

python™

a programming language
changes the world

Content is ready...

NOW we need the final support by additional sponsors to produce and distribute.

Ad Sponsorship

The production of the brochure will be refinanced using sponsorship ads.

Don't miss your chance to present your company in the first Python image brochure – space is limited!

Reference Entry

With the reference entry sponsorship you can support the production too!

Don't miss your chance to present your company in the first Python image brochure – space is limited!

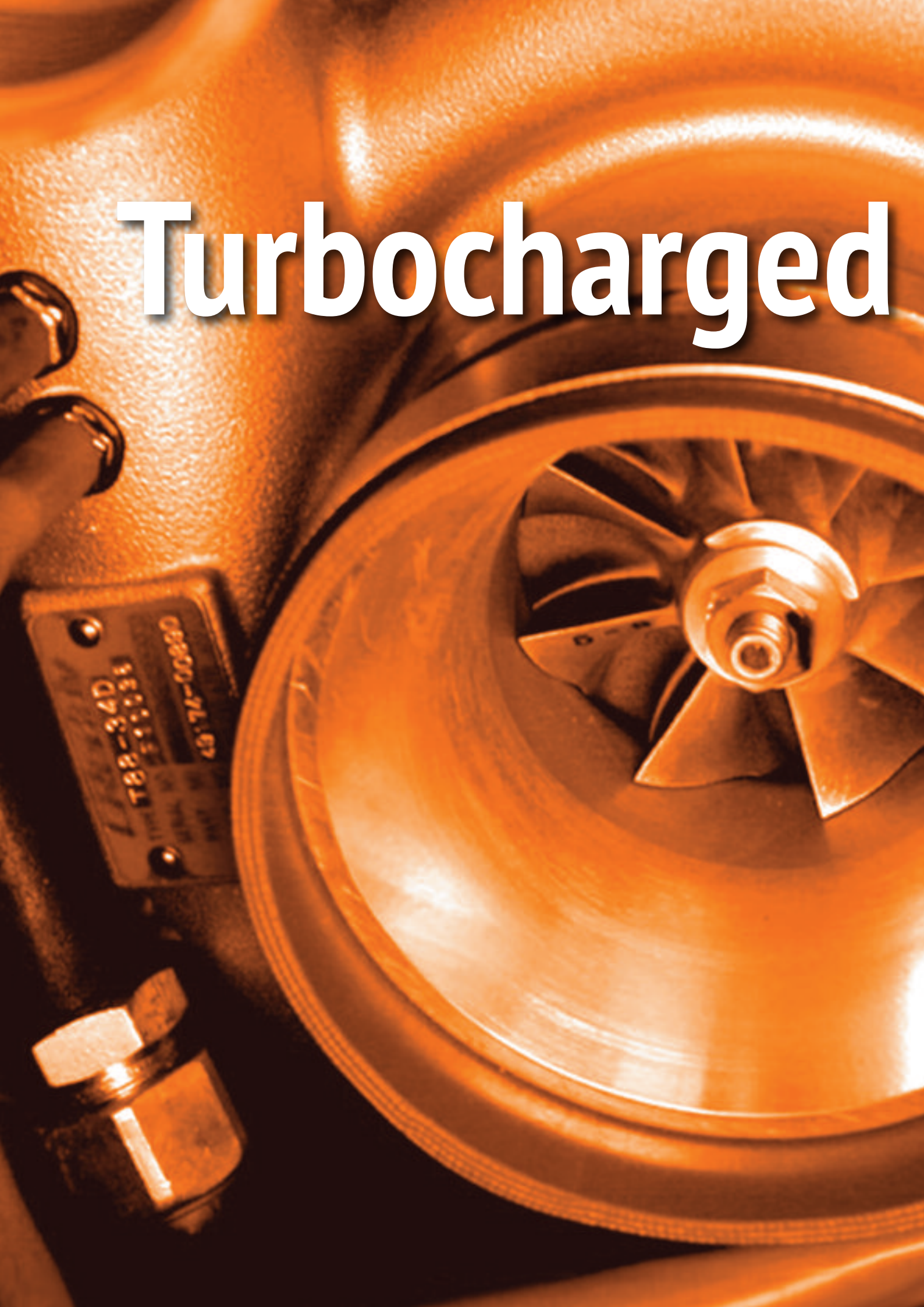
Final Production Content

Case Studies & Success Stories

Vol. I



Turbocharged



Programming with Python

Software quality is a vital ingredient to success in industry and science. Ubiquitous IT systems control the business processes of the global economy. Increasingly powerful computers and sophisticated algorithms provide the platform for new scientific discoveries. And global communication is inconceivable without intelligent software.

In the race for customers, the pole position belongs to those who get to market faster than their competitors. Better and more creative solutions combined with the ability to respond instantly to new challenges drive the race. Writing secure and reliable programs in a fraction of the time normally required gets you first across the finish line.

The Programming Language of the Agile Era

Agility is the hallmark of our times and Python is the programming language of the agile era. The Python universal programming language is the turbocharger of the IT department. Compared with other modern programming languages such as Java or C, Python achieves superior results in significantly shorter timescales for a number of different reasons.

For example, Python is a very lean programming language. Python programs are a great deal shorter than code written in other modern programming languages. As a result, both development times and maintenance costs are drastically reduced. Less code means fewer errors, meaning the cost of identifying and eliminating these errors is also reduced.

A comprehensive standard library and thousands of additional libraries in the Python Package Index provide developers with high-quality solutions which they can easily integrate into their applications to meet virtually any requirement.

In this way, Python frees up vast resources, which can be earmarked for more productive use elsewhere.

The Master Key for System Integration

Python offers unique benefits for system integration. On the one hand, there are huge numbers of Python libraries, with which virtually any third-party system can be integrated. On the other hand, the libraries of many other programming languages can also be used in Python.

Once they have been programmed, Python applications can run on all operating systems for which a Python interpreter exists, significantly reducing the cost of operating-system-specific applications.

The Language that has Changed Everything

For over 20 years, Python has been used successfully throughout the world as a programming language in industry, in the service sector, and also in research and science to meet a wide range of different requirements. In this time, the language has changed many things.

The Python programming language is easy to learn. It has blurred the boundaries between users and developers. Increasing numbers of scientists, engineers, financial experts and others with little programming experience are using Python to solve specific complex technical problems.

The Web

Python impresses Mozilla



Tarek Ziadé – member of the Mozilla Service Team, on the reasons for success.

The Mozilla Corporation, manufacturer of the Firefox web browser and Thunderbird e-mail client, is impressed with Python and uses the language for its web services. The addon.mozilla.org and support.mozilla.com websites and the Socorro crash-reporting system are based on Django and Python.

Firefox Sync Server

The Firefox Sync Server, which is used to synchronize bookmarks, browsing histories, passwords, and open tabs on different computers and mobile devices, was also written in Python.

«The Python programming language supports many programming paradigms and can be put to productive use virtually anywhere. What's more, Python is not restricted to the web. For example, we also use Python for our packaging and build systems.

The Python ecosystem is very rich and well-developed. Our developers can incorporate existing libraries into their projects and only need to develop the new functions that they need.

Python's concise syntax is simple and yet highly productive. This means that new developers can very quickly get involved in our projects, even if they are not yet familiar with Python.»

Keep Your Firefox
IN SYNC

Seamlessly sync your Firefoxes, access your data no matter which device you use.

Learn more about Firefox Sync

Have a question about Firefox Sync? Ask it in our forums.

Just follow these SIMPLE STEPS:

1. Go to your Firefox Preferences and click the 'Sync' tab.
(Don't see where your preferences panel is?)
2. Click 'Set Up Firefox Sync' and 'Create a New Account'.
3. Enter your email address, create a password and confirm it. Then check "I agree to the Terms of Service and Privacy Policy" and click Continue.
4. Confirm that you are not a robot by entering the text from the image.
5. Click 'Done' to close the setup window and take you back to the Sync panel.
6. You're done! Firefox Sync will now automatically sync your data in the background from here on.

Want more detailed instructions?

STAY SAFE
Firefox Sync works behind the scenes to keep you safe.

For Your Eyes Only:
Your data is always protected, so only you have access to information like your passwords and browsing history.

Safety First:
Firefox Sync is the only service with end-to-end encryption. Sync's advanced security measures mean you're never vulnerable to online bad guys or companies that will sell your information.

Get Up and Go:
Enjoy the benefits of sharing personal info across all your devices while still maintaining military-grade security and privacy.

Get Firefox for Mobile
Or visit Firefox.com/m on your mobile browser

>>> import success

Less Code is Better

A task that requires an average of twenty lines of code in C and seven in Java can often be done with just one line in Python.



«Python has been an important part of Google since the beginning and remains so as the system grows and evolves. Today dozens of Google engineers use Python, and we're looking for more people with skills in this language.»

Peter Norvig – Director of Research at Google Inc

Powered by Python

In 1998, a small company from California, with the help of the Python programming language, changed the way we search for relevant information on the Internet. The company had an unusual-sounding name: Google.

Twelve years on, this name is valued at 111 billion* US dollars, making it one of the most valuable brands in the world. Thanks to its innovative products, Google has quickly risen to its position as a market-leading online information provider.

«Google's mission is to organize the world's information and make it universally accessible and useful.»

It is safe to assume that without Python, the World Wide Web would not exist in its present form. Not only Google, but also many other companies and open-source communities have changed the web, and with it our lives, using Python.

*Millward Brown BrandZ Study 2011

Tailor-made for Web Applications

The Python programming language is tailor-made to perform new tasks on the World Wide Web. Frameworks and application servers such as Zope, Django, Turbogears and Pylons have enabled small start-ups to implement new services and business models on the Internet. Quickly, securely and with built-in scalability.



«Python is fast enough for our site and allows us to produce maintainable features in record times, with a minimum of developers.»

Cuong Do – Software Architect, YouTube.com

It is easy enough to work out what this means for a code base consisting of several thousand lines of code. Huge amounts of time can be saved in the development and long-term maintenance of Python programs. With every line of code that can be left out, a possible source of error is also avoided.

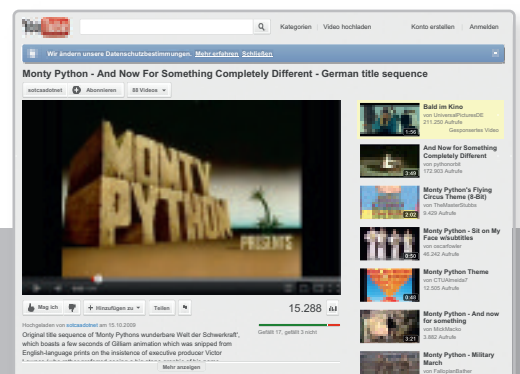


How a Shortener goes Large

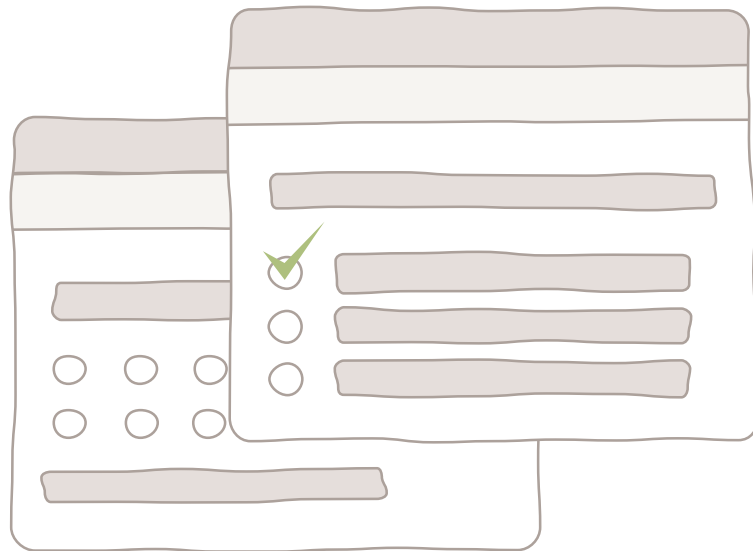
Bitly is a URL shortener and analytics service based in New York City, handling hundreds of millions of events per day. The majority of the code is written in Python. Python scales well and means that the shortener really can do it large.

Bitly shortens approximately 80 million new URLs per day and handles hundreds of millions of clicks on those URLs.

Bitly chose Python for its ability to develop rapidly, and for its vast standard library. The company also makes extensive use of Python's numerical computing frameworks. Python serves all user-facing pages as well as a large portion of public API traffic. Additionally, Python does the heavy lifting for all asynchronous work in the backend systems. All of bitly's internal APIs are written or prototyped in Python.



“New developers learn to use the leaner code base much more quickly.”



From C# .NET to Python

How SurveyMonkey attracts and retains Top Developers

SurveyMonkey is the world's largest survey company, with over one million survey responses every day on its website. The company has built tools based on over 10 years of experience in survey methodology and web development. Following an acquisition in 2009 it has grown to over 100 employees throughout the United States and Portugal, with its headquarters in Palo Alto, CA.

At the time of the acquisition, the company was running a single monolithic C# .NET web application. While the system was running smoothly, it was slow to add, test, and deploy new features. SurveyMonkey decided to bite the bullet and commit to rewriting the application in Python, breaking core features into separate services.

Separate Services

The services communicate via hardened web APIs. This decomposition makes it possible to build features on smaller codebases that are more easily manageable and can be released on independent schedules.



SurveyMonkey®

«The transition to Python from .NET has been a great success. Our Python applications handle heavy traffic without problems. Developers have found it possible to add new features much more quickly than before. New employees get up to speed in a manner of days, and recruiting has become easier.»

Chuck Groom – Head of Engineering
at the Seattle SurveyMonkey office

```
>>> import success
```

Lively Open Source

Python is used by a large dynamic community.

The combined creativity and productivity of the community and the intensive transfer of knowledge and expertise within the community makes it an extended workbench of a company's own IT department.



«Esri users range from veterans who have been using GIS through several generations of our platform to young people educated in the latest software approaches. Using Python as our core scripting language lets us reach everyone, whether to build a simple repeatable workflow or conduct complex GIS analysis.»

Jack Dangermond – President, Esri

Esri and ArcGIS

Mapping Knowledge with Python

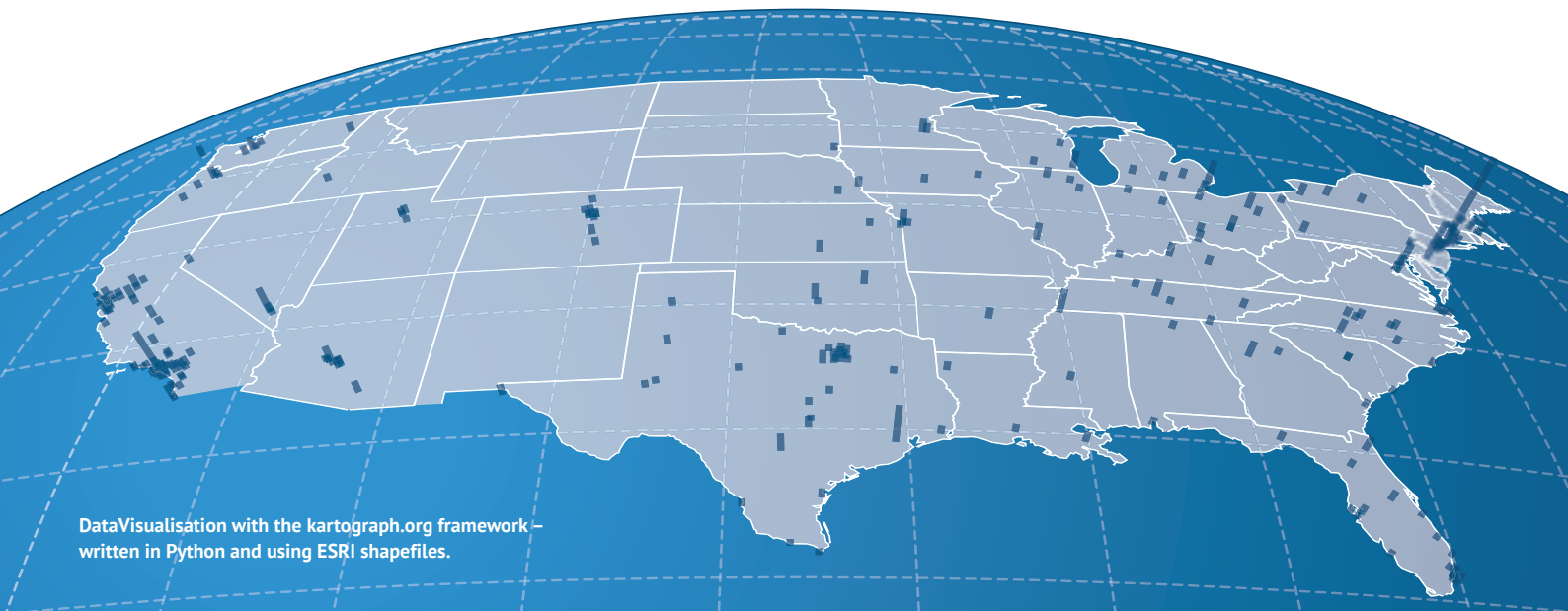
Scientists, thought leaders, and innovators, including commercial and government workforces, are increasingly sharing their knowledge using geographic information systems (GIS) to better understand problems and make better decisions.

To help their users, Esri, leading provider of GIS solutions, created a Python module for their ArcGIS system called ArcPy. Python was chosen for its ease of use, extensibility, readability, openness and power. Now, professionals and programmers from nearly any discipline can incorporate their work and

methods into ArcGIS. Python has become the language for data analysis, conversion, and management, which in turn is increasing the productivity of ArcGIS users, as well expanding the use of ArcGIS into new markets.

Increasing Productivity

GIS application developers also benefit from ArcPy. They can rapidly prototype and test scripts in an interactive environment, scale applications to meet demand, and port applications across platforms.



Data Visualisation with the kartograph.org framework – written in Python and using ESRI shapefiles.

Solutions to many problems have already been devised by other developers. Extensive online documentation, numerous books, forums, and mailing lists, and also congresses and conferences across the globe make it easy to learn this fascinating programming language.

“The Python community... an extended workbench of a company's own IT department...”

OpenERP

It's all about Agility

OpenERP is a full-featured enterprise resource planning suite written in Python. One of the key strengths of OpenERP is that it is agile at all levels: customizations, modularity, and quick start for newcomers.

These characteristics are beneficial for both customers and developers. Customers need agility to react to changing business needs immediately, and OpenERP developers need agility to drive the software forward and stay ahead of the competition. All of this is possible because of the Python framework upon which OpenERP is based.

Fast-tracking New Professionals to Productivity

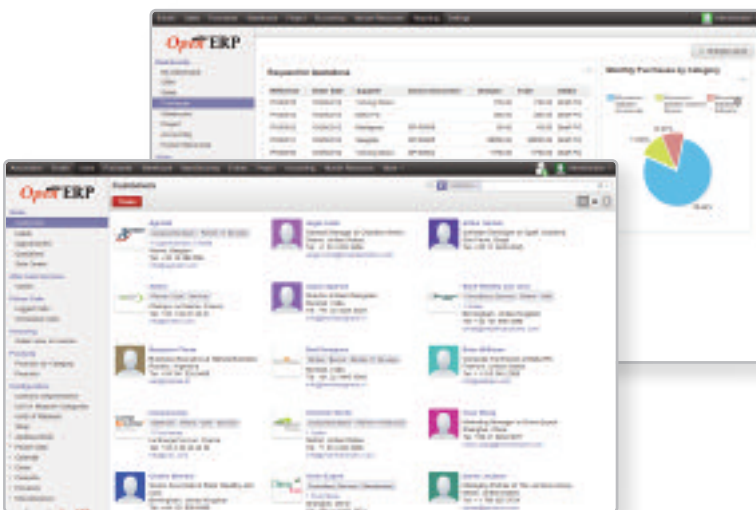
Due to the fast growth and high demand for OpenERP, the company has to train integration partners every day, so that they can become proficient and start customer implementations in just one week! Most of them don't have advanced knowledge of Python when they start, but in just a few hours of training, they can have a new OpenERP module up and running.



«Python played a strategic role in the development of OpenERP. The flexibility of Python gave us the capability to adapt to market or design decision changes over the years. It's very important for large projects as the customer demand evolves very quickly: five years ago, the best ERPs were three-tier applications with a rich user interface; today you need a web interface to be on the top; tomorrow you will need to be on mobile or tablet PCs.»

Most of our competitors did not succeed in evolving with the market and this is what makes OpenERP a killer application: we have all the features of seven years of development application but with the user interface of recent and cool applications.»

Fabien Pinckaers – OpenERP CEO



OpenERP developers need agility to drive the software forward and stay ahead of the competition.

>>> import success

Agile Development

Python is the language of agile software development.



eGenix.com

«We wouldn't have been able to do this with any other technology. Only Python with its extremely high efficiency, clear syntax and large number of professional quality extensions and frameworks made this possible.»

eGenix.com

VAT Revenue Collection in Ghana

Marc-André Lemburg
CEO eGenix.com

Ghana is one of the few countries in Western Africa with a growing economy. Value added tax (VAT) is a major source of revenue for the government of Ghana.

Unfortunately, collecting VAT is difficult: Many small traders have trouble calculating and reporting the VAT tax on their sold products.

A Swiss company, Aya Technologies AG, had an idea of how to improve this situation: instead of letting the traders do the VAT calculation and reporting themselves, they suggested having the traders use an online terminal, which works as simple electronic cash register with integrated book-keeping, VAT calculation, and reporting. To convince the Ghana VAT Service, they agreed to run a pilot project.

Since the project was time-critical, Aya contacted eGenix.com, a German consulting company, to get the backend server for the system implemented. eGenix designed and rolled out a complete end-to-end system based on PostgreSQL and Python, communicating with the trader terminals, managing trader sales in the database and providing live monitoring via a web interface to the VAT Service – all in just three months!

The Ghana VAT Service was immediately sold on the idea and is now using the system to improve VAT revenue for traders in their small traders program.

Since Python is an interpreted language, the entire compilation stage is eliminated from the development process, which normally entails three working steps (programming, compiling, debugging). Optimization errors by the compiler, resulting in crashes which are difficult to analyze, are similarly ruled out.

Furthermore, programming errors do not necessarily result in crashes. A comprehensive log is created for each exception, which significantly reduces error analysis costs.

In addition, Python is designed to ensure that certain errors cannot even occur in the first place.

“Competitive programs offering maximum benefit to the customer can be created very rapidly.”

Python therefore opens up entirely new dimensions in agile development. Prototypes can be developed much more quickly and can be adapted in real time to user requirements.



ForecastWatch

Yesterday's Weather

Tomorrow we will know how accurate yesterday's weather forecast was. By then, however, no one is interested anymore. Or are they?

Improved Quality with Python

Meteorologists would of course like to compare their forecasts with the weather that actually occurred, i.e., with the real measurement data, in order to optimize their calculation models and improve the quality of their forecasts. In the USA, Canada and globally, Python and the web-based system ForecastWatch are used for this purpose.

ForecastWatch is used by The Weather Channel and The Weather Network in Canada, among others. The system collects weather forecasts from a large number of national and international forecasting companies and compares them with actual observations from about 1200 locations around the world. The meteorologists can see aggregated data on temperature, precipitation, wind, and opacity forecasts, presented in a web-based view, and they can filter that data down from a nationwide summary all the way down to individual local forecasts. The entire system is written in Python.



Nachbau

«For ForecastWatch, Python's power to do a lot with very few lines of code allowed us to get the ForecastWatch prototype up in a matter of weeks. And it has allowed us to continue to use that code as the product has expanded. Having come from C++ and Java, Python's interactivity made testing code easy and made trying out new ideas or features painless and quick. This has allowed us to be nimble in meeting customer needs and to react quickly when web page formats change or new feeds are added.»

Eric Floehr – Owner of Intellovations and maker of ForecastWatch



Global Earthquake Model

Calculating the Inconceivable

Over half a million people died in the last decade due to earthquakes, most of these in the developing world, where risk is increasing due to rapid population growth and urbanization. Recent earthquakes in Haiti, Japan, New Zealand and Turkey painfully reminded the world of the destructive impact of seismic events and the importance of the availability of reliable earthquake risk information.

In many earthquake-prone regions no risk models exist to provide such information. There are also no agreed global standards for risk assessment, which are critical for the effective and unambiguous communication of seismic risk and are essential as input for increasing risk awareness and undertaking mitigating action.

For the Benefit of Society

OECD's Global Science Forum created the opportunity for an initiative to leverage scientific knowledge on earthquake risk for the benefit of

society, worldwide. In January 2009 the Global Earthquake Model (GEM) Foundation started to develop uniform standards, open software and tools for worldwide earthquake risk assessment.

Currently hundreds of organizations and individuals are working together on the OpenQuake Platform, which is powered by OpenQuake engine, an open source application written in Python which is used by scientists, engineers, government officials, and the financial sector to perform earthquake hazard and risk analysis on a global scale.



*«We find Python to be a highly productive language which facilitates rapid prototyping and the frequent adjustments needed to keep up with the latest user requirements,» says **Lars Butler**, one of the chief developers at OpenQuake. «We also find Python to be an appropriate tool for the job, given the availability of several powerful libraries which do most of the heavy lifting in our mission-critical components.»*

Python adds New Impetus to Industry

Maximal output for minimal input – the benefits of Python continue to impress more and more companies. Engineers use the programming language to make scientific calculations, design systems, or as a script language for design and simulation software. The language is also used to manage production processes.

Roll out the Future

Continuous casting plants need a lot of energy. Cost savings can be made and CO² emissions can be indirectly reduced by reducing power consumption.

```
>>> import success
```

Reusability and Scalability

Python programs can be developed in the form of modules, so that the functionality, once developed, can be reused in other programs.

This enables economical use of the resources of IT departments, saves time, and speeds up the market launch of new products. In Python's extensive standard library, developers find ready-made modules for regularly recurring programming tasks. Programs developed in modular form can therefore be quickly and easily scaled, thereby significantly increasing the flexibility of development projects.

SMS Siemag AG

Saving Energy with Python

SMS Siemag AG is one of the leading companies in metallurgical plant and rolling mill technology. In order to improve the efficiency and environmental impact of its continuous casting plants, the company is working together with its customers to devise energy-saving methods of operation, known as “Eco Mode.”

In Eco Mode, generator units not needed for production are automatically shut down or switched to a power-saving stand-by mode. Python-based software is used to measure and record the power consumption of the various consumers so that the reactive and active power of the individual generator units in the different operating modes is logged and evaluated. The most energy-efficient system control, the Eco Mode, is then determined for each operating mode from the data obtained. Eco Mode allows energy savings of up to 7%.

The agile programming language allows engineers to make real-time system modifications, collect new data during live operation, or change the evaluation criteria.



DLR German Aerospace Center Lift off with Python

The DLR is Germany's national research center for aeronautics and space and describes itself as one of Germany's leading software houses. Around 1 000 employees develop software for the numerous research and development projects run by the DLR.

The DLR mainly uses Python to develop engineering software and tools for knowledge management. The DLR also uses a Python-based system called Allura to manage its software development projects.



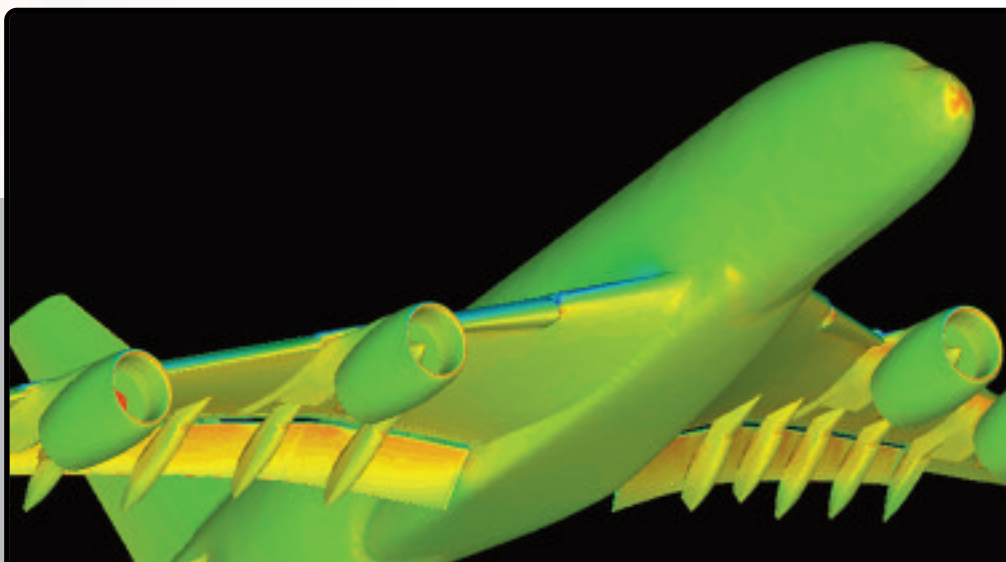
VAMPzero: Aircraft Conceptual Design with Python

VAMPzero, a Python-based software tool for the conceptual design of aircraft enables DLR to address two main challenges during the conceptual design stage for aircraft: flexibility and transparency. As in every design, requirements change during the course of development. The same is especially true for aircraft design and becomes even more important due to the integration of novel technologies. Based on well known handbook

methods the design of new configurations with VAMPzero includes outer geometry as well as structures, engines, systems, mission analysis and costs. VAMPzero supports working in multi-disciplinary and multi-fidelity environments.

VAMPzero is the first aircraft conceptual design code that is published as open source. It is available through software.dlr.de.

“Around 13 000 freely accessible libraries offer a ready-made implemented solution for virtually every conceivable task.”





Swisscom IT Services AG

Quantax Post-Trade Risk Management

Quantax addresses post-trade risk management for complex portfolios of financial instruments. It is currently used on bank trading floors and for professional asset management in four countries.



swisscom

«Python has a low technical and conceptual overhead, yet includes everything you need to build professional applications.»

Martin Gfeller – Swisscom

There are a number of applications in the risk management domain. They are typically either Excel-based solutions at the low end or solutions developed as complex, multi-layered client-server systems at the high-end, with very little in between.

Swisscom IT Services, a provider of professional consulting and expert implementations of IT solutions in the financial sector, decided to build Quantax from the ground up as an object-oriented, web-based application, using technologies in which domain experts can utilize their expertise directly. Python excels at low technical overhead

and is therefore used as the sole programming language in Quantax.

Quantax offers broad coverage of financial instruments, ranging from shares to complex credit derivatives and exotics, with real-time valuation, position keeping, and risk management. The Python environment and the Zope application server already include many components and can easily be expanded with modules provided by the Python community and ecosystem. Quantax's total cost of ownership therefore compares very favorably with its high-end competitors.

>>> import success

Security

Python is a highly stable and secure programming language.

By the end of 2011, only 125 security-related errors were detected in Python, one tenth of the errors in Java and a fraction of the possible security breaches detected to date in C.

Risk-free programming:

- No buffer overflows
- No buffer underruns
- No memory leaks
- No stack manipulations
- No memory management errors
- No access violations
- No segmentation faults

Taming bulls and bears



The world's stock markets can no longer survive without computers. Billions are traded every second in the electronic global networks.

Complex, random calculations and simulations form the basis for buy and sell orders. Financial mathematicians like to use Python for this purpose.

Specialist libraries and Python-based applications provide them with all finance-related mathematical functions in an intuitive, comprehensible format.

They can use powerful tools without having to worry about the numerous underlying programming parameters such as memory management, violations, and debugging.

Similar conditions prevail in the insurance sector. Actuaries use mathematical models to assess risks and calculate insurance premiums, all with Python.

PyTAF Python - The Adaptor Framework

Landesbank Baden-Württemberg

Landesbank Baden-Württemberg (LBBW) is the central savings bank in Baden-Württemberg, Rhineland Palatinate and Saxony, with total assets of around EUR 373 billion*.

*as of December 31, 2011

In 1999, LBBW IT Financial Markets designed a new application integration architecture based on a message-oriented middleware layer and asynchronous event-driven communications. A toolkit was needed to achieve the project objectives of technically decoupling applications, standardizing the implementation of interface programs and meeting the requirements of real-time straight-through-processing of financial markets trade data.

As no vendor solutions fulfilled all of the key requirements the decision was taken to implement an integration framework in-house: PyTAF builds upon a simple model-driven development approach based on data flow graph-oriented development of programs on an interface editor GUI, the generation of executable programs, and a rich, extensible component library.

PyTAF allows the developer to rapidly develop interface programs (adaptors) using a wide range of connectivity and data transformation components.

Python has helped to enable PyTAF with its extensive standard library, extensibility, and dynamic nature. Python is key to PyTAF's success. Since PyTAF went live in 2000 it has seen a steady increase in use, with over 100 PyTAF-based adaptors in production today, processing millions of EUROS of trade data every day.



«There are lots of other programming languages with outstanding features.

We don't know of any other programming language with features and a syntax as elegant and concise as Python, though.»

Holger Joukl, Manuela Kälberer, Rainer Kluger
Landesbank Baden-Württemberg IT Financial Markets

Due to the language design in Python, several classes of error cannot even occur in the first place. Application code, for example, cannot trigger buffer overflows, underruns or memory leaks, since the validity of all Python indices is automatically checked. As the program stack is not administered by the application code but by the Python interpreter, stack manipulations are more or less excluded. A garbage collector ensures the real-time, automated clearance of unused objects from the memory. The programmer does not have

This is why Python has become the language of choice for security-conscious developers in many sensitive fields of application.

to worry about memory management, so manual implementation errors are avoided from the outset. As there are no null pointers in Python, errors due to access violations or segmentation faults can only occur – if at all – in the interpreter itself or in a C extension.



>>> *import success*

Test-driven Development

Python supports modern test-driven development methods based on unit tests and integration tests.

Learning to Program with Python

When Guido van Rossum began to develop the Python programming language in the 1980s, he worked at the Centrum voor Wiskunde en Informatica (CWI) in Amsterdam (Netherlands) in the ABC Working Group. ABC was a programming language that had been designed to teach IT students.

Guido van Rossum took many design concepts from ABC and implemented them in Python. This may explain why Python is ideally suited for use as a teaching language in schools and universities.

Numerous textbooks have been written for children and young people, proving that even ten-year-olds can be successfully taught the basic concepts of programming.

OLPC: One Laptop per Child

To Empower the World's Poorest Children through Education

Python is involved in one of the world's most ambitious education projects, the One Laptop per Child project, which was presented by MIT Professor Nicholas Negroponte at the World Economic Summit in Davos in 2005. The aim of the initiative is described on the project website as follows: «We aim to provide each child with a rugged, low-cost, low-power, connected laptop. To this end, we have designed hardware, content and software for collaborative, joyful, and self-empowered learning. With access to this type of tool, children are engaged in their own education, and learn, share, and create together. They become connected to each other, to the world and to a brighter future.»

The successful educational outcome of the project depends entirely on the learning software provided. To make it as simple as possible for teachers and educationalists to develop learning software for the OLPC, the project has relied on Python from the outset. The Sugar user interface, specifically optimized for children, and all application programs were written in Python.

Over 2.5 million children and teachers in 42 countries have XO laptops



ExpEYES - Eyes for Science

Low Cost Experiments for Young Engineers and Scientists

ExpEYES is a portable, expandable, and affordable science laboratory, supporting a wide range of experiments from high school to post-graduate level. It combines the power available from Python for doing scientific computation and data visualization with control via a data acquisition hardware device. Mathematical analysis and visualization of experimental data are done using modules like Scipy, Matplotlib and Tkinter. GUI programs are available for around 50 experiments. ExpEYES can change the way in which science and engineering is taught and encourages the concept of learning by doing.

ExpEYES is a part of the PHOENIX project (Physics with Home-made Equipment and Innovative Experiments) started in 2004 by the Inter University Accelerator Centre (IUAC) in India. The software was initially written in C language but shifted to Python. This resulted in two major benefits: writing the GUI code became much easier, and accessing the hardware from inside the Python Interpreter made the development of new experiments much easier. The switch to Python also made training teachers easier. In just a couple of weeks teachers have a sufficient command of Python for productive work.



Doctests provide Python with a unique tool for embedding tests into the software documentation. With Python, software development can be seamlessly integrated into automated test environments.

Python therefore produces powerful software with unprecedented levels of operational security.

With Python, software testing is as simple as can be, producing powerful software with maximum operational security.

The Language of Science

In hardly any other field of application has Python been so quickly and enthusiastically embraced as in the sciences. The language's design complements the thinking of scientists. Python does not force them to think like computer scientists, but enables them to formulate scientific tasks quickly and efficiently as programmatic algorithms in Python.

Libraries for complex mathematical and statistical calculations, for data visualization in 2D and 3D, and for specific technical purposes equip scientists from all disciplines with powerful tools to achieve their goals more quickly.

Big Bang Experiment



«Most developers in the CMS experiment are physics students looking for new physics in the data. Usually they don't have any formal IT training. Python allows them to be productive from the very start and to dedicate most of their time on the research they want to carry out.»

Benedikt Hegner – CERN

Large Hadron Collider at CERN

Chasing the Higgs Boson with Python

It is the largest machine ever built: the Large Hadron Collider (LHC) at CERN. The ring-shaped particle accelerator has a circumference of 26.7 km.

In 2012 with the LHC, physicists finally found the first direct evidence for the existence of the Higgs boson, an elementary particle predicted by the Standard Model of particle physics which gives mass to other particles. The signal was observed in both, the ATLAS and CMS particle detectors.

The CMS (Compact Muon Solenoid) is one of several experiments at the LHC. The goal of the experiment is to answer a number of unresolved key questions in particle physics and in our understanding of the universe at large.

Managing Petabytes of Data

The detectors create an unprecedented amount of experimental data, in the range of multiple petabytes each year. The challenge of analyzing these results not only involves the discovery of new physics, but also has a social aspect: thousands of scientists around the world want to access and analyze these data concurrently. To make this possible, a distributed computing and data storage infrastructure was developed: the Worldwide LHC Computing Grid. More than 34 countries and more than 100 computing centers are involved in this project.

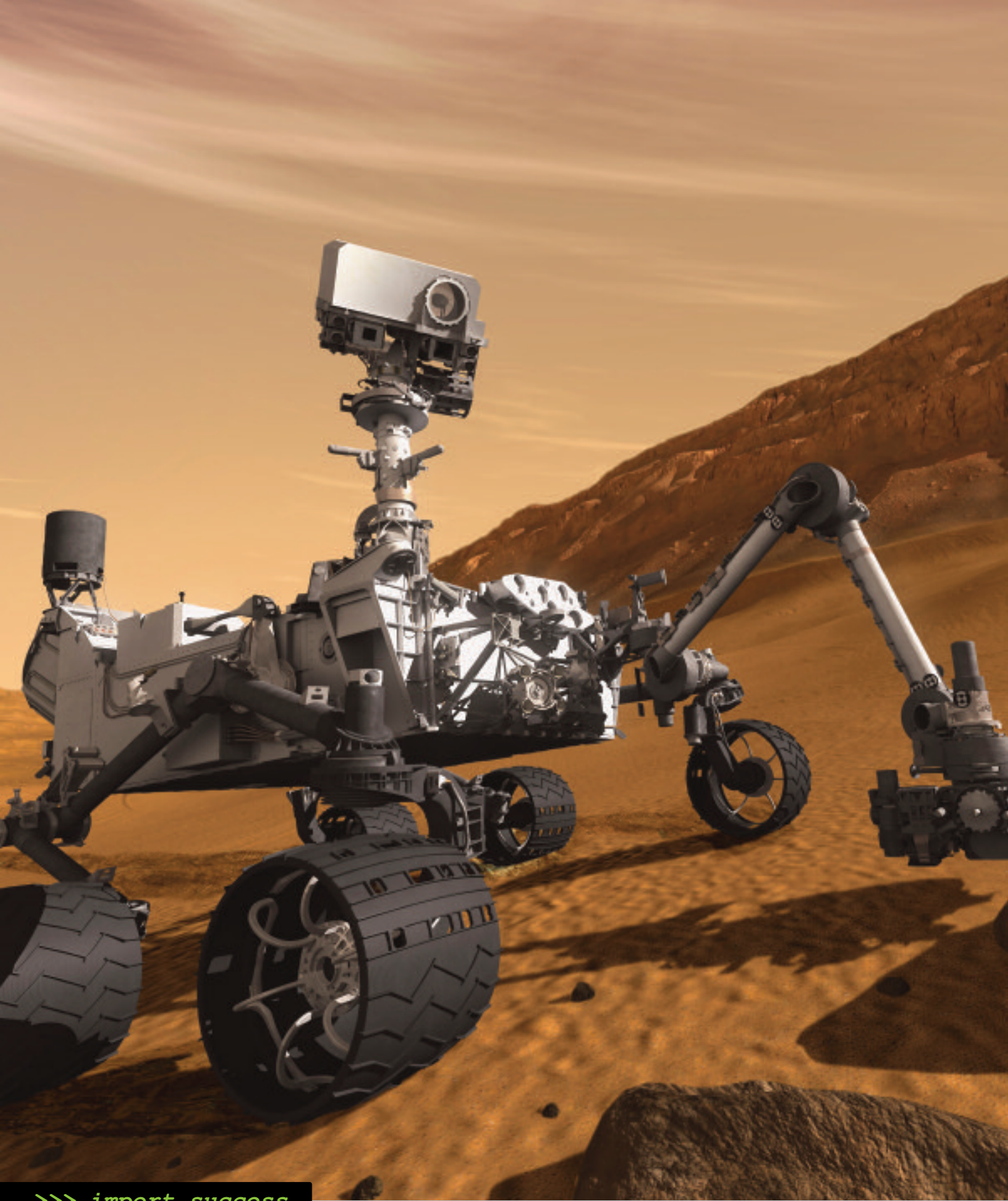
New tools needed to be written to handle the management of the data and the associated workflows, and the language of choice for these was Python. «*The large functionality available in Python's library in combination with the flexibility provided by the language itself was the main reason for the choice.*» says Dr. Andreas Pfeiffer, CMS. The full system has been up and running since late 2009 when the data collection began and has already handled a few Petabytes of data in total.

Configuring 13 000 Parameters

Every physicist is interested in different data selections and computations, and the C++ based analysis framework can be configured dynamically for this purpose. A typical analysis setup contains over 13000 configurable parameters. To help the physicists in choosing the proper parameter set, a domain specific configuration language was developed.

Soon it became apparent that the chosen configuration solution was too inflexible. Every functionality change came with a huge maintenance burden. Looking for alternatives, CERN decided to replace the domain-specific language with Python and a lightweight C++/Python bridge.

Overall, the entire process from the strategy decision, an automated translation, up to the eventual deprecation of the old configuration language was finished in less than six months. This new system has been in use for a few years now, and many problems that could not be solved using the previous system are now literally one-liners.



>>> *import success*

Openness and Interoperability

Software development is increasingly focused on the integration of multi-vendor systems.

This development has boosted Python's popularity, given that Python is more suitable than virtually any other programming language for the integration of different systems.

Python runs on all conventional operating systems and is open to databases and protocols of all types. Modules from the extensive software library give access to the hardware and software systems of a wide range of different manufacturers.

«Python, for someone who enjoys programming and with a wide-ranging taste for beautiful and powerful languages, is a winning card. It's really amazing.»

Yann Le Du – Chief Developer at the HPU4Science Project

EPRI

Looking for Life on Mars

Training a neural network to recognize biosignatures on Mars

In the year 2020, the European Space Agency (ESA) intends to send a probe to Mars which will then bring rock samples back to Earth. Back on Earth, the samples will be examined for signs of life. However, as the probe can only bring around 500 g of Mars rock back to Earth, the samples on Mars must be very carefully selected. Scientists at Chimie-ParisTech are investigating the possibility of a novel in situ selection device and they make heavy use of Python to achieve this goal.

Electron Paramagnetic Resonance Imaging spectroscopy (EPR) and imaging (EPRI) could be used to search for organic carbon embedded in rocks. This is on the discussion table at the French Space Agency (CNES). The distinctive biosignatures left behind by fossilized carbon in rocks, for example, can be readily identifiable to the EPR expert eye.

Making the Right Selection

The challenge in EPR and EPRI is to enable a Mars probe to make the right selection without human assistance.

In the Laboratoire de Chimie de la Matière Condensée in Paris (LCMCP), spectroscopists and computer scientists from the CNRS and Chimie-ParisTech are developing a new imaging pipeline algorithm for EPR and EPRI allowing automatic image reconstruction without human intervention.

Technical Details

Working closely with expert users of EPR and EPRI, the workgroup developed and coded a system based on machine learning. The involved artificial neural network is trained on the Hybrid Processing Units for Science (HPU4Science) cluster, which was assembled completely using only consumer-grade computer parts that includes graphical processors (GPUs). The goal is to obtain a network that could then be implemented in a field programmable gate array itself coupled to an EPRI spectrometer onboard a probe that could be sent to Mars in future missions.

Heavy use of Python

The scientists make heavy use of Python. It is both used for the general functioning of the HPU4Science cluster and for the computational code that runs on standard GPUs (Graphic Processing Units) through PyCUDA, a Python wrapper for NVIDIA's CUDA libraries.

However, Python code can also be embedded directly into other programming languages. IronPython provides a Python implementation for the Common Language Infrastructure (CLI), so that Python can be used as a script language for the automation of .NET applications.

Jython is a Java implementation of Python which enables all Java libraries to be imported into Python. Conversely, JavaVM can be embedded into Python to provide access from Python to Java libraries. Operating system libraries can be integrated via Python's C extensions

Python code can also be embedded directly into other programming languages.

and via SWIG, Boost, SIP, and Cython. Native hardware optimization of Python programs can be performed with the Pyco JIT compiler.



Blender

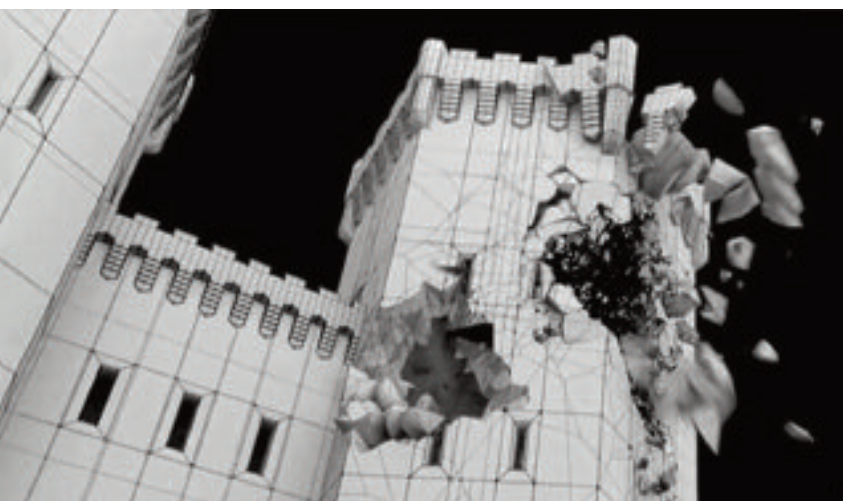
It's not just commercial software products that benefit from Python code. Python is also used in many open-source programs such as Blender to automate complex work processes efficiently.

Blender is a free open-source 3D content creation suite, available for all major operating systems under the GNU General Public License.

Side Effects Software and Houdini

Pure Magic

Side Effects Software develops Houdini, a 3D animation and VFX software package used by top studios to add CG to feature films, commercials, and games. Houdini has been in development since 1993 and was initially developed entirely in C++.



«The benefits of choosing Python continue to reveal themselves. High-level tasks that once seemed dauntingly complex can be expressed succinctly, letting us create higher quality visual effects tools. As Python use continues to grow in our industry, we have seen an explosion of customers integrating it successfully with their proprietary in-house code.»

Luke Moore

Director of Special Projects Side Effects Software

```
>>> import success
```

Python as a Scripting Language for Third-Party Systems

Due to its elegant, easy-to-learn syntax, Python can readily be used as a scripting language to control other programs.

As it then offers completely new user facilities, software manufacturers are increasingly building Python interfaces into their products.

Python is adopted widely in the film industry

Side Effects started using Python in 2005 as a replacement for their in-house scripting language. Python was chosen because they wanted a real-world, heavily used, robust language with a large set of supporting libraries. Python was also being adopted widely in the film industry and Side Effects wanted to facilitate interoperability between Houdini and other packages and libraries.

Control Houdini with Python

Side Effects therefore integrated the Python interpreter into Houdini and at the same time designed a new programming interface to control Houdini from Python. Python has become the language of choice for everything but the most performance-critical elements.

Houdini now has support for Python throughout its interface, from small expressions to whole blocks of code that set up geometric operations, physical simulations, and image processing.



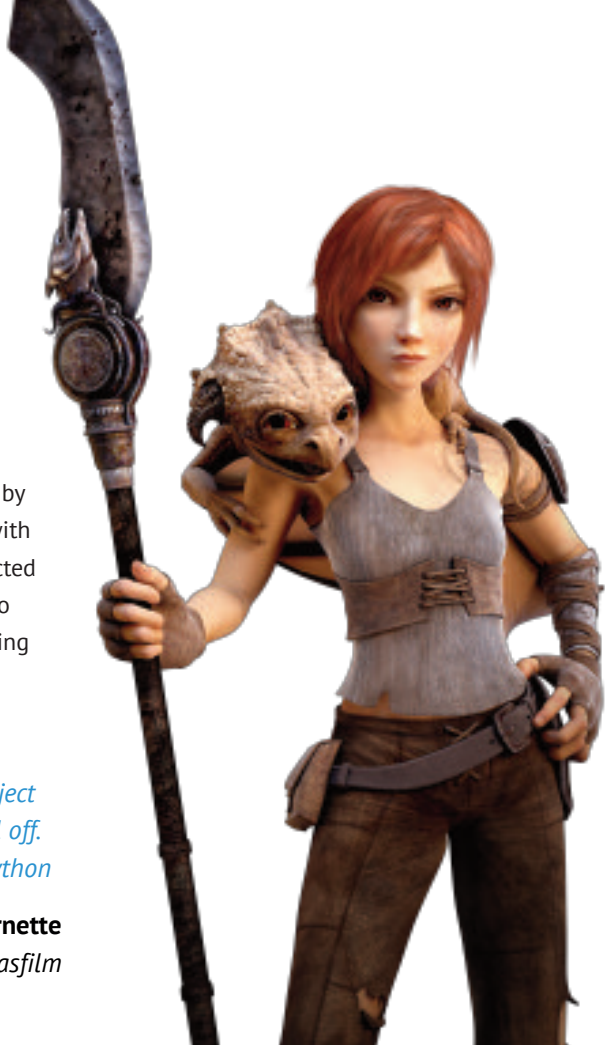
Images from ›Sintel‹, an animation film made with Blender (© Blender Foundation)

«And Action!»

Python in the director's chair

It doesn't matter whether it's Star Wars, Jurassic Park, Titanic, or Avatar – films can no longer survive today without special effects and computer animations. The Python programming language plays a key role here.

For example, at Industrial Light and Magic (ILM), a special effects company founded by George Lucas, thousands of computers with different operating systems, equipped with numerous components and commercial and in-house software, have been interconnected to form a production line which is used on a daily basis by hundreds of employees to produce computer-generated film sequences. Since the mid-1990s, ILM has been using Python to control and largely automate this extensive production process.



«Python plays a key role in our production pipeline. Without it a project the size of The Avengers would have been very difficult to pull off. From crowd rendering to batch processing to compositing, Python binds all things together.»

Tommy Burnette
Head of Global Pipeline, Lucasfilm

Furthermore, as Python is an open-source language, individual adaptations can be made to the core system. For example, Python interpreters can be developed for autonomous embedded systems or other highly specialized hardware systems.

Software manufacturers are increasingly building Python interfaces into their products.

System Admins speak Python

As an interactive scripting language, Python is ideal for system administration. System administrators are increasingly replacing their shell scripts with Python scripts, which are capable of more and easier to write.

Commercial distributors of LINUX operating systems such as Red Hat and Canonical go one step further and use Python to help them develop administrative tools for their distributions, simplifying the work of system administrators.

Managing Data Centers with Python

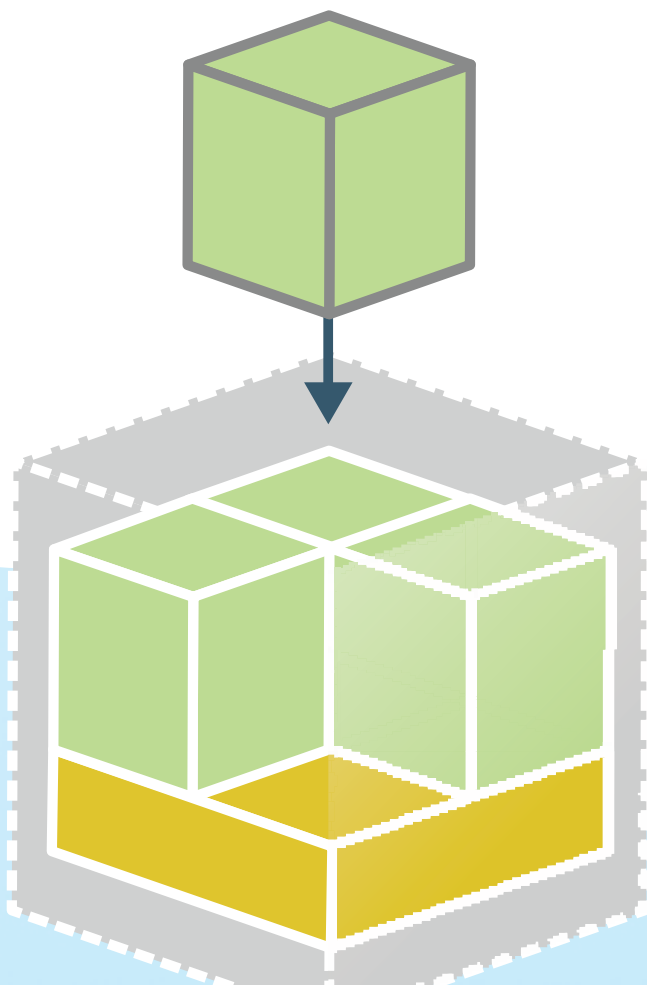
Many companies are using Python-based management programs for the efficient and secure management of data centers with hundreds of servers. No wonder that Python features prominently in the storming markets of virtualization and cloud technology.

OpenStack

Open Source Cloud Computing

OpenStack is open source software for building clouds. Created to drive industry standards, end cloud lock-in and speed cloud adoption, OpenStack has become one of the fastest growing open source projects of all time.

The open source cloud operating system enables businesses to manage compute, storage and networking resources via a self-service portal and APIs on standard hardware at massive scale.



It was at the O'Reilly OSCON 2010 conference that the open source community was first introduced to the OpenStack project, an initial collaboration between Rackspace Hosting and NASA to establish the de facto standard in cloud computing infrastructure. Since this launch over 150 companies have joined the community including such industry leaders as AT&T, Cisco, Dell, HP, IBM and Intel, and an independent Foundation was established in September 2012.

Technology

The initial release of OpenStack consisted of two primary projects: OpenStack Compute and OpenStack Object Storage. OpenStack has since grown to seven core projects spanning Compute, Storage, Networking and Shared Services, including an identity service, disk image serve and self-service dashboard.

OpenStack is architected to be flexible and pluggable, with no proprietary hardware or software requirements and the ability to integrate with legacy systems and third party technologies. The cloud software is free to download under the Apache 2.0 license, and is now included in every major Linux distribution.


```
>>> import success
```



Google App Engine

First Programming Language for Google App Engine

Python was the first programming language to be supported by Google's App Engine.

The App Engine is a virtualized programming environment which interprets Python code. The cloud provides the application on demand with the computing, storage, and network capacity resources that are currently required. The deployment of a web application is so drastically simplified that the user has no need to worry about the overall server infrastructure.

Why Python?

The OpenStack community chose Python as the exclusive development language for the project and requires all new development to be in Python as well. The founders of the community chose Python for the following reasons:

- Rapid Product Development – Leverage of a variety of pre-existing tools and components
- Infrastructure Flexibility – Python allows for internal infrastructure swapping of components with significantly less coding required
- Future Proof – The rapidly growing Python community delivers a plethora of new projects to leverage for the benefit of OpenStack
- Open Source – As an open source community, OpenStack believes in supporting other open source projects by consuming their deliverables and also giving back

«If cloud computing is the future, then understanding how to make that future open is one of the great technology challenges of our day.»

Tim O'Reilly – *oreilly.com*

Python Implementations

Various implementations and interpreters are available for the Python programming language. As a result, Python code can be executed in a wide range of different environments, which helps reduce development costs.

CPython

CPython is the Python language specification interpreter written in the C programming language. It is the most frequently used programming language implementation and is regarded as the de facto reference implementation.

Jython

Jython is a Java implementation of the Python programming language which enables Python code to be executed on Java platforms. Furthermore, all Java libraries can be imported and dynamically used.

IronPython

IronPython is a Python implementation compliant with the Common Language Infrastructure (CLI), a standard for language-neutral and platform-neutral application development and execution. IronPython is written in C# and allows Python code to be executed in CLI-compatible runtime environments such as .NET and Mono.

PyPy

PyPy is a Python implementation in RPython, a subset of the Python specification which is ported by translators into the relevant target language (C, Java, C#). A JIT compiler can be added at runtime to translate frequently used code.

Stackless Python

In Stackless Python, the context of the interpreter is completely segregated from the CPU stack, enabling the use of asynchronously running "tasklets," which are more efficient than OS threads.

Python on Mobiles

Python has been ported to a number of mobile phone platforms such as the iPhone, Android, Windows Mobile, Nokia S60 and others, so you can enjoy the same spirit of programming while on the move.

Find the needle in the haystack.

Companies in every industry generate huge volumes of data. Thriving businesses look to **Python** to convert this “haystack” of data into informed business decisions. **Continuum** provides your organization with expert tools and ideas for array-oriented, lightning fast computing of large data sets -- on a single machine or over a cluster. **It's NumPy, SciPy and Matplotlib on steroids.**

Manage. Analyze. Decide. Thrive. **Continuum.**

<http://continuum.io/psf>



**CONTINUUM
ANALYTICS**



The Beginning of a Good Development

Start your mission with DLR

Bright ideas? Creative solutions? Analytical talents? Then you will be in good company at DLR: we support cutting edge research in aeronautics and aerospace, energy, transport and security with our powerful software technology – for aircraft design, space systems and beyond. At DLR, you will not only find great challenges, but also freedom for your own ideas. Turn your visions into reality – with software solutions for tomorrow's world: www.DLR.de/jobs.



DLR

German
Aerospace Center

»»» Open courses and tailored in-house trainings for companies and institutions.

»»» We had a wide range of Python experience in our group and each person gained something valuable to take away...

Dr. Ryan Woodard –
Chair of Entrepreneurial Risks ETH Zurich, Switzerland



»»» Share our international experience.

Take advantage of our tailored solutions.

_ to meet your exact needs. We combine, adapt, and extend our extensive collection of Python training modules to create the best course for you. We teach at your premises worldwide.

Dr. Mike Müller – CEO

Training

»»» Python for programmers and the customized "Python for Experts" course were a great success.

Bart Hillaert –
Alcatel-Lucent Belgium

Coaching

»»» Bring your Python coding skills to a new level.

We guide your developers to the optimal usage of Python for better solutions.

Take advantage of our experience.

Consulting

»»» We program and explain.

We develop solutions applying the full power of Python.

We offer code review workshops to make sure you understand the source code to the level of detail that suits your needs.

python-academy.com

>>> from pya import take_course

Looking for python experts?



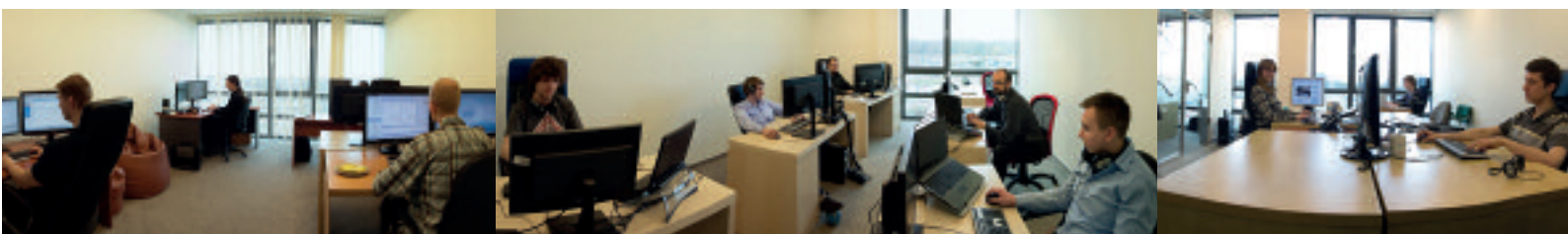
Our software house is the answer to your needs. We provide:

- **Python development resources**
Plone, Pyramid, Twisted, App engine, django
- **Web and mobile development resources**
HTML, javascript, CSS, iOS, Android, Sencha, PhoneGap, jQuery Mobile
- **Expertise based on experience**
Projects for biggest banks in Poland
Dozens of completed Python projects worldwide (USA, UK, NL, SE)

Why us ?

Because only from us you can get experienced team of Python developers within few days notice

<http://www.stxnext.pl/offer>



booked



The CMS giving
you peace of mind

Having a solid security record, proven scalability and an experienced community
Plone is the decision you'll never regret

Take a Tour

<http://plone.com/qr/taketour#0763>



Visit Plone.com
and discover...

- Industrial Strength Security
- Robust Extensibility
- Flexible and Adaptable Workflows
- Designer-Friendly
- Extensive Integration Capabilities

Industry
Media
Education
Intranet
Government
Small Business
NGO



The Plone® CMS/WCM is © 2000-2013 the Plone Foundation and friends. Plone® and the Plone logo are registered trademarks of the Plone Foundation.
<http://plone.org/foundation>

Plone.com, CommunityM, 20130113, 14:03:26/001 www.plone.com

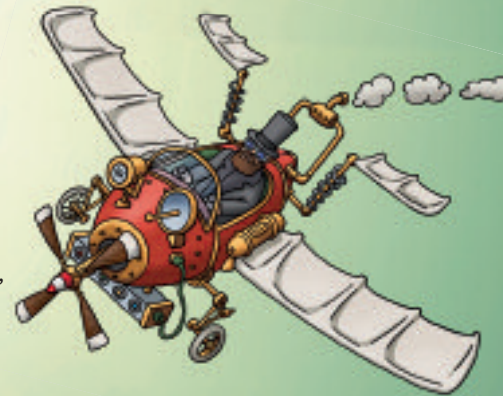
FLYING CIRCUS

LET YOUR WEB APP FLY



Flying Circus is an integrated, Python-optimised development and operations environment for web applications of all sizes – operations-as-a-service tailored to Python.

With Flying Circus you will enjoy a comfortable Linux environment. You can use pre-configured best-of-breed components like nginx, haproxy, Postgresql, and many more. Flying Circus takes care of operational duties like monitoring, backup, security, and infrastructure.



Let your web app fly and get ready for take-off on flyingcircus.io!

AD 1/2

Ad Sponsorship

The production of the brochure will be refinanced using sponsorship ads.

Don't miss your chance to present your company in the first Python image brochure – space is limited!

...introducing the new professional quality

PSF Python Brochure

“Python is already very well known among developers and becoming increasingly popular as the rapid development language of choice. We would like to extend this popularity to everyone who programs in industry, governments, science and education.

The brochure project is intended to reach out to these people, by showing how wide-spread Python already is and how its efficiency and simplicity can make a major difference.”

Marc-André Lemburg
PSF Board Member and Project Lead



Our Target Groups

CIOs
Chief architects
Scientists
Engineers
Programmers
Researchers

Teachers
Lecturers
Students
Customers
Clients
Managers

Featured topics

Web
Science
Finance
Industry
Education

Aerospace
Business
IT Infrastructure
Cloud Technology
Film and Animation

Python Software Foundation

“As chairman of the PSF I am happy to see the Foundation support this effort to bring Python to the attention of a wider public.

Businesses need to understand how Python can lower their programming and maintenance costs. Programmers need to understand how well Python integrates with other languages. Politicians need to understand the significance of open source to national strategies for science and education.

I really hope that we can eventually have the brochure translated, and include examples of Python use from all over the world.

The Python Software Foundation would love your help in bringing Python to a broad international community.”

Steve Holden
Chairman of the
Python Software Foundation

“The editors team enjoyed talking to many fantastic people in the last months, seeing the real value behind using Python and collecting inspiring feedback from very important projects all over the world.”

Armin Carl Stroß-Radschinski
Producer

“For me, working on the Python Brochure was a really interesting experience. Of course, I knew about the typical list of advantages people tell you when trying to sell Python. **What I didn't know was that Python has indeed changed the world.**

The success stories prove that Python has the potential to open up software engineering to a broader public, turning it into a skill that many people can learn and benefit from.”

Jan Ulrich Hasecke
Editor

Sponsoring Options

You can support the PSF Python Brochure by buying ads or a reference entry in the brochure as well as pre-ordering copies for your own purpose.

Ad sponsors

The brochure will have 4 DIN A4 pages for ads. Ad sponsors can buy a **half page ad** in the first edition. Each ad sponsor will receive a box of around 120 free copies of the printed brochure after production.

There is room for 8 ad sponsors.

Reference sponsors

To help develop the business community around Python, we will have 2 pages of fixed-layout reference listings for interested companies. Reference sponsors can buy a **listing** in the first edition. The companies will get listed using their logo, a one line description of what they do, and their address details. Each reference sponsor will receive a box of around 120 free copies of the printed brochure after production.

There is room for 30-40 reference sponsors.

Subscription sponsors

You can pre-order additional copies in excess of the 10,000 copies we will have printed and distributed by the PSF. **You can use these copies for local distribution, marketing Python to your customers, at conferences and events.**

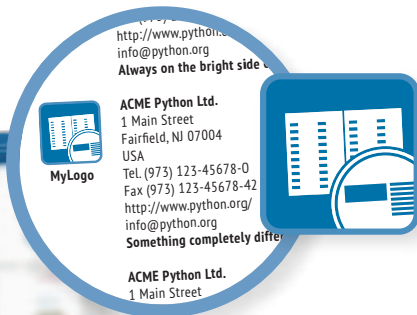
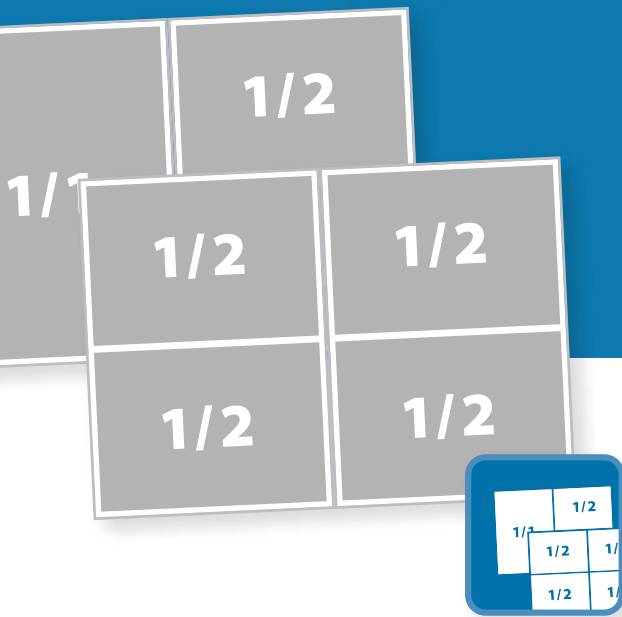
These sponsors will not get a listing in the brochure itself, but will help refinance the costs of producing the brochure and also help spread the word more widely.

We can ship world-wide. The costs for the subscriptions vary depending on the destination country. Please contact for more detailed information and other destinations.

Education Sponsorship

Sponsor educational institutions by having the PSF send subscription packages to universities and schools on your behalf.

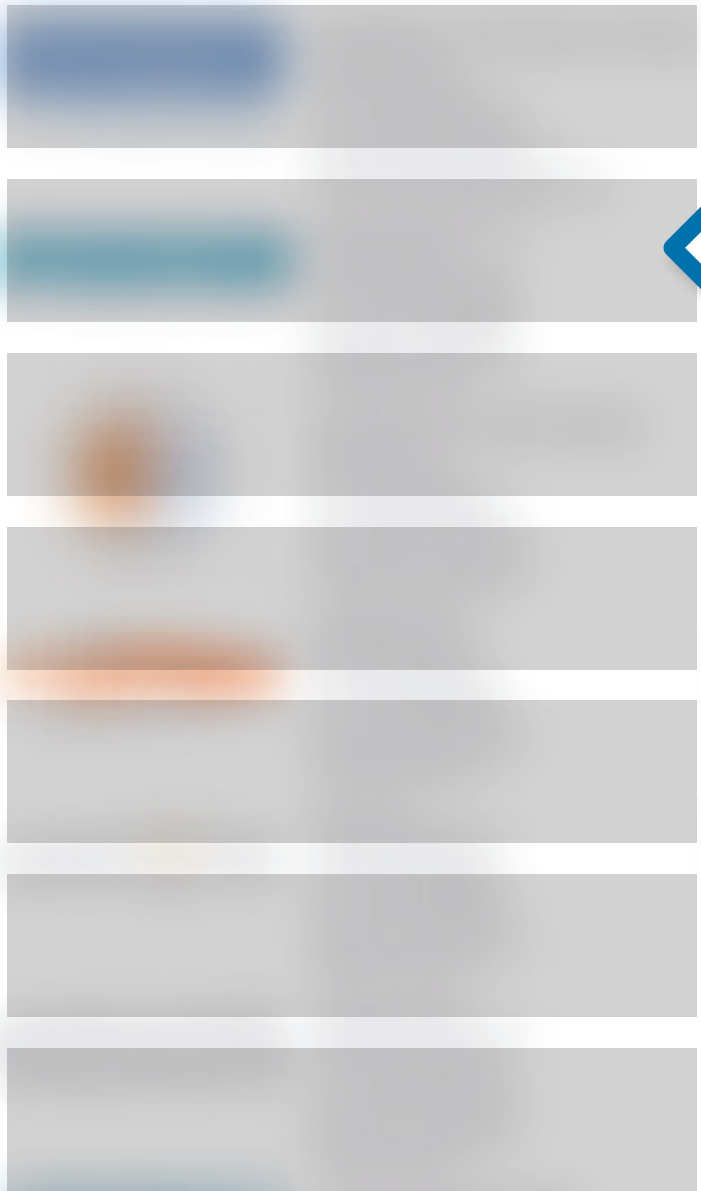
You can direct the PSF to send the packages to a specific institution or let the PSF select one from a list of interested institutions. Billing and payment will be handled by our producer *evenios publishing*.



Don't miss the chance to present your company in the first Python image brochure – space is limited !



<http://brochure.getpython.info/>



Reference Entry

With the reference entry sponsorship you can support the production too!

Don't miss your chance to present your company in the first Python image brochure – space is limited!

Some of the stories did not make it into this brochure but are worth a look. Please make sure to visit the project on the web and check out more of the fantastic world of Python and check out how this wonderful language has already changed in the world.

<http://brochure.getpython.info>



MyLogo

<http://www.python.org/info@python.org>
Always on the bright side

ACME Python Ltd.
1 Main Street
Fairfield, NJ 07004
USA
Tel. (973) 123-45678-0
Fax (973) 123-45678-42
<http://www.python.org/info@python.org>
Something completely differ

ACME Python Ltd.
1 Main Street

This could be your entry!

Please make sure to look at the project website for general information

- **Ad Guidelines**
Media data for creative people & print professionals.
- **Reference Entry Guidelines**
How to get listed in the Python brochure as Python support or service company.
- **Subscription & Education Sponsorship**
How to order additional extra copies of the brochure for your company, university or event.
- **Media Data**
Please visit our media data details pages regarding available ad sizes, ad and reference listing guidelines and ordering procedures.



Ad Sponsorship

The production costs of the brochure will be financed using sponsorship ads.

Please get in touch with us !

Contact us via email:

brochure@getpython.info

Sign up for our newsletter:

brochure.getpython.info

Follow us on Twitter:

twitter.com/pythonbrochure

Don't miss the chance to present your company in the first Python image brochure – space is limited !

<http://brochure.getpython.info/>

PreRelease

Ad Sponsorship

The production of the brochure will be refinanced using sponsorship ads.

Don't miss your chance to present your company in the first Python image brochure – space is limited!

Reference Entry

With the reference entry sponsorship you can support the production too!

Don't miss your chance to present your company in the first Python image brochure – space is limited!

About the PSF

The Python Software Foundation (PSF) is a non-profit organization devoted to advancing Open Source technology related to the Python programming language. The PSF holds the intellectual property rights to Python and plans to fund portions of future development of the language. Additional information on the PSF is available at <http://www.python.org/psf>. To make a tax-deductible donation, please visit <http://www.python.org/psf/donations.html>. Corporate sponsorships are also being accepted.



PythonBrochure_2013103112_20-04-12_LV82 - evenios.com - hasecke.com

Copyrights & Licensing
Except where otherwise noted, content in this brochure is Copyright 2011-2012 by evenios.com | Armin Stross-Radschinski (acsrde) | Ian Ulrich Hasecke (hasecke.com) | Marc-Andre Lemburg (lemburg.com) and the respective authors of the success stories, case studies and images published in this brochure. Contributed text and image material used with permission. The English text and some images are available under a CC BY-SA 3.0 License except otherwise noted here. The following articles are exclusively licensed to the PSF only for this brochure with a CC BY 3.0 license: p.18,19 © CERN. Please contact the PSF or the publisher for more details.
Image & Logo Reference: The Python logo and the Python device are trademarks of the Python Software Foundation and used by permission. All other trademarks & logos are property of their respective owners, used under fair use policies for information purposes only and not part of the CC BY-SA 3.0 License. Reuse of third party logos in derivative work may need relicensing for the respective purpose. Cover: illustration acsrde; thumbs see: - Photos: Source fitckr.com (p.2 /x/matt: CC BY 2.0 | p.4 /mitot: used by permission | p.9 /x/vitalieida CC BY 2.0 | p.10/11 UN Photo/Logan Abassi CC BY 2.0 | p.14/15 /dionhinchcliffe, sculpture artist built & bear: Dachlauer, Reinhard (1922 - 1995); background: /kenteegardin, www.SeniorLiving.Org; CC BY-SA 2.0); (p.4 screenshot mozilla.com taken on 2012-02-12); (p.5 screenshot youtube.com taken 2012-02-12); (p.6 SurveyMonkey and the SurveyMonkey logo are trademarks of SurveyMonkey.com, LLC and its affiliates and are used under license); (p.7 kartograph.org; CC BY-SA 2.0); (p.8 openerp.com, used by permission); (p.10 noaa.gov, public domain); (p.12 © SMS Siemag-AG; all rights reserved, used by permission); (p.13 © DLIR; CC-BY 3.0); (p.16 one.laptop.org; CC BY 2.0); (p.18,19 the picture of the CMS detector: © CERN, for the benefit of the CMS collaboration. Terms of use: <http://copyright.web.cern.ch/>); (p.20 NASA/PL-Caltech, public domain); (p.22 sidefx.com, all rights reserved, fair use); (p.23 Blender Foundation | www.sintel.org CC BY 3.0); (p.24 The OpenStack logo is a trademark of the OpenStack Foundation, all rights reserved, used by community use general licence)