EuroPython 2007

Monday 9 July 2007 - Wednesday 11 July 2007

Vilnius, Lithuania

Programme

This year we are aiming not so much at rigidly defined tracks as at *themes*.

At previous the EuroPython conferences, the talks on the themes listed on this page have been of interest.

Science

Talks with this theme will focus on the use of Python in science and industry, where tasks include modelling complex systems (thermics, fluid dynamics, mechanics, aeronautics, biology, chemistry, etc.), processing very large data sets and performing very CPU-intensive and long calculations efficiently. Speakers will present tool sets, frameworks and examples of successful applications based on Python and integrated with the other usual tools and applications used in the field.

Python Language and Libraries

A theme centred on "Python the Language" with all batteries included. Suitable for consideration within this theme are talks about the language, language evolution, patterns and idioms, implementations (CPython, IronPython, Jython, PyPy, CLPython, ShedSkin, and especially any new ones presented for the first time) along with implementation issues and attempts to port Python implementations to new or novel devices. Talks about the standard library or interesting third-party libraries (and frameworks) are also welcome, unless the talks would fit better into one of the other themes.

Web Related Technologies

Python's role in implementing Web applications and solutions continues to evolve, revealing new success stories and a continuous stream of new tools and projects. As with last year's conference, talks traditionally part of the Zope track will take place alongside other Web-related talks as part of a wider Web technologies theme. Join us as we hear about success stories, new ideas about Web programming, and the potential for increased collaboration between projects in the Web domain.

Education

Are you trying to teach Python to somebody, or are you using Python to try to teach something else? Whether you have something to say about the use of Python as a teaching language in its own right, or simply use it to augment an existing curriculum, attendees are interested in hearing about your experiences.

In previous years, talks on this theme have covered a wide range of subjects and situations, from teaching mathematics to elementary school children to the use of Python to teach science to graduate students. Previous talks have also addressed specific Python-based educational tools - these kinds of talks often generate useful feedback for developers of such tools, and often lead to stimulating discussions.

All people who are using Python in a teaching environment are welcome.

Games

Games have always provided a compelling motivation for people new to computing and computers to enter the field, often inspiring them to go on and design and implement new games and other forms of entertainment systems. Python provides such people, as well as experienced developers, with a platform to develop anything from simple puzzle games and "retro" arcade games through to three dimensional action games and massive multiplayer online gaming systems.

Python is used commercially for games infrastructure, scripting, and as the main development language of entire games. Meanwhile, there are community events which encourage participants to build a game from scratch in Python in only a week.

However, not all the tools that people use for writing games end up being used that way: many game-related projects are also used for developing entertainment solutions such as personal video recorder (PVR) systems, audio and video players, and presentation tools. Talks with this theme might cover the practicalities of getting started and choosing the right tools, building larger games or gaming platforms, applying game and multimedia libraries in other areas, or even sharing secrets about developing a game in a week and actually getting it finished!

Agile Experiences and Testing

What agile practices and experiences have you had in your open source or proprietary development projects since last years EuroPython?

We encourage talks that share experiences regarding:
agile practices/methodologies (XP, Scrum, Crystal, test-driven development, sprints, pair-programming etc) used with regards to what worked and what did not work (and why?).
agile practices when working distributed:
the dos and don'ts (and why?)
supporting tools for (amongst other things) testing, timetracking, collaborative planning and project management.

And other "agile" related experiences you might have and that others could benefit from hearing about!

Social Skills

If these areas of experiences ring a bell then please present a talk and let others hear about how you survived and what survival tactics you employed!
how to deal with difficult behaviour
how to communicate more efficiently
how to successfully influence people

leadership: the dos and don'ts
team dynamics: what worked and what did not work so well?
evolving communities: the dos and don'ts

...or other social skills related experiences you think the Python community **really** need to hear about.

Refereed Paper

Ever since the International Python Conference (IPC) stopped running as a separate event, the Python world has lacked a properly prestigious peer-reviewed forum for presenting technical and scientific papers. To fill that gap you can decide if your EuroPython contribution should be refereed or not.

Business and Applications

This is where EuroPython thinks outside our Python community - about the applications we have written for ordinary people and businesses, and about how we've sold them to the outside world.

What Python apps have you written? Tell your fellow Pythonistas about them. Exchange knowledge and maybe gain new business partners.

How do you sell your apps and services into the business community?

What strategies have you used to convince potential customers and what works for you? Come to think of it, what doesn't work?

How do you license your apps? Do you use a Free Software licence or is your application proprietary? Tell us what path you have chosen and why.

What have you learnt about introducing new technology into userland?

Share your experiences with the community and go home enthusiastic and enlightened.

At previous EuroPythons we have heard about applications as diverse as indexing and searching the US patent database, and payroll. We have had panels on software patents (more work to do yet, I'm afraid) and licensing.

If you have any questions about how you can contribute to this track, please contact the Track Chairs, but most of all, please send us your proposals for talks for the Applications and Business Track.