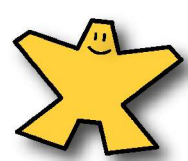


> Python for middleware scripting



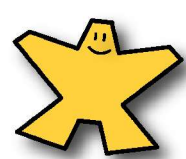
Stefane Fermigier
<sf@nuxeo.com>

2005/06/21



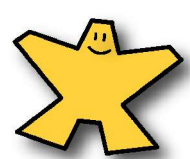
NO to software patents in EU!





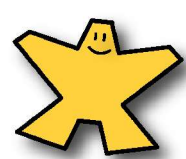
Who I am

- ◆ Python programmer since 1995
- ◆ Contributor to Python and Zope since 1998
- ◆ Founder and CEO of Nuxeo since 2000
 - ◆ 25 persons company with offices in Paris, Grenoble, Bucharest
 - ◆ Specialists of Enterprise Content Management and Collaborative Work
 - ◆ Developers of CPS, open source platform for ECM based on Zope
 - ◆ Clients: majors french and international administrations and corporations



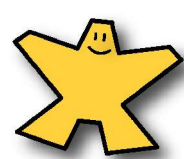
Python – an agile programming

- ◆ (Very) High level language
 - ◆ Expressivity
 - ◆ Code lisibility
 - ◆ Optimised for development time more than execution time
- ◆ Dynamic typing
 - ◆ AKA Duck typing (« if it quacks like a duck, then it's a duck »)
 - ◆ Benefits: flexibility and reuse
- ◆ Balance between simplicity and power
- ◆ Multi-platforms



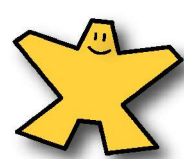
Implementations

- ◆ CPython
 - ◆ Reference implementation
 - ◆ Current version: 2.4.1
 - ◆ Interfaces to many systems libraries
 - ◆ Several O'Reilly books
- ◆ Jython
 - ◆ Python on the JVM
 - ◆ Natural interface to Java libraries
 - ◆ Started in 1998
 - ◆ Slightly lagging behind the language specs (v. 2.1)
 - ◆ One O'Reilly book
 - ◆ Could use more volunteers



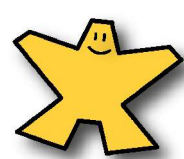
Implementations (II)

- ◆ IronPython
 - ◆ .NET and Mono
 - ◆ Fast! (up to 1.7x faster than CPython)
 - ◆ Development financed by Microsoft
 - ◆ Status: pre-alpha
- ◆ PyPy
 - ◆ Python in Python
 - ◆ SLOW (up to 1000 slower than CPython)
 - ◆ Useful for research in the language, its semantic and execution
 - ◆ Research program financed by the EC
 - ◆ « Most interesting approach » according to GvR !



Python, what for ?

- ◆ System or complex applications scripting
 - ◆ Tools to generate C, C++, Fortran, Java, .NET, ObjectiveC... interfaces
 - ◆ Examples: SWIG, SIP, Boost.Python, CXX, Pyrex, JPytype...
- ◆ Uses
 - ◆ « Shell-like » scripting
 - ◆ Automatic testing
 - ◆ Loose coupling of components



Python, what for? (II)

- ◆ Rapid applications prototyping
 - ◆ Productivity between 3x and 10x than non-agile languages (says Bruce Eckel)
 - ◆ Good if time to market is critical or for evolutionary development
 - ◆ Sometimes critical parts have to be rewritten in a non-agile programming language (C, C++, Java...)
- ◆ Full-features large-scale applications
 - ◆ Ex: CPS, ERP5, Chandler...



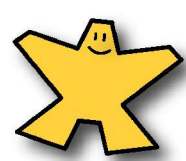
A clean and extensible object model

- ◆ The object model « map well » to the objects models of common languages like C++, Java or ObjectiveC
 - ◆ Easy to script and sometimes extend classes in Python
- ◆ Easy to bind procedural libraries (from C for instance) and make them « pythonic » (object oriented)
- ◆ Supports meta-programming
 - ◆ Possible extensions: AOP, active objects...



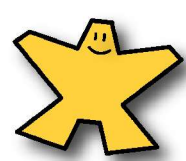
A rich standard library

- ◆ OS API (POSIX...)
- ◆ Strings (including Unicode), regular expressions
- ◆ Network protocols
 - ◆ http, ftp, pop, imap, smtp, news...
- ◆ XML
 - ◆ SAX, DOM
- ◆ Debugger, profiler, unit tests, doc generators
- ◆ ...



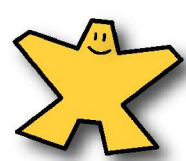
Third-party libraries - databases

- ◆ Associatives
 - ◆ dbm, gdbm, Berkeley DB...
- ◆ Relational – SQL
 - ◆ MySQL, PostgreSQL, Oracle, Sybase...
- ◆ XML
 - ◆ Berkeley DB XML, Tamino, Xindice
- ◆ Object
 - ◆ ZODB
- ◆ Object-relational mappings
 - ◆ Several projects (APE, SQLObjects, Modeling...)



Third-party libraries – web templating

- ◆ DTML (Zope)
- ◆ ZPT (Zope)
- ◆ Cheetah
- ◆ Nevow (Divmod Quotient)
- ◆ Woven (Twisted)
- ◆ PSP (Webware)
- ◆ ClearSilver bindings
- ◆ ...
- ◆ All try to find the right balance between too much and too little logic



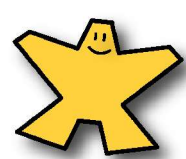
Third-party libraries – GUI

- ◆ Tkinter (Tk)
- ◆ PyGTK / PyGNOME
- ◆ WxPython (WxWidgets)
- ◆ PyQt / PyKDE
- ◆ Bindings for the MacOS X GUI
- ◆ Bindings for the Windows GUI
- ◆ Swing, SWT (via Jython or natively)
- ◆ ...



Third-party libraries – distributed computing

- ◆ Web services
 - ◆ XML-RPC
 - ◆ SOAP
- ◆ CORBA
 - ◆ OMG-approved Python bindings specs
 - ◆ Implementations: OmniORB, Fnord, ORBit-python
- ◆ Python-specific protocols



Third-party librairies – full frameworks

- ◆ Twisted
 - ◆ Event-driven library to implement network protocols
 - ◆ Ex: conch, an SSH implementation in 7000 LOC
 - ◆ Provides also a web application framework
- ◆ PEAK
 - ◆ Framework for components assembly, similar to J2EE
- ◆ Webware, SkunkWeb, Quixote, CherryPy, Paste, Subway, Aquarium...
 - ◆ Web applications frameworks



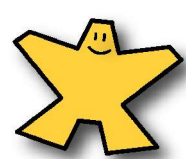
Third-party librairies – full frameworks

- ◆ Zope
 - ◆ Web applications and content management
 - ◆ « Object publication environnement » = application server + object database + services
 - ◆ Many users and applications
- ◆ Zope 3
 - ◆ Major rewrite started in 2001
 - ◆ Same object database, templating language, similar object publishing mechanism
 - ◆ Easier to configure, to write tests, to integrate code from other frameworks



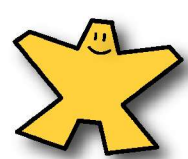
Third-party libraries - text processing

- ◆ XML
 - ◆ Many available libraries and toolkits
 - ◆ Read « The state of Python XML » by Uche Ogbuji on oreillynnet.
- ◆ Parser generators
 - ◆ Spark, PLY, Yapps, Dparser, PyBison, pyparsing...
- ◆ Natural language processing
 - ◆ Natural Language Toolkit (NLTK)



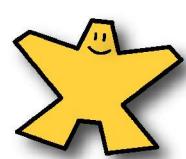
Development tools – IDE

- ◆ Proprietary
 - ◆ Wing IDE
 - ◆ ActiveState Komodo (IDE)
 - ◆ ActiveState VisualPython (plugin for MS-Visual Studio)
 - ◆ Tru Studio (Eclipse plugin)
- ◆ Open source
 - ◆ IDLE (Tk)
 - ◆ Boa Constructor (WxWindows)
 - ◆ Eric (Qt/KDE)
 - ◆ DrPython (inspired par DrScheme)
 - ◆ Several Eclipse plugins (for either CPython or Jython): pydev, Red Robin...



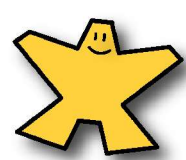
Development tools – testing

- ◆ Automated testing is a fundamental part of an agile development process
- ◆ Libraries
 - ◆ PyUnit (standard)
 - ◆ Doctest (tests are written as « stories » and are also used as documentation)
 - ◆ Others
- ◆ Python is the scripting language of several web testing application
 - ◆ ParaSoft WebKing, MaxQ, PushToTest...
- ◆ Python tests shipped with several complex applications (ex: Subversion)



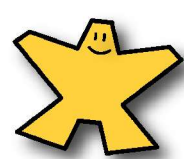
Python as an Open Source project

- ◆ Project started in 1990 by Guido van Rossum
- ◆ Open Source BSD-style licence
- ◆ Development team: BDFL (GvR), + core team + hundreds of contributors
- ◆ Language evolution is controlled by the PEPs (*Python Enhancement Proposals*)
- ◆ Current rythm: one release / year
- ◆ Python 3 (or 3000): major evolution, incompatible, no financing for now



Users and supporters

- ◆ Google (extensive internal use, has released some open source libraries)
- ◆ Apple (Python included in Mac OS 10.2)
- ◆ Microsoft (supports IronPython)
- ◆ Red Hat (uses Python extensively for its installer and admin tools)
- ◆ OpenOffice.org 2.0, Mozilla 2.0
- ◆ NASA, EDF R&D, CEA
- ◆ ILM (CGI, ex: Shrek)
- ◆ ...
- ◆ (Références: www.pythonology.com)



Import this !

```
fermigier@x10.nuxeo.com: /home/fermigier/cvs/python/python - Terminal - Konsol
Session  Édition  Affichage  Signets  Configuration  Aide

[fermigier@x10 python]$ python
Python 2.3.4 (#2, Aug 19 2004, 15:49:40)
[GCC 3.4.1 (Mandrakelinux (Alpha 3.4.1-3mdk))] on linux
Type "help", "copyright", "credits" or "license" for
>>> import this
The Zen of Python, by Tim Peters

Beautiful is better than ugly.
Explicit is better than implicit.
Simple is better than complex.
Complex is better than complicated.
Flat is better than nested.
Sparse is better than dense.
Readability counts.
Special cases aren't special enough to break the rule
```