData » sont mis en avant ?**

Parmi les nombreuses interventions de type « retour d'expérience » qui auront lieu pendant la

conférence, citons, par exemple, celles d'Airbus et de Nexedi dans le domaine de la **maintenance prédictive**, soit l'utilisation du Big Data et du Machine Learning pour prévoir les pannes sur des équipements avant qu'elles ne se produisent. D'autres intervenants « industriels », au sens large, témoignent aussi de la richesse des applications actuelles de PyData : la prédiction de trafic pour les VTC (Lyft), le e-commerce (PriceMinister), la vente prédictive (Data Publica), la transparence du marché de l'énergie (Kpler) ou encore l'assurance (Axa).

**Lire aussi :**

[Java, C et Python se disputent la tête des langages de programmation](#)

[Salaire des développeurs : maîtriser Ruby ou Python paie !](#)

[« Big Data, après-PC et qualité logicielle sont les défis du logiciel libre »](#)

**crédit photo de une © agsandrew - Shutterstock**