



Programme



Contents

| | |
|---|-----------|
| Introduction | 8 |
| Map of Chalmers | 9 |
| Venue Map | 10 |
| Monday Timetable | 11 |
| Sponsors | 12 |
| Tuesday Timetable | 13 |
| Sponsors | 14 |
| Wednesday Timetable | 15 |
| Session Listings | 16 |
| Zope keynote | 16 |
| Scripting a JAVA search engine with Python | 16 |
| The art of giving a talk. Get your message across and keep the audience alive! | 16 |
| Developing large-scale applications in Python | 17 |
| a Python sales pitch | 17 |
| Architecture of a large Zope 3 system | 17 |
| TVTK: A Pythonic VTK | 18 |
| A Python Framework for Rapid Application Development | 18 |
| CPSWorkflow : From stateful to activity based workflows | 19 |
| Implementing Continuous Time Simulation Systems in Python | 19 |
| Who else hates selling? How to make your customer want to do business with you. | 20 |
| Why I invented Python | 20 |
| XML information extraction in Python | 20 |
| Making Money on Plone - Why The Model Works | 21 |
| Document Library: a Zope 3 based Document Management System | 21 |
| CPS Platform Framework - | 22 |
| AlphaFlow: Background and Technical Details | 22 |
| The Python Community in Brazil | 22 |
| Taste Python, part 1 | 22 |
| CMFEditions Versioning Framework for CMF/Plone | 23 |
| Community and Cash panel | 23 |
| MayaVi-2: The next generation | 23 |
| A Virtual File System for Python | 24 |
| The world according to Leo | 24 |
| Ixml, a Pythonic binding for libxml2 | 25 |
| Taste python, Part 2 | 25 |
| Managing a distributed team | 25 |
| State of Plone - Plone Grows Up | 26 |
| The itools XML infrastructure | 26 |
| Batteries included - Introduction to the Python Standard Library | 26 |
| Chandler | 27 |
| Make Poverty History with Plone | 27 |
| Teaching computational engineering using Python | 27 |
| Versioning and relation management in a Zope ECM | 28 |



EuroPython Conference

| | |
|---|----|
| What is Open Source Business? | 28 |
| CAPSconference | 29 |
| A gentle introduction to Python's way | 29 |
| Enabling bare Python as universal connector for ad-hoc networks | 30 |
| Batteries included, part 2 | 30 |
| In Theaters Now!--How Python Runs the Largest Independent Cinema Chain in the UK | 30 |
| Zope Topic Maps - building metadata-based portals with open standards | 31 |
| Pulling Java Lucene into Python: PyLucene | 31 |
| Using Python to glue Windows and AS/400 applications together | 33 |
| CPSMailAcces : a webmail application in CPS3 using Zope 3 and Five | 33 |
| Zope/Zeo in Real Life Deployment | 34 |
| Python on a Phone | 34 |
| A powerfull Cell within the batteries: decimal for beginners | 34 |
| Unit-testing PyGTK applications | 35 |
| Experience with Python for Series 60 in the PDIS project | 35 |
| Breaking the Rules of Commercial Software Development : Why and How we used Python to Produce a New Deal in Payroll Software. | 36 |
| club travel invoices with Python-Excel | 36 |
| FSF Europe - Free Software - Free Society | 36 |
| Open Space | 37 |
| Kamaelia | 37 |
| CherryPy tutorial | 38 |
| The Creative Commons Project | 38 |
| Electronic Measurement Automation | 38 |
| Licensing panel | 39 |
| 2 + 3 = ? -- A hands-on approach to slow migration | 39 |
| Visualization of three- and four-dimensional microscopy images with BioImageXD | 39 |
| GUI programming in Python using Gtk+ | 40 |
| Open Space | 40 |
| Complex security with Zope 3 and an RDB | 40 |
| A Laboratory Information Management System for High-throughput, High-content Screening | 41 |
| Five in action | 41 |
| Modular toolkit for Data Processing (MDP) | 41 |
| Twisted news | 42 |
| SchoolBell -- calendaring in Zope 3 | 42 |
| C-Lab: a portal for Computational Chemistry at AstraZeneca | 43 |
| Moellus - a soft layer on Durus | 43 |
| Writing XMMS2 clients in Python | 44 |
| PyPy as a Compiler | 44 |
| Plone4Artists | 45 |
| Z3 ECM Panel | 45 |
| Python as a Tool in Resource Optimization Systems | 46 |
| Python and a few more animals - using Python in science & research at MPI-CBG | 46 |
| BioSimGrid : An Environment to Archive and Analyse Biosimulation Data | 46 |



EuroPython Conference

| | |
|--|-----------|
| Cyprix - A 'Swiss army knife' for operating SQL databases | 47 |
| Recoverable Exceptions In Python | 47 |
| Plone used for semantic web in the construction industry | 48 |
| Pythonic Mathematics | 48 |
| Introduction to regression testing using TextTest | 49 |
| Greenlets: coroutines aren't stranger than generators | 49 |
| Where metaclasses surpass decorators | 50 |
| Storage Resource Broker (SRB), Large scientific data and Python | 50 |
| RhoZeta and Phi ² - master the challenge of forms with the power of Python | 51 |
| Solving puzzles with Python | 51 |
| A Python Experiment in a Brazilian University | 52 |
| A Simple Python Framework for Introducing Component Principles | 52 |
| iKaaro Case Study: Traveluni | 52 |
| Smart Package Manager | 52 |
| Stupidity and laser cat toys: Indexing the US Patent Database with Xapian and Twisted | 53 |
| Fun matters | 53 |
| Localizer 1.1, and Case Studies | 53 |
| The Python revolution in the publishing industry | 54 |
| PyPy and sprint-driven development | 54 |
| Python in Pension Funds | 54 |
| Pimp my site! | 55 |
| ItsATree - creating a multimedia editor | 55 |
| Choosing Good Names | 56 |
| Python and Money: Discussion | 56 |
| Web Application Testing with Selenium | 56 |
| py.test - making testing fun | 57 |
| Python Logo and Website Redesign | 57 |
| Money module | 57 |
| Using tests for team motivation | 58 |
| WYSIWYG interface design with CPSSkins and CPSPortlets | 58 |
| The Python Virus - Using Python for ten years in ePublishing business | 59 |
| The Personal Internet Endpoint: Using Python and Twisted to write Reliable Peer-To-Peer Programs | 59 |
| Method panel - you don't get paid to have fun! | 60 |
| Zope 3 Common Criteria Certification | 60 |
| Zope and Zope Foundation | 60 |
| Lightning talks | 61 |
| Zope Lightning talks 1 | 61 |
| Zope Lightning talks 2 | 61 |
| Lightning talks | 61 |
| Presenter Listings | 62 |
| Albertas Agejevas | 62 |
| Andrea Agili | 62 |
| Steve Alexander | 62 |
| Johan Andersson | 62 |



EuroPython Conference

| | |
|---------------------------|----|
| Julien Anguenot | 62 |
| Martin Aspeli | 62 |
| Nate Aune | 62 |
| Emily Bache | 63 |
| Geoff Bache | 63 |
| Facundo Batista | 63 |
| Facundo Batista | 63 |
| Luis Belmar-Letelier | 63 |
| Pietro Berkes | 63 |
| Alberto Berti | 64 |
| Aaron Bingham | 64 |
| kit blake | 64 |
| Carl Friedrich Bolz | 64 |
| Duncan Booth | 64 |
| Pravir Chawdhry | 64 |
| Rob Collins | 64 |
| Richard Cooper | 65 |
| Johan Dahlin | 65 |
| Guy Dalberto | 65 |
| Theo F. de Ridder | 65 |
| Remi Delon | 65 |
| Remi Delon | 65 |
| André Deparade | 65 |
| Vincenzo di Somma | 65 |
| Beatrice During | 66 |
| Anders Egneus | 66 |
| Paul Everitt | 66 |
| Martijn Faassen | 66 |
| Hans Fangohr | 66 |
| Farzad Faramarzi | 66 |
| Stefane Fermigier | 66 |
| mathias foehr | 66 |
| Volker Gietz | 66 |
| Katherine Goodwin | 67 |
| Florent Guillaume | 67 |
| Jacob Hallén | 67 |
| Tero Hasu | 67 |
| Astrid Hellwig | 67 |
| Stefan H. Holec | 67 |
| Michael Hudson | 67 |
| Juan David Ibáñez Palomar | 67 |
| Andreas Rübner Johnsen | 68 |
| Steven Johnston | 68 |
| Andreas Jung | 68 |
| Norman Khine | 68 |



EuroPython Conference

| | |
|----------------------|----|
| Mathias Klang | 68 |
| holger krekel | 68 |
| Holger Krekel | 68 |
| Mikel Larreategi | 68 |
| Jukka Laurila | 69 |
| Marc-André Lemburg | 69 |
| Alexander Limi | 69 |
| Johan Lindberg | 69 |
| Arnar Lundesgaard | 69 |
| Benno Luthiger | 69 |
| Magnus Lyckå | 69 |
| Kevin Manley | 70 |
| Raphael Marvie | 70 |
| Stefano Masini | 70 |
| Harald Armin Massa | 70 |
| Gintautas Miliauskas | 70 |
| Stuart Murdock | 71 |
| Gustavo Niemeyer | 71 |
| Paul J. Nolan | 71 |
| Jean-Marc Orliaguet | 71 |
| Rob Page | 71 |
| Kalle Pahajoki | 71 |
| Tim Parkin | 71 |
| John Pinner | 71 |
| Brian Quinlan | 71 |
| Prabhu Ramachandran | 71 |
| Edward Ream | 72 |
| Reinout van Rees | 72 |
| Armin Rigo | 72 |
| Ken Rimey | 72 |
| André Roberge | 72 |
| Maik Roeder | 72 |
| Maik Roeder | 72 |
| Mario Ruggier | 72 |
| Tobias Rundström | 73 |
| Michael Salib | 73 |
| Michael Salib | 73 |
| Henrik Sandklef | 73 |
| Rodrigo Senra | 73 |
| Michele Simionato | 73 |
| Aroldo Souza-Leite | 73 |
| Michael Sparks | 73 |
| Christian Theune | 74 |
| Christian Tismer | 74 |
| Heikki Toivonen | 74 |



EuroPython Conference

| | |
|---------------------------|-----------|
| Kirby Uner | 74 |
| Kirby Uner | 74 |
| Andi Vajda | 74 |
| Guido van Rossum | 74 |
| Tommi Virtanen | 74 |
| Michael Volkmer | 74 |
| Philipp von Weitershausen | 74 |
| arjen Voogt | 74 |
| Grégoire Weber | 74 |
| Joe Wrigley | 75 |
| Tarek Ziade | 75 |
| Manuel Zini | 75 |
| Tiziano Zito | 75 |
| About ReportLab | 76 |



EuroPython Conference

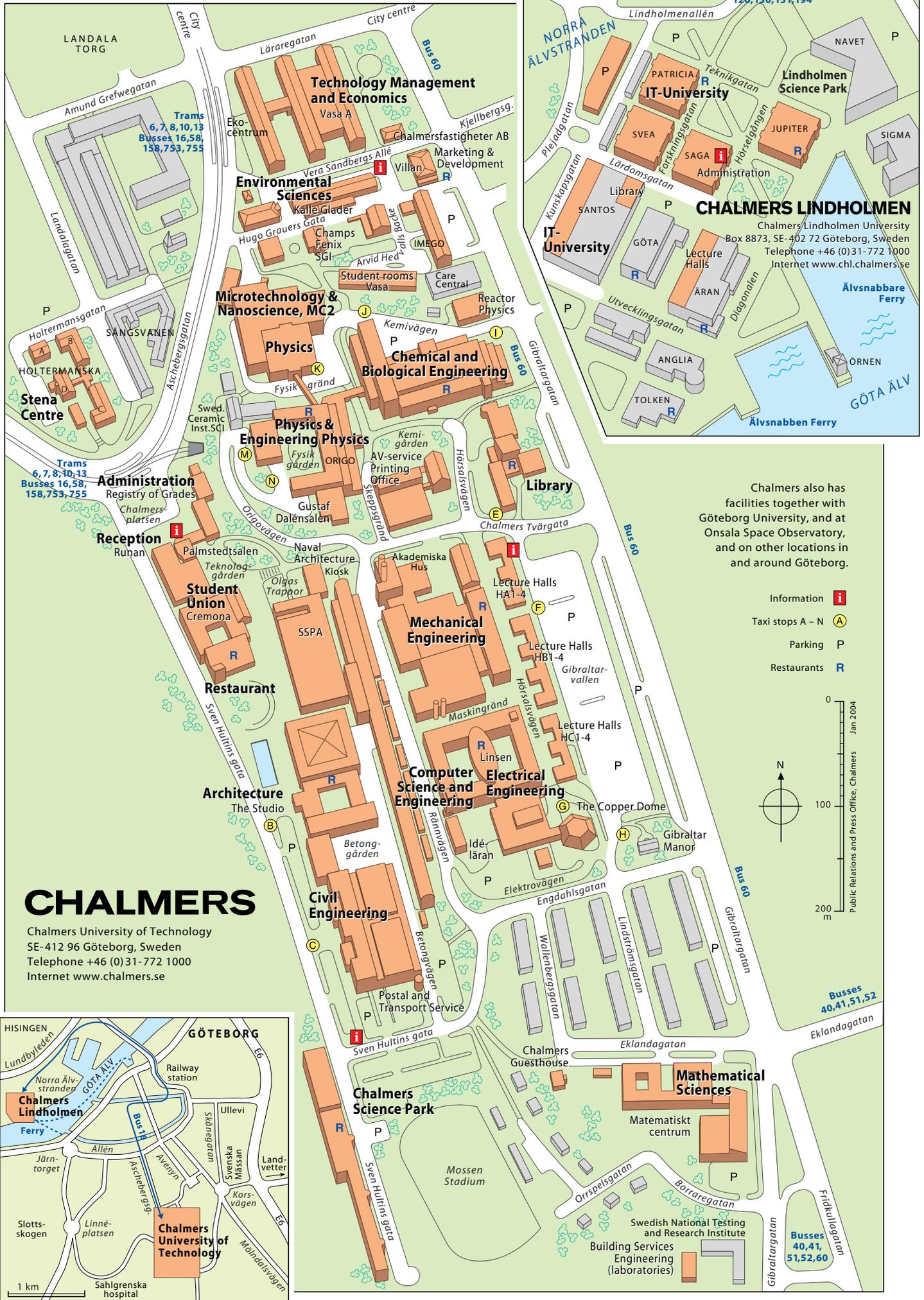
Introduction

Welcome to the EuroPython Conference 2005. While working on the arrangements for this years conference, it has become quite clear that the Python world is rapidly changing. In science Python has attained such a strong position that it is more surprising if you are not using Python than if you are. This is reflected in a small science track. In the business world, Python is just getting a foothold, and this shows in the strong interest in the Business, Social skills and Tutorials tracks.

The productivity benefits of Python can be seen in the large number of framework talks. You no longer need a staff of hundreds of programmers to build enterprise platforms. Among the frameworks, Zope has a special position with its large developer base and the large variety of structures built on top. It will be very exciting to see if the new Zope versions will attract newcomers to the field, or if other frameworks will have a stronger appeal.

This is the second and final year that Europython is located in Göteborg. Next years conference is planned to be held at CERN in Swizerland. Since Europython requires quite a bit of effort to arrange, and there are several members of this years staff who will have limited time next year, the new organisers will need many new volunteers to help with arranging the conference.

In our minds, a good conference is not only about listening to talks. It is just as much about meeting interesting people, being social and having fun.

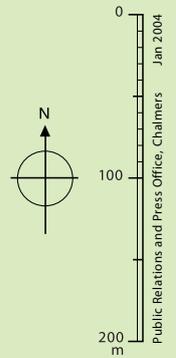


CHALMERS

Chalmers University of Technology
 SE-412 96 Göteborg, Sweden
 Telephone +46 (0) 31-772 1000
 Internet www.chalmers.se

Chalmers also has facilities together with Göteborg University, and at Onsala Space Observatory, and on other locations in and around Göteborg.

- Information **i**
- Taxi stops A - N **A**
- Parking **P**
- Restaurants **R**



Busses 40,41,51,52

Busses 40,41,51,52,60

Trams 6,7,8,10,13
 Busses 16,58,158,753,755

Trams 6,7,8,10,13
 Busses 16,58,158,753,755

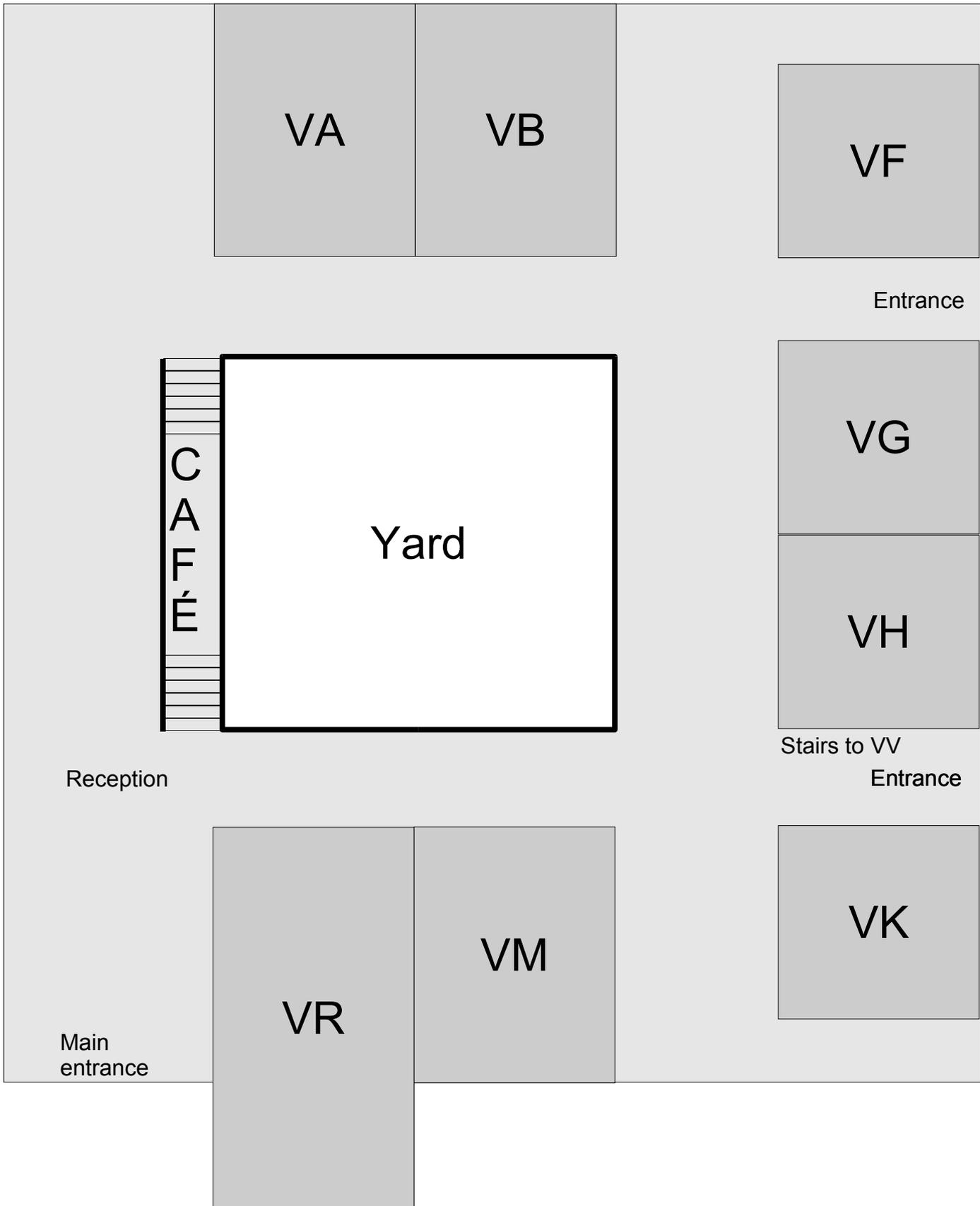
Busses 16,31,99,126,130,131,194

Busses 40,41,51,52

Busses 40,41,51,52,60

Venue map

To SGS



Entrance

VG

VH

Stairs to VV

Entrance

VK

VA

VB

VF

Yard

CAFÉ

Reception

VM

VR

Main entrance

To Student Union
and Trams





EuroPython Conference

Monday Timetable

| | VR | VA | VB | VF | VV | |
|-------|--|---|---|--|--|-------|
| 09:00 | Developing large-scale applications in Python [Lemburg] | The art of giving a talk. Get your message across and keep the audience alive! [Hellwig] | Zope keynote [Everitt] | Scripting a JAVA search engine with Python [Agili & Zini] | | 09:00 |
| 09:30 | | a Python sales pitch [Massa] | Architecture of a large Zope 3 system [Alexander] | TVTK: A Pythonic VTK [Ramachandran] | | 09:30 |
| 10:00 | A Python Framework for Rapid Application Development [Goodwin & Wrigley] | Who else hates selling? How to make your customer want to do business with you. [Hellwig] | CPSWorkflow : From stateful to activity based workflows [Anguenot] | Implementing Continuous Time Simulation Systems in Python [Nolan] | | 10:00 |
| 10:30 | | | | | | 10:30 |
| 11:00 | XML information extraction in Python [Quinlan] | Making Money on Plone - Why The Model Works [Aspeli] | Document Library: a Zope 3 based Document Management System [blake] | CPS Platform Framework - [Anguenot & Fermigier & Guillaume & Oriaguet & Ziade] | Why I invented Python [van Rossum] | 11:00 |
| 11:30 | | The Python Community in Brazil [Senra] | AlphaFlow: Background and Technical Details [Theune] | | | 11:30 |
| 11:45 | | | | | Taste Python, part 1 [Collins & Simionato] | 11:45 |
| 12:00 | A Virtual File System for Python [Ibáñez Palomar] | Community and Cash panel [Aspeli & Faassen & Hellwig & Massa & Senra] | CMFEditions Versioning Framework for CMF/Plone [Berti & di Somma & Weber] | MayaVi-2: The next generation [Ramachandran] | | 12:00 |
| 12:30 | | | | | | 12:30 |
| 14:00 | Managing a distributed team [Alexander] | The world according to Leo [Ream] | State of Plone - Plone Grows Up [Limi] | lxml, a Pythonic binding for libxml2 [Faassen] | Taste python, Part 2 [Collins & Simionato] | 14:00 |
| 14:30 | Chandler [Toivonen] | | Make Poverty History with Plone [Booth] | The itools XML infrastructure [Ibáñez Palomar] | Batteries included - Introduction to the Python Standard Library [Quinlan] | 14:30 |
| 15:00 | What is Open Source Business? [Everitt] | Teaching computational engineering using Python [Fangohr] | Versioning and relation management in a Zope ECM [Guillaume] | CAPSconference [Hallén] | | 15:00 |
| 15:30 | | | | | | 15:30 |
| 16:00 | In Theaters Now!--How Python Runs the Largest Independent Cinema Chain in the UK [Cooper & Delon & Manley] | A gentle introduction to Python's way [Roberge] | Zope Topic Maps - building metadata-based portals with open standards [Johnsen & Lundesgaard] | Enabling bare Python as universal connector for ad-hoc networks [de Ridder] | Batteries included, part 2 [Quinlan] | 16:00 |
| 16:30 | | | CPSMailAcces : a webmail application in CPS3 using Zope 3 and Five [Ziade] | Pulling Java Lucene into Python: PyLucene [Vajda] | Using Python to glue Windows and AS/400 applications together [Lindberg] | 16:30 |
| 17:00 | Python on a Phone [Laurila] | | Zope/Zeo in Real Life Deployment [Chawdhry] | Unit-testing PyGTK applications [Miliauskas] | A powerfull Cell within the batteries: decimal for beginners [Batista] | 17:00 |
| 17:30 | | | | | | 17:30 |
| 18:00 | Keynote | | | | | 18:00 |

Wing IDE for Python

- ∞ **Code Intelligence:** Auto-completion, Call Tips, Source Browser, Goto-Definition, Error Indicators, Syntax Highlighting, Indent Guides
- ∞ **Graphical Debugger:** Conditional Breakpoints, Interactive Prompt, Watch or Change Data, Debug Stand-alone or Embedded Code
- ∞ **Full-Featured Editor:** Split Views, Configurable Key Bindings, Code Folding, Rectangular Select, Reformatting Tools,
- ∞ **And More:** Zope Support, Scripting API, Integrated Tutorial and Documentation, German Localization
- ∞ **Available on:** Windows, OS X, Linux

Free Trial at <http://wingware.com/wingide/trial>

Wingware *Software Development Tools Designed for Python*



CHALMERS

AixtraWare





EuroPython Conference

Tuesday Timetable

| | VR | VA | VB | VF | VV | | |
|-------|---|--|---|---|--|-------|-------|
| 09:00 | club travel invoices with Python-Excel <i>[Dalberto]</i> | Experience with Python for Series 60 in the PDIS project <i>[Hasu & Rimey]</i> | Breaking the Rules of Commercial Software Development : Why and How we used Python to Produce a New Deal in Payroll Software. | FSF Europe - Free Software - Free Society <i>[Sandklef]</i> | Open Space | 09:00 | |
| 09:30 | CherryPy tutorial <i>[Delon]</i> | Kamaelia <i>[Sparks]</i> | <i>[Pinner]</i> Electronic Measurement Automation <i>[Dalberto]</i> | The Creative Commons Project <i>[Klang]</i> | Open Space | 09:30 | |
| 10:00 | | | | Licensing panel <i>[Klang & Sandklef]</i> | | 10:00 | |
| 10:15 | | | | | | | 10:15 |
| 10:30 | | | | | | | 10:30 |
| 11:00 | GUI programming in Python using Gtk+ <i>[Dahlin]</i> | Complex security with Zope 3 and an RDB <i>[Alexander]</i> | 2 + 3 = ? -- A hands-on approach to slow migration <i>[von Weitershausen]</i> | Visualization of three- and four-dimensional microscopy images with BioImageXD <i>[Pahajoki]</i> | Open Space | 11:00 | |
| 11:30 | | | Five in action <i>[Faassen]</i> | A Laboratory Information Management System for High-throughput, High-content Screening <i>[Bingham]</i> | | 11:30 | |
| 12:00 | | Twisted news <i>[Virtanen]</i> | SchoolBell -- calendaring in Zope 3 <i>[Agejevas]</i> | Modular toolkit for Data Processing (MDP) <i>[Berkes & Zito]</i> | | 12:00 | |
| 12:30 | | | | | | 12:30 | |
| 14:00 | PyPy as a Compiler <i>[Bolz & Krekel & Rigo & Tismer]</i> | Moellus - a soft layer on Durus <i>[Ruggier]</i> | Plone4Artists <i>[Aune]</i> | C-Lab: a portal for Computational Chemistry at AstraZeneca <i>[Bache & Egneus & Faramarzi]</i> | Writing XMMS2 clients in Python <i>[Runström]</i> | 14:00 | |
| 14:30 | | Python as a Tool in Resource Optimization Systems <i>[Lyckå]</i> | Z3 ECM Panel <i>[Everitt]</i> | Python and a few more animals - using Python in science & research at MPI-CBG <i>[Volkmer]</i> | | 14:30 | |
| 15:00 | Recoverable Exceptions In Python <i>[Hudson]</i> | Cyprix - A 'Swiss army knife' for operating SQL databases <i>[Deperade]</i> | | BioSimGrid : An Environment to Archive and Analyse Biosimulation Data <i>[Murdock]</i> | | 15:00 | |
| 15:30 | | | | | | 15:30 | |
| 16:00 | Greenlets: coroutines aren't stranger than generators <i>[Rigo]</i> | | Plone used for semantic web in the construction industry <i>[Rees]</i> | Pythonic Mathematics <i>[Urner & Urner]</i> | Introduction to regression testing using TextTest <i>[Bache & Bache]</i> | 16:00 | |
| 16:30 | Where metaclasses surpass decorators <i>[de Ridder]</i> | | Storage Resource Broker (SRB), Large scientific data and | | | 16:30 | |
| 16:45 | | RhoZeta and Phi ² - master the challenge of forms with the power of Python <i>[Massa]</i> | Python <i>[Johnston]</i> A Simple Python Framework for Introducing Component Principles <i>[Marvie]</i> | A Python Experiment in a Brazilian University <i>[Senra]</i> | | 16:45 | |
| 17:00 | Solving puzzles with Python <i>[Niemeyer]</i> | | | | | 17:00 | |
| 17:30 | | | | | | 17:30 | |
| 18:00 | Keynote | | | | | 18:00 | |

**Build your own
image of STRAKT**

www.strakt.com

business

collaboration

people

powered by

STRAKT

AB STRAKT

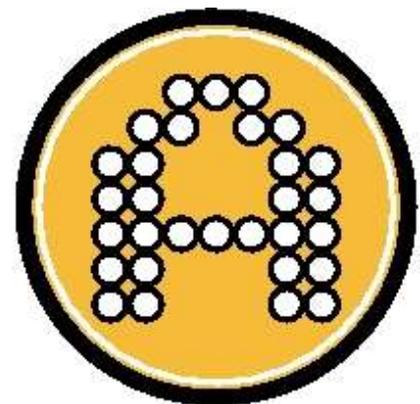
phone +46 31 749 08 80 | fax +46 31 749 08 81 | www.strakt.com | info@strakt.com

www.webwitches.com



WEBWITCHES

Open and proud of it
ouverts et fiers de l'être



AMAZE



**Zope & Plone
Intranet Services
Content Management
Educational Environments**

www.amazesolutions.com



Wednesday Timetable

| | VR | VA | VB | VF | VV | |
|-------|---|---|---|--|----|-------|
| 09:00 | Smart Package Manager <i>[Niemeier]</i> | Fun matters <i>[Luthiger]</i> | Stupidity and laser cat toys: Indexing the US Patent Database with Xapian and Twisted <i>[Salib]</i> | iKaaro Case Study: Traveluni <i>[Belmar-Letelier & Khine]</i> | | 09:00 |
| 09:30 | The Python revolution in the publishing industry <i>[Masini]</i> | PyPy and sprint-driven development <i>[During]</i> | Python in Pension Funds <i>[foehr]</i> | Localizer 1.1, and Case Studies <i>[Ibáñez Palomar & Larreategi]</i> | | 09:30 |
| 10:00 | ItsATree - creating a multimedia editor <i>[Gietz]</i> | Choosing Good Names <i>[Holek]</i> | | Pimp my site! <i>[Voogt]</i> | | 10:00 |
| 10:30 | | | | | | 10:30 |
| 11:00 | | py.test - making testing fun <i>[krekel]</i> | Web Application Testing with Selenium <i>[Roeder & Roeder]</i> | Python and Money: Discussion <i>[Batista & Pinner]</i> | | 11:00 |
| 11:30 | Python Logo and Website Redesign <i>[Parkin]</i> | Using tests for team motivation <i>[Andersson & Bache]</i> | WYSIWYG interface design with CPSSkins and CPSPortlets <i>[Anguenot & Orliaguet]</i> | Money module <i>[Batista]</i> | | 11:30 |
| 12:00 | The Personal Internet Endpoint: Using Python and Twisted to write Reliable Peer-To-Peer Programs <i>[Salib]</i> | Method panel - you don't get paid to have fun! <i>[Andersson & During & Holek & krekel & Luthiger]</i> | Zope 3 Common Criteria Certification <i>[Souza-Leite]</i> | The Python Virus - Using Python for ten years in ePublishing business <i>[Jung]</i> | | 12:00 |
| 12:30 | | | | | | 12:30 |
| 14:00 | Lightning talks | | Zope and Zope Foundation <i>[Page]</i> | | | 14:00 |
| 14:45 | | | Zope Lightning talks 1 | | | 14:45 |
| 15:30 | | | | | | 15:30 |
| 16:00 | Lightning talks | | Zope Lightning talks 2 | | | 16:00 |



Session Listings

Zope keynote

Presenter: Paul Everitt

Track: Zope

Day: Monday, June 27

Time: 09:00 - 09:30

Location: VB

Zope keynote

Scripting a JAVA search engine with Python

Presenters: Andrea Agili, Manuel Zini

Track: Misfits

Day: Monday, June 27

Time: 09:00 - 09:30

Location: VF

Isobel is a highly configurable and customizable open source knowledge management framework. Isobel is fully functional out of the box, but it's meant mainly as a framework to build complex information retrieval and analysis systems. Isobel offers ready to use services like multi protocol content fetching with fine grained scheduling, formats conversion (pdf, doc, etc.), hyperlink graph storage and analysis, content storage and indexing, ontology services, xml configuration, logging and statistics. On top of this services a pipeline plug-in model allows the programmer to easily extend the system with specific advanced functions. That's where we used Python, plug-ins can be developed in Python and added to the processing pipeline. The Python code has access to all the isobel services and environment. The Python integration is achieved through Jython platform embedding.

The art of giving a talk. Get your message across and keep the audience alive!

Presenter: Astrid Hellwig

Track: Social skills

Day: Monday, June 27

Time: 09:00 - 09:30

Location: VA

Everybody wants to and has to sell something - all the time. Either oneself, one's ideas and aspirations, one's time, one's products or one's services. In some cases, we are natural talents in selling, in others, we are not. Selling one's idea in a conference talk can be a tough thing to achieve. This session will give an enticing first insight into basic presentational skills. The audience is asked to experiment a little among themselves and see what kind of presentation has what kind (or no kind) of effect on the targeted audience. The session offers a simple but effective approach to giving better talks. The main focus is on basics such as eye contact, an articulate speech, and body language. Lots of examples and on stage acting will help the audience to immediately see and feel the difference it makes! Without stage fear, without practicing presentational skills time and again and without taking oneself too serious, the participants will remember the important and useful basics of how to give an effective talk. Thus, this session can contribute to a more lively conference, to more fun and, most important, to an even better outcome for the speakers: They will get their message across.



Developing large-scale applications in Python

Presenter: Marc-André Lemburg
Track: Python Frameworks
Day: Monday, June 27
Time: 09:00 - 10:00
Location: VR

Python is widely and somewhat inaccurately referred to as a scripting language. While Python is an ideal platform for small scripting tasks, it does in fact cover all the concepts needed for large scale object oriented application development. However, complex applications bring different challenges.

This talk will draw on the speaker's experience with large scale application development using Python as central implementation language and provide a cookbook approach to many of the the problems you face when designing and organizing complex application frameworks.

The approach has been proven in many real-life projects, ranging from web application servers in varying fields of application to complete finance trading systems. Python has always proven to be a truly good choice with an outstanding time-to-market advantage over other implementation languages such as Java or C++. It has literally made the difference for a number of small software development companies competing against multi-million dollar ventures.

a Python sales pitch

Presenter: Harald Armin Massa
Track: Social skills
Day: Monday, June 27
Time: 09:30 - 10:00
Location: VA

Sales is the oldest business of the universe. Some say other business is older - but hey, that is just a subdivision of sales (Brian Tracy, brilliant Salesman of himself). Successful salesman are more often than not great storytellers, who are able to wrap technical arguments in fine words, often without even knowing that technical arguments. This seminar will provide a rather pythonic way of selling. Combined with fine short stories to improve your persuasive power. Topics include within others:

- exploiting the "WE" while still being strictly honest
- using basic NLP knowledge to leverage FBS
- paint the dynamics of Python in business colours
- translate LOC to ROI

Architecture of a large Zope 3 system

Presenter: Steve Alexander
Track: Zope
Day: Monday, June 27
Time: 09:30 - 10:00
Location: VB



TVTK: A Pythonic VTK

Presenter: Prabhu Ramachandran
Track: Refereed Papers
Day: Monday, June 27
Time: 09:30 - 10:00
Location: VF

The Visualization Toolkit (VTK_) is a popular, open source, high-level visualization library. VTK is implemented in C++ and provides convenient wrappers for Python, Tcl and Java. While the VTK-Python wrappers are very useful and powerful, their API is not "Pythonic". For example, VTK has its own iterator mechanism and array interface and does not support pickling.

TVTK (Traited VTK) is a Pythonic wrapper around VTK-Python implemented mostly in pure Python. The entire VTK class hierarchy is wrapped at build time. TVTK objects support traits_ and may be pickled (with some limitations). VTK iterators are mirrored as Python iterables, properties of VTK objects are represented as traits and Numeric arrays are handled transparently. In addition, every TVTK object provides a default and customizable GUI editor. TVTK also includes a TVTK pipeline browser, a high-level Octaviz_ like API called `mlab`, and Envisage_ plug-ins. This paper describes TVTK in detail and how it works under the hood. We describe how we parse the entire VTK API (of around 800 classes), generate the wrapper classes, enable for their efficient usage and support the various features of TVTK.

.. _VTK: <http://www.vtk.org> .. _traits: <http://www.enthought.com/traits> .. _Octaviz:
<http://octaviz.sourceforge.net/index.php?page=manpages> .. _Envisage:
<http://python.fyxm.net/pycon/2005/papers/81/PyCon-2005.txt>

A Python Framework for Rapid Application Development

Presenters: Katherine Goodwin, Joe Wrigley
Track: Python Frameworks
Day: Monday, June 27
Time: 10:00 - 10:30
Location: VR

Writing data presentation code should consist of no more than painting widgets in a GUI designer. But too many "visual" systems create software with an attractive, polished exterior and gnarled, rotten innards -- rather like a gilded Egyptian sarcophagus. Pity the maintainer who gets to open the lid and root through the guts. Four years ago, John Pinner, CEO of Clockwork Software Systems, snapped. His customers had been served faithfully by low-maintenance bespoke systems, up to two decades old, but built on a now-discontinued platform. He could not bear to replace these with one-size-fits-all "solutions" which had slick interfaces but failed to meet customers' needs. Instead, he started planning how to move the proven systems into the 21st century incrementally, without throwing away the data and business rules on which the old software had been honed.

This talk looks at one area of Clockwork's expertise: taking legacy applications, built on proprietary technology, and evolving them incrementally into modern, extensible, open systems that can tap into the wealth of available open-source tools. It also looks at Thyme, the Python-based application development framework created by Clockwork, by drawing on the cream of open-source modules and stealing all the credit. It has the following qualities: - Cross-platform - Database-independent - Low bandwidth-friendly: runs across dialup connections - GUI agnostic - Extensible: MVC component architecture - Open-source: GPL - Good Sense of Humour - Kind to Animals - XML-RPC-enabled



CPSWorkflow : From stateful to activity based workflows

Presenter: Julien Anguenot

Track: Zope

Day: Monday, June 27

Time: 10:00 - 10:30

Location: VB

Introduction to workflow systems with a description of the stateful and activity based models.

Overview of complex collaborative workflow needs we implemented using CPSWorkflow.(including live demonstration)

CPS and the approach of a workflow driven framework.

Technical presentation of CPSWorkflow and its extensions. We will see how we are reaching the limit of the stateful model in this case and why we need to switch to an activity based model.

Presentation of the next generation workflow framework we are working on for Zope3 based on the wfmc component included within X3.1. We will see how we will be able to address some complex business use cases for collaborative work and as well for ERP systems.

OUTLINE : -----

1 - Introduction to workflows

1.1 : What are they ? 1.2 : Stateful workflows 1.3 : Activity based workflows

2 - Common use cases of the industry with demo

2.1 : Misill use case 2.2 : Messenger use case 2.3 : BRIQUES use case 2.4 : Live demonstration 3 - CPSWorkflow technical review

3.1 : CPS is 'workflow driven' 3.2 : DCWorkflow based 3.3 : CPSWorkflow extensions overview 3.4 : Stack workflows API

4 - ecmworkflow for Zope3

4.1 : Why switching to an activity based model ? 4.2 : wfmc model 4.3 : xpdI 4.4 : Zope3 wfmc component 4.5 : ecmworkflow early design and motivation in a CMS scope 4.6 : suitable for CMS's and ERP's 4.7 : Demonstration

Implementing Continuous Time Simulation Systems in Python

Presenter: Paul J. Nolan

Track: Refereed Papers

Day: Monday, June 27

Time: 10:00 - 10:30

Location: VF



Who else hates selling? How to make your customer want to do business with you.

Presenter: Astrid Hellwig
Track: Social skills
Day: Monday, June 27
Time: 10:00 - 10:30
Location: VA

When trying to find customers for your services, you can either go the traditional way of selling yourself and your services to them. Zillions of people can teach you smart ways of talking and saying the right things, no doubt! Yet, you may still find it dreadful to pick up the phone and try to SELL your services to someone who might be bothered to even listen to you, the sales-person. Therefore, this talk is about an intuitive approach to reaching out to the inside of your customers, to their wishes and their needs. This approach shows you a way how to see and present yourself as a consultant who's first job is to listen carefully to what the customer wants to say. While asking some questions to make your customer talk, you run some sort of an analysis for yourself. The outcome will show you whether this person is "eligible" to become your customer or not. All the while, it is you who is steering the process of selling. It is you who decides whether you want to continue if the customer does not pass one stage of your "test". You control the course of the "interview" since your time is scarce and your marketing budget is limited. While we go through this analysis, a lot of examples will ring a bell in you. In the end, you will be able to see yourself as the person you really are: the consultant, the expert, the service provider and not the sales person who you always dreaded to be. Now, go out there and test it. Enjoy your success!

Why I invented Python

Presenter: Guido van Rossum
Track: Tutorials
Day: Monday, June 27
Time: 11:00 - 11:45
Location: VV

Guido van Rossum is the inventor of Python and BDFL. Python is a language designed by one man, not by a committee ... so, to taste it really, we should listen up to it's inventor.

Please note that this talk is given for the benefit of those following the Python Neopythes track. It is intentionally placed in a small room in order to have an intimate atmosphere. If there is extra seating available after the Neopythes have taken theirs, it is ok to join the talk. Please respect this, so we can give newcomers to Python a really nice experience.

XML information extraction in Python

Presenter: Brian Quinlan
Track: Python Frameworks
Day: Monday, June 27
Time: 11:00 - 11:45
Location: VR

There are a bevy of Python libraries that can be used to extract information from XML documents. This talk will examine several popular means of processing XML documents in Python including ElementTree, DOM, SAX, and XPath. The advantages and disadvantages of each will explained in the context of solving different information retrieval problems.



This talk is targeted towards people with at least a basic knowledge of Python, who know what XML is (i.e. that it looks like HTML but the tags are different) and who have some need to find information inside XML documents.

Making Money on Plone - Why The Model Works

Presenter: Martin Aspeli

Track: Social skills

Day: Monday, June 27

Time: 11:00 - 11:30

Location: VA

Specifically, the following issues will be given consideration:

- o What is the business model underpinning Plone? Why is it open source at all?
- o What role does the technology stack, underpinned by Python, play in this model?
- o What role does the community and its processes play? How have these evolved? Where are they going?
- o From philanthropy to profit - the balance and path between getting involved and making money
- o What can other open source projects and businesses learn from Plone?

Document Library: a Zope 3 based Document Management System

Presenter: kit blake

Track: Zope

Day: Monday, June 27

Time: 11:00 - 11:30

Location: VB

The application is built using Zope 3 technologies. The management environment is Zope 3 native. Public listing display is rendered by Silva, running in Zope 2, but any OAI-PMH capable web application or CMS could make use of the same information.

The ability to manage large quantities of electronic documents is becoming steadily more important for large organizations. More and more documents become available or are produced in electronic form. New legal requirements for document management are also coming into being, such as the UK Freedom of Information Act.

In contrast with (web) content management systems, document management systems focus on managing documents in an organization produced for purposes other than web publication. A document management system provides check-in, check-out, storage and retrieval of electronic documents, often in the form of word processor files or PDFs. Ongoing maintenance of documents is eased and guided by delegation, workflow, and notification.

The presentation will outline the goals and specification for the archive, demonstrate management of documents for Authors and Librarians, and show public display and search. In the last third of the presentation Infrae's experiences using Zope 3 will be covered.



CPS Platform Framework -

Presenters: Julien Anguenot, Stefane Fermigier, Florent Guillaume, Jean-Marc Orliaguet, Tarek Ziade

Track: Refereed Papers

Day: Monday, June 27

Time: 11:00 - 12:00

Location: VF

NB: draft paper at http://www.cps-project.org/sections/about/cps_overview will be completed and reworked for the conference if accepted.

AlphaFlow: Background and Technical Details

Presenter: Christian Theune

Track: Zope

Day: Monday, June 27

Time: 11:30 - 12:00

Location: VB

This talk will go into the technical details of the engines concepts and how those influence the modelling of specific workflow patterns. Especially the hybride properties of activity-based and document-oriented workflows will show how to mix the best of both worlds together.

It will also show how the engine's infrastructure allows application developers to build their own toolbox of specialised and re-usable activities.

The Python Community in Brazil

Presenter: Rodrigo Senra

Track: Social skills

Day: Monday, June 27

Time: 11:30 - 12:00

Location: VA

This speech will give an overview about: the size and growth of the Brazilian Python Community, our first PyConBrasil held last April, present some cases showing how Python/Zope/Plone are being adopted in Government and Corporations. Last but not least, I intend to present some interesting projects developed by Brazilians, that take part in the international community.

Taste Python, part 1

Presenters: Rob Collins, Michele Simionato

Track: Tutorials

Day: Monday, June 27

Time: 11:45 - 12:30

Location: VV

taste the spirit of Python. A very interactive training seminar for real newbies and those just new to python, and all others also. We shall start with some "hello world" within Python. Through extensive staffing with two experienced high profile trainers we will be able to accomodate this seminar dynamically to the knowledge level of the audience, transferring the highly dynamic nature of Python to the concept of trainings.



So for all who need some static bindings first, we will have

a) some real "hello world" stuff b) tricks and powers of the interactive console c) dynamic modules taken from - web scraping with python (beautiful soup, urllib..) - database access (WITHOUT sqlalchemy, which is presented separately) - some basic system Administration Stuff - do a PDF

CMFEditions Versioning Framework for CMF/Plone

Presenters: Alberto Berti, Vincenzo di Somma, Grégoire Weber

Track: Zope

Day: Monday, June 27

Time: 12:00 - 12:30

Location: VB

In the first part of the presentation we talk about what CMFEditions is for and what it isn't for (yet):

- version CMF content types and AT content types - managing folderish content types - how Archetypes references are handled - how to make a content type version aware

The second part of the presentation is about the pluggable architecture of CMFEditions and the reasons behind it. An important topic will be how CMFEditions achieves managing references between content objects.

The third part is a demonstration that shows the presented functionality.

Community and Cash panel

Presenters: Martin Aspeli, Martijn Faassen, Astrid Hellwig, Harald Armin Massa, Rodrigo Senra

Track: Social skills

Day: Monday, June 27

Time: 12:00 - 12:30

Location: VA

MayaVi-2: The next generation

Presenter: Prabhu Ramachandran

Track: Refereed Papers

Day: Monday, June 27

Time: 12:00 - 12:30

Location: VF

In this paper we describe MayaVi-2, the next generation of MayaVi_. MayaVi_ is an open source, scientific data visualization tool for scalar, vector and tensor field data in 2D and 3D. It is written entirely in Python. MayaVi uses VTK_ (the Visualization Toolkit) and provides a Tkinter based GUI. The 1.x releases of MayaVi suffer from several problems. For example, the internal design is not clean and easily scriptable. The GUI does not use an MVC based design, is not easily changed and is not very pretty.

MayaVi-2 is an Envisage_ plug-in and addresses several of these issues. It uses the Enthought tool-suite in the form of traits_, TVTK_ and Envisage_. Thus, bulk of the GUI code comes for free and an MVC design is forced upon us! The MayaVi-2 code base is therefore smaller, cleaner, easier to script and extend and the resulting application also looks prettier. Being an Envisage plug-in makes it



easy to use other plug-ins with MayaVi-2. This paper outlines the basic design and structure of MayaVi-2, how it works and also highlights the advantages of using the Enthought tool-suite. The paper also presents a simple idea that allows for a general and flexible persistence implementation for complex applications.

.. _MayaVi: <http://mayavi.sf.net> .. _VTK: <http://www.vtk.org> .. _TVTK: <http://www.enthought.com/svn/enthought/branches/converge/tvtk/doc/README.txt> .. _traits: <http://www.enthought.com/traits> .. _Envisage: <http://python.fyxm.net/pycon/2005/papers/81/PyCon-2005.txt>

A Virtual File System for Python

Presenter: Juan David Ibáñez Palomar
Track: Python Frameworks
Day: Monday, June 27
Time: 12:00 - 12:30
Location: VR

At the core of itools there is the resource/handler model. This talk will explain the three packages that make it up:

- itools.uri provides an API to manage URIs, which are used to identify and locate resources;
- itools.resources is an abstraction layer over resources that let to manage them with a consistent API, independently of where they are stored;
- itools.handlers provides resource handlers, they are non persistent classes that add specific semantic to resources, for example there are handlers for file formats like XML, CSV, PO, etc..

The programming model behind will be explained, and we will overview the handler classes available out of the box.

The world according to Leo

Presenter: Edward Ream
Track: Education
Day: Monday, June 27
Time: 14:00 - 15:00
Location: VA

Leo's world is the world of understanding and using relationships in complex data. Leo can transform how you program, work, think and learn.

1. Leo outlines are quasi-hierarchies; they allow arbitrarily many reorderings of data to exist within a single outline. Reorderings create new views of data. Clones are the feature that make reorderings possible.
2. Views (reordered information) are exact visual analogs of mathematical relations (i.e., conceptual relationships). Leo's views are lists of nodes; mathematical relations are just tuples. Leo creates a world in which seeing relationships is natural. This is the key to understanding, using and managing complex data. In particular, views simplify programmer's work flow.
3. Leo integrates outlines with literate programming tools. Scripts can use outline structure explicitly. Conversely, outline structure is an essential part of programs and scripts. (Scripts `_are_` outlines) With Leo, program = data + algorithm + outline structure. This is a fundamental addition to programming.



4. Leo redefines literate programming. Leo outlines simplify and clarify literate programming webs, and make narrative exposition unnecessary. User-defined views (reorderings) are the key to understanding. Typography has almost no information content. Leo outlines are essentially user-definable indices and tables of contents.

The talk will be given by Edward K. Ream, the creator of Leo

Ixml, a Pythonic binding for libxml2

Presenter: Martijn Faassen
Track: Python Frameworks
Day: Monday, June 27
Time: 14:00 - 14:30
Location: VF

The C libraries libxml2 and libxslt have huge benefits, such as standards-compliant XML support, full-featured, actively maintained by XML experts and fast. fast! FAST!

These libraries already ship with Python bindings, but these Python bindings have problems. In particular, they are very low level and C-ish (not Pythonic), underdocumented and huge, you get lost in them, UTF-8 in API, instead of Python unicode strings, they can cause segfaults from Python, programmer has to do manual memory management.

Ixml is a new Python binding for libxml2 and libxslt, completely independent from these existing Python bindings. Its aims are a Pythonic API, documented, uses Python unicode strings in API, safe (no segfaults), and no manual memory management needed.

Ixml aims to provide a Pythonic API by following as much as possible the ElementTree API. We're trying to avoid having to invent too many new APIs, or you having to learn new things -- XML is complicated enough.

Taste python, Part 2

Presenters: Rob Collins, Michele Simionato
Track: Tutorials
Day: Monday, June 27
Time: 14:00 - 14:30
Location: VV

The second part - we even give you a break, and are sure that you will be back!

Some tasty bites of the DITMs (Dynamically Integratable Training Modules) we prepared: - A superfast toor of IPython - reStructuredText, how to make Web pages and PDFs - Twill, i.e. how to interact with the Web the easy way

Managing a distributed team

Presenter: Steve Alexander
Track: Business
Day: Monday, June 27
Time: 14:00 - 14:30
Location: VR

I, and the others at my company, have developed a various ways of working with a distributed team, including a process for specifying new features, an approach to time management, use of gtimelog, a



format for irc meetings, and various other things.

Distributed teams are becoming more common. But, there is a lack of experience in how to manage these teams effectively.

I propose to talk about some of the things we do that work for us, and try to explain why.

State of Plone - Plone Grows Up

Presenter: Alexander Limi
Track: Zope
Day: Monday, June 27
Time: 14:00 - 14:30
Location: VB

With the release of Plone 2.1, Plone has grown up - and co-founder Alexander Limi will guide you through the latest developments in the Plone world - and give you a peek of what to expect in the future. Other topics are Zope 3 integration, the next evolution in content management and what's going on behind the scenes in the Ploniverse.

The itools XML infrastructure

Presenter: Juan David Ibáñez Palomar
Track: Python Frameworks
Day: Monday, June 27
Time: 14:30 - 15:00
Location: VF

We will start with the low level event driven parser, which will be compared to expat in terms of the API. Second, we will look at how the XML namespace handlers provide type information that we use to deserialize node values.

Next, the resource handler for bare XML documents, which provides a DOM-like API, will be examined. Then we will see the specific support for XHTML documents, specially the tool chain provided by itools to translate them.

To close the talk we will see the Simple Template Language, probably the simplest and fastest in town.

Batteries included - Introduction to the Python Standard Library

Presenter: Brian Quinlan
Track: Tutorials
Day: Monday, June 27
Time: 14:30 - 15:30
Location: VV

Python is often described as "batteries included" - meaning that it has an extensive standard library. This talk will attempt to provide an understanding of some of the more than one hundred modules in this library.

The talk will be divided into two parts. The first part will provide a brief survey of many of the most important library modules. The second part will take a more in-depth look at some of the more complex and interesting modules e.g. network communications, date/time processing.



Chandler

Presenter: Heikki Toivonen
Track: Business
Day: Monday, June 27
Time: 14:30 - 15:00
Location: VR

Chandler aspires to be an innovative open source personal information manager (PIM). In addition to being written in Python, Chandler uses several other open source libraries, like BerkeleyDB, M2Crypto, Twisted, PyLucene, and wxPython/wxWidgets. Chandler is designed to be an extensible PIM. Chandler's unit of extensibility is called a parcel, and Chandler's "built-in" functionality is itself composed of parcels. Internally, Chandler is designed as layers of frameworks which provide applications functionality to parcels. Parcels communicate with each other via the data in the Chandler repository.

There are several opportunities to get involved with the Chandler project:

1. Many developers will be interested in producing their own parcels. These parcels can extend Chandler to deal with new data types and new user interfaces to that data. These parcels can leverage any data type in the Chandler repository, whether the data type is supplied by the base Chandler application or by another parcel.
2. Some developers will be interested in extending/improving/bugfixing the core application frameworks of the Chandler system, or doing QA or documentation for the project.
3. Businesses might be interested in acquiring alternative licenses to the Chandler code base, similar to the MySQL model.

The goal of the presentation and paper is to introduce Chandler and give pointers to allow anyone to develop a parcel that extends the Chandler user interface.

Make Poverty History with Plone

Presenter: Duncan Booth
Track: Zope
Day: Monday, June 27
Time: 14:30 - 15:00
Location: VB

This talk will demonstrate some of the products which have come out of the Oxfam/ZEA collaboration, looks at how they fit within the larger Plone community, and how Oxfam's use of Plone may develop in the future.

Products which have come partly or wholly from this collaboration include Kupu, Plone Multisite, Versioning support in CMFEditions, Improved multi-language support (LinguaPlone), and the ability to use Plone's search across multiple sites, not all of which run Plone at all (Indexing).

Teaching computational engineering using Python

Presenter: Hans Fangohr
Track: Education
Day: Monday, June 27
Time: 15:00 - 15:30
Location: VA



EuroPython Conference

Computers are increasingly used for a variety of purposes in engineering and science including control, data analysis, simulations and design optimisation. It is therefore becoming increasingly more important for engineering students to have a robust understanding of computer hardware and software. Here we address the software side and outline the difficulties to teach the very basics of programming and numerical methods in an academic context.

We distinguish between two phases in the process of converting a given problem into a computer program that can provide a solution: (i) finding an algorithmic solution and (ii) implementing this in a particular programming language. It is argued that it is most important for the understanding of the students to perform the first step whereas the actual implementation in a programming language is of secondary importance for the learning of problem-solving techniques. This implies that -- at least initially -- a language should be used that is easy to pick up by the learner.

We describe our experiences with using C, MATLAB and Python as teaching languages for undergraduate first and second year engineering students.

Versioning and relation management in a Zope ECM

Presenter: Florent Guillaume
Track: Zope
Day: Monday, June 27
Time: 15:00 - 15:30
Location: VB

This talk studies the design and implementation of existing versioning systems in Zope frameworks. It presents versioning concepts in terms of relations between objects. CPS Proxies, Plone Staging, Silva, Occams, Zope Version Control and others are examined. The implementation of relations in SchoolBell, Zsemantic (RDF), among others, are also examined.

We then present a framework for relation and versioning management in Zope 3 that allows the following to be implemented efficiently:

- User-controlled relationships between documents (references)
- Versioning of documents
- Versioning of sets of related documents
- Staging
- Translations of documents seen as alternative versions

A demo will show these features in a Zope 3 system, and how they work together.

What is Open Source Business?

Presenter: Paul Everitt
Track: Business
Day: Monday, June 27
Time: 15:00 - 15:30
Location: VR

Customers like the benefits of open source software. Should they look for these benefits in their service providers? This talk discusses a framework for defining an open source business, then evaluates Zope Europe Association in terms of this framework.



CAPSconference

Presenter: Jacob Hallén
Track: Python Frameworks
Day: Monday, June 27
Time: 15:00 - 15:30
Location: VF

CAPS is an Enterprise Systems Platform, entirely written in Python. The core parts are only about 20 000 lines of code, while clients and applications add another 100 000 lines or so. CAPS allows you to develop tailor made applications in a very short period of time, and we will show you how this is done. The base Europython system was built by one developer who had never seen the system before in less than 6 weeks. This system handles tracks, talks, registration, accomodation, merchandise, billing, and general issue tracking. It has a GUI interface, a Web gateway and email integration.

A gentle introduction to Python's way

Presenter: André Roberge
Track: Education
Day: Monday, June 27
Time: 16:00 - 17:30
Location: VA

Imagine that writing a complex computer application is like building a house: building materials (bricks, wood, windows, etc.) are the standard libraries; mortar, nails, screws are the higher-level functions written by the programmer in an attempt to glue together the building materials; molecules from which the mortar, nails and screws are made are the basic instructions that the so-called higher level language (C, Java, C++, etc.) provide. Atoms are the machine-code instructions.

In this, admittedly somewhat stretched analogy, computer programmers are people who are first taught chemistry in order to later become carpenters. No wonder learning computer programming can seem to be such a daunting task!

In order to facilitate learning programming, a number of so-called « mini languages » [1] have been designed over the years, many of them inspired by Richard Pattis' book Karel the Robot, a Gentle Introduction to the Art of Programming, written in 1981. In this book, Pattis introduces the main concepts of sequential programming (including loops and decisions, but not variable assignments) using the paradigm of instructing a robot capable of only four basic actions (turning left, moving one step forward, picking up and putting down beepers). Through the "magic" of programming, the robot "learns" to combine those four basic actions in order to accomplish tasks of increasing complexity.

Mini languages are used successfully today to teach the fundamentals of computer programming. Their main strength is their limited « vocabulary » and simplified syntax. The same limited vocabulary and simplified syntax unfortunately create a significant gap when the time comes to graduate to learning a « real » language.

RUR-PLE attempts to bridge the gap between a typical mini language and a « real » computer language: Python. In fact, because of Python's strength, RUR-PLE provides an environment where both the traditional procedural approach and the more modern Object-Oriented approach can be introduced in a very natural way. To our knowledge, no other mini language gives the same flexibility, since none other is based on Python!

This talk should be of interest to educators interested in teaching Python, to beginners who want to learn the basics of programming using Python in a fun tutorial, as well as to those who are wondering if they should learn Python [2].



[1] Mini-languages: A Way to Learn Programming Principles, (<http://www.contrib.andrew.cmu.edu/~plb/papers/minilang.html>) [2] The answer is « Yes! »

Enabling bare Python as universal connector for ad-hoc networks

Presenter: Theo F. de Ridder
Track: Python Frameworks
Day: Monday, June 27
Time: 16:00 - 16:30
Location: VF

GlassDict offers an amazingly simple and powerful architecture for playing with ad-hoc networks. It models clients and servers as networknodes with just four classes: Partner, Code, Context, and Path. Partner handles asynchronous streams of Code instances for each connection. Code pickles data-objects for arbitrary data-exchange, and marshalls code-objects for execution in server mode. A server Context hosts the global impacts of calling Code instances, a client Context hosts callback functionality for received results of calls or published impacts. Path indicates a changed value as impact in Context and is used for publish/subscribe specifications. There are some unusual characteristics of GlassDict as a server. Its Context is empty when started and dynamically and transparently updated and controlled through code compiled and sent by its clients. Its single autonomous activity is publishing impacts to subscribers. This rather restricted usage of Python facilities in the server role enables challenging opportunities for stripped-down interpreters within very small networknodes.

Batteries included, part 2

Presenter: Brian Quinlan
Track: Tutorials
Day: Monday, June 27
Time: 16:00 - 16:30
Location: VV

Batteries included, continued from previous session

In Theaters Now!--How Python Runs the Largest Independent Cinema Chain in the UK

Presenters: Richard Cooper, Remi Delon, Kevin Manley
Track: Business
Day: Monday, June 27
Time: 16:00 - 17:00
Location: VR

Arts Alliance Media (AAM) used Python to build a mission critical enterprise system (Newman) for City Screen Cinemas, the largest independent cinema chain in the UK. This system manages all backoffice operations (film data, distributor deals, show schedules, products and pricing, transaction history, customer data) and frontoffice operations (sales of tickets, kiosk and bar items, memberships) for the cinema and is the backbone of their operations. The distributed system uses a sophisticated replication scheme to keep the cinemas and central server synchronised while being resilient to network problems. New shows published through the backoffice become available for sale immediately at the boxoffice. Sales data from each transaction is replicated to the central server in near realtime. The system consists of a number of components: - Branch server: a multithreaded xmlrpc application server that manages the central database - Backoffice server: a multithreaded HTTP server implemented in CherryPy/CherryObjects which provides a web UI for the branch server -



EuroPython Conference

Venue servers: multithreaded xmlrpc application servers that manage transactions at each cinema boxoffice - GUI point of sale: wxPython-based touchscreen app which talks to the venue server. This is the main selling interface at the cinema boxoffice and phone center. - Web point of sale: another multithreaded CherryPy HTTP server, which talks to the branch and venue servers and enables membership and ticket sales over the web.

The overall system is one of the largest we know of implemented in Python, consisting of over 100K lines of code and 50K lines of XML metadata (the system is largely data-driven). The central SQL Server database consists of 210 tables, 581 views, and 105 stored procedures. This system is currently deployed at all 18 City Screen cinemas, plus the central phone sales center. The system processes on average 65K transactions worth a total of £420K (615K) per week. This talk, given by the architect and developers (a team of 3 people in total), will discuss the technical challenges and rewards of developing a large scale distributed application in Python. We will also discuss our development process, and our experiences working with the standard library and 3rd party tools including xmlrpclib, cElementTree, mxODBC, CherryPy, wxPython, and others. An important part of the discussion will focus on the challenges of convincing management to sign off on using Python for such a large, ambitious, and risky project.

Zope Topic Maps - building metadata-based portals with open standards

Presenters: Andreas Rübner Johnsen, Arnar Lundesgaard

Track: Zope

Day: Monday, June 27

Time: 16:00 - 16:30

Location: VB

Topic map-based, or ontology-driven portals have become a hot topic in the Norwegian market and several quotations have a "MUST HAVE" requirement for Topic Map support.

Zope Topic Maps (ZTM) is a CMF based implementation of the Topic Map model. Topic Maps being a semantic web technology that has proven benefits to developers, editors and visitors of websites. ZTM is now in use by several customers with professional needs.

This presentation is about how and why we have combined Zope/CMF and Topic Maps:

- it's about why Topic Maps solve many of the challenges that most content management solutions face.
- it's about why Zope is a very suitable platform for Topic Maps solutions.
- it's about how Topic Maps has been implemented on Zope/ZODB/CMF
- it's about how developer, information architect or even the customer can easily create and maintain the information structure of the site.

Pulling Java Lucene into Python: PyLucene

Presenter: Andi Vajda

Track: Python Frameworks

Day: Monday, June 27

Time: 16:30 - 17:00

Location: VF

OSAF's flagship project, Chandler (<http://www.osafoundation.org>), is a personal information manager. As such, it needs the ability to run unstructured full text queries over arbitrarily large repositories of text.



EuroPython Conference

There are not that many open source text search engines available. Lucene is considered among the better ones and it is licensed under the Apache license, both of which make it a very attractive solution.

But it is written in Java.

For various reasons OSAF would prefer not to ship Chandler requiring a fully fledged JVM, which made a Jython (<http://www.jython.org>) or JPyype (<http://jpyype.sourceforge.net>) based solution undesirable.

There are several ports of Java Lucene to other languages:

- C++/CLucene (<http://sourceforge.net/projects/clucene/>) While 4 times faster than the original java version, CLucene, like most other ports, is behind, and like most C++ projects, comes with its own set of bugs.
- Python/Lupy (<http://www.divmod.org/Home/Projects/Lupy>) The advantage of a fully native python port is lost by an order of magnitude worse performance than the original java version.
- .net/dotLucene (<http://openlucene.net>) While not behind on the porting curve, dotLucene only swaps one problem for another, it requires a .net VM.

OSAF wanted something that would be simple to deliver, ideally no more than a handful of shared libraries. In theory, it should be possible to put Java Lucene, GNU's java compiler gcj and SWIG together into a native shared library built as a python extension. It would have to run on Linux, Mac OS X and Windows, be stable, support threading, and the text indexes would have to be part of Chandler's repository, sharing transactions.

The PyLucene project was started in December 2003 with a number of unresolved challenges down the road. It really started as an exploration, by getting acquainted with several projects:

- Java Lucene (<http://jakarta.apache.org/lucene/docs/index.html>) is a decently written 100% Java code base using very little arcane java features, no GUI, no threading in the core, and putting a lot of effort in maintaining backwards compatibility with JVMs going back to 1.2.x releases. The Java Lucene project is being actively developed by a sizable community of volunteers.

Would Lucene be using java constructs triggering bugs in gcj ? Would Lucene be using as yet unsupported APIs in libgcj ?

- GNU's gcj compiler (<http://gcc.gnu.org/java>) is a derivative of their C++ compiler with a massive Java Runtime library, libgcj, a garbage collector, Boehm-GC, claiming support for most standard JVM APIs up to 1.4.x. Like Java Lucene, gcj is under active development as well. GNU's gcj compiles java classes from sources or bytecode into a native shared library, making the code available as if these classes were C++ via the Compiled Native Interface (CNI), an alternative to the Java Native Interface (JNI).

Would there be proper support on Windows ? Would the bugs be just too unbearable ? How would Python threads and Java threads integrate ?

- SWIG (<http://www.swig.org>), is a software development tool that connects programs written in C and C++ with a variety of high-level programming languages including Python. SWIG generates a lot of boilerplate code from source written in a special C-like syntax.

Would SWIG let itself be bent to accommodate something very much looking like C++ but whose memory is managed rather differently ?

- Berkeley DB's C library integration with Java is a wrapper around its C library just like the `_bsddb` python extension (<http://sourceforge.net/projects/pybsddb>).

Would it be possible to pass db objects such as transactions, databases or environments from python to compiled java such that PyLucene and Chandler were able to use the same transactions when persisting data ?



EuroPython Conference

During this talk we will explore the solutions we developed for PyLucene, including:

- Compiler support on various operating systems: Mac OS X, Windows, Linux - Building PyLucene from sources: getting all the pieces together - Thread integration issues: the libgcj garbage collector's needs - Memory management differences: ref counting versus garbage collection - 'Extending' java classes from python via wrappers: reverse SWIG - Footprint issues: statically linking libgcj with the python extension - Code samples: examples of using PyLucene in Python code

We'll also discuss future developments such as:

- How to apply the same techniques to other projects or other languages (Ruby, Perl, etc)

The PyLucene project was made into a separate project in June 2004. PyLucene is hosted by OSAF and licensed under the MIT license (<http://www.opensource.org/licenses/mit-license.php>). PyLucene is under active development and has a small community of regular users. It has been deployed in a handful of controlled projects.

PyLucene's homepage is <http://pylucene.osafoundation.org>.

Using Python to glue Windows and AS/400 applications together

Presenter: Johan Lindberg
Track: Tutorials
Day: Monday, June 27
Time: 16:30 - 17:00
Location: VV

Background:

I've spent the last year or so working on Office automation and lightweight integration (via OLE/COM) using IBM Personal Communications tool (talking to an AS/400 machine).

We've used Python rather extensively as "glue" between our AS/400 and first and foremost the MS Office suite but also other PC applications. We've used it to automate several tasks that was previously rather time consuming, repetitive and dull (which makes them error-prone).

Examples: - sales reports (from core system to Excel), - customer letters (from core system to Word), - information about insurance "events" to sales people (from core system to Outlook).

CPSMailAcces : a webmail application in CPS3 using Zope 3 and Five

Presenter: Tarek Ziade
Track: Zope
Day: Monday, June 27
Time: 16:30 - 17:00
Location: VB

CPSMailAcces is an enhanced webmail application that offer users integrated features like spam detection or advanced email search.

First of all this talk will present the architecture of the tool and how : - it uses ZODB persistence - it has been thought to run into CPS3/Zope 2 environment with the maximum use of Zope 3. - it interacts with an IMAP server

Next it will present some its enhanced features, based on some products such as Zsemantic for semantic searches in mails and Reverend, for bayesian classifications.



Last but not least, a live demo will present the whole tool within CPS3 and finally a list of the steps needed to make it work in a pure Zope 3 environment.

Zope/Zeo in Real Life Deployment

Presenter: Pravir Chawdhry
Track: Zope
Day: Monday, June 27
Time: 17:00 - 17:30
Location: VB

Python on a Phone

Presenter: Jukka Laurila
Track: Misfits
Day: Monday, June 27
Time: 17:00 - 17:30
Location: VR

The past, present and a glimpse into the future of the Nokia port of Python to the Series 60 smartphones. What is it? What can you do with it? Why did we do it? I'll also discuss some of the ins and outs of porting the interpreter and wrapping the native interfaces into Python modules, and show some demos of the latest features.

Notes to the organisers: Since the demos will involve Bluetooth communication between the phone and the PC I will have to use my own laptop for them. Also, to show the demos live it would be `_very_` convenient to be able to show the phone screen and the laptop screen simultaneously, so having a second projector and a camera would be nice. Another solution would be to use a USB webcam connected to the laptop. Please inform me beforehand what equipment you have available so I know what to bring with me.

A powerfull Cell within the batteries: decimal for beginners

Presenter: Facundo Batista
Track: Tutorials
Day: Monday, June 27
Time: 17:00 - 17:30
Location: VV

I'll explain what is Decimal, how to use it, and when to use it. The idea is to let the reader to feel confident about this new data type in order to start using it.

I'll start with the detail of the Decimal intrinsic structure (floating point, bounded precision), the major differences with binary floating point, and other advantages.

Later I'll show how to use the module, from the simplest way (just use it) to a more advanced interaction (from showing its methods to explain how to use the Context, including rounding methods, signals, etc.).

Finally, I'll present some cases to show when use and not use Decimal instead of binary floating point.



Unit-testing PyGTK applications

Presenter: Gintautas Miliauskas
Track: Python Frameworks
Day: Monday, June 27
Time: 17:00 - 17:30
Location: VF

Frameworks exist for testing GUI applications, but most of them are of the functional type, i.e., they run the application being tested in almost real-life conditions. Even though such testing is quite reliable, it usually limits direct access to the the running program, making it hard to use common techniques (e.g., stub and mock objects) and to test internal components. As a consequence, tests grow complex and therefore complete test coverage is hard to attain. One of the solutions to this situation is to stub out the GUI library itself. That is no trivial task, though in many cases even a very limited set of stubs is quite helpful if a few neat tricks (read: dirty hacks) are employed. An immature, but already practical implementation of a such a collection for PyGTK in the form of a library will be presented. While such an approach to testing is significantly less reliable than functional testing, it requires much less overhead and lets the programmer sidestep GUI code providing direct access to inner application components if needed. The presentation will also briefly cover a few of the other testing tools available, including functional testing tools (LDTP) and unit-testing helpers for other GUI toolkits (PyGUIUnit, NDtestmaker).

Experience with Python for Series 60 in the PDIS project

Presenters: Tero Hasu, Ken Rimey
Track: Python Frameworks
Day: Tuesday, June 28
Time: 09:00 - 09:30
Location: VA

We launched the Personal Distributed Information Store (PDIS) project at the Helsinki Institute for Information Technology (HIIT) in 2003 to study next-generation data synchronization, with the vision of enabling people to store replicas of their data on several electronic devices, and several computers, and to keep these replicas in sync.

The project has developed the PDIS repository, an update-anywhere replicated XML database that runs on Series 60 mobile phones as well as personal computers. It allows any device to be synchronized with any other at any time and is ideal for storing personal databases (contacts, calendars, etc.) and metadata for digital mediafiles.

The software is written in Python and is available from <http://pdis.hiit.fi/> under the MIT License. PDIS is perhaps the most ambitious project undertaken using Python for Series 60, which Nokia released publicly in December 2004.

In addition to our raw source tree, we have also made a few components available separately as free-standing packages, including the following:

Miso - a library providing access to a number of native phone APIs from Python for Series 60. For example, it let's you fetch a JPEG from the camera by simply calling `take_photo()`.

Pyexpat for Series 60 - a port of the expat XML parser to Python for Series 60 enabling the standard Python xml package to be used in Python applications running on Series 60 phones.

PDIS XPath - a pure-Python XPath evaluator based on the elementtree toolkit.



Series 60 Compatibility Library - a wxPython-based implementation of some of the user interface toolkit APIs defined by Python for Series 60. It enables running many Python for Series 60 scripts under Mac OS X, Linux, or Windows.

Breaking the Rules of Commercial Software Development : Why and How we used Python to Produce a New Deal in Payroll Software.

Presenter: John Pinner
Track: Business
Day: Tuesday, June 28
Time: 09:00 - 10:00
Location: VB

This talk covers two areas, business and technical:

Business-wise, it will examine the reasons for developing PayThyme, the aspirations behind its business model, and the work involved in gaining commercial acceptability and Inland Revenue accreditation. The resolution of any conflict between the need to provide a fully-supported, credible, and income-generating product, and the desire to release it as genuine Open Source software, will be discussed.

Technically, the product itself will be demonstrated, and used as example software in showing the tools and methodologies used in developing GUI applications with Python, PyQt and Qt in the Thyme development environment. This will cover creating database table definitions (with information to speed GUI screen design), the automatic generation of GUI screen forms and maintenance programs from the table definitions, modifying the screens with Qt Designer, and enhancing the generated programs, with particular reference to using Thyme's data-aware custom widgets.

club travel invoices with Python-Excel

Presenter: Guy Dalberto
Track: Tutorials
Day: Tuesday, June 28
Time: 09:00 - 09:30
Location: VR

After each competition or social event we prepare an Excel worksheet with info about what was paid, who paid it and what each person should pay. About once every two monthes, I run a small Python program that reads invoice data from those worksheets, and another worksheet with info about emails, adresses and family relations. It then creates for each person that must pay something a PDF file that recapitulates the reasons and amounts. Using Excel to specify the event sharing rules make the system very flexible. Python adds the power. I have less paper work to do.

FSF Europe - Free Software - Free Society

Presenter: Henrik Sandklef
Track: Social skills
Day: Tuesday, June 28
Time: 09:00 - 09:30
Location: VF

0. Introduction

whoami



EuroPython Conference

1. GNU + FSF

What is GNU History Free Software Directory Unesco 2. FSF Europe

History Current activities eu / bxl un / wsis

3. Software industry

History Software patents Trusted computing What you can do

4. FSF Europe Fellowship

what is the fellowship How do you join

5. Take action

6. Links

Open Space

Track: Misfits

Day: Tuesday, June 28

Time: 09:00 - 10:30

Location: VV

If you want to have a place to have a BOF (Birds of a Feather session) or have some other informal session, this is where you can do it. The Cafeteria is also available for informal sessions, but a bit noisier.

Kamaelia

Presenter: Michael Sparks

Track: Python Frameworks

Day: Tuesday, June 28

Time: 09:30 - 10:15

Location: VA

Scalable concurrent systems do not have to be hard - this is a fundamental aim of the Kamaelia project. The real world contains large numbers of items we interact with daily on a concurrent basis, be it every electronic item we own - from kettles to computers through to simple things like roads, office workers and dance partners.

Concurrency in software often takes 3 approaches - process, thread or state machine based. For portability and scalability, state machine based systems are normally the preferred approach. However, even best of breed systems often have a steep learning curve and can be hard for beginners to learn, and can be difficult to debug.

Kamaelia represents concurrency as components communicating along unidirectional linkages. Components are python generators embedded into a class augmented by inboxes and outboxes. New components can be written sequentially, and then made concurrent slowly and simply in a controlled fashion.

Building systems is pretty much like using good old fashioned unix pipelines. You choose the components that do the work and string their outputs and inputs together. You can build pipelines or graphs which may also change dynamically. Components also nest simplifying systems. This approach also encourages high levels of reuse.



Using python generators seems to result in performance similar to that of traditional state machines, with an ease of use similar to that of process based concurrency. BBC R&D is building Kamaelia as a testbed for developing open large scale internet media delivery protocols. However, this talk will focus on the core technology allowing others to use the technology for other uses.

CherryPy tutorial

Presenter: Remi Delon
Track: Tutorials
Day: Tuesday, June 28
Time: 09:30 - 10:15
Location: VR

CherryPy is a pythonic, object-oriented web development framework. It allows developers to build web applications in much the same way they would build any other object-oriented Python program. This tutorial will show how to build simple websites with CherryPy: - Getting started - Organizing your site and source code - Using a templating language - Using sessions - Generating web forms - Deploying four site for production

The Creative Commons Project

Presenter: Mathias Klang
Track: Social skills
Day: Tuesday, June 28
Time: 09:30 - 10:00
Location: VF

The Creative Commons (CC) project was developed to help creative people share copyrighted material. This was done by creating a licensing scheme that could be applied, understood and communicated by non-lawyers. The project was launched in 2001 and took inspiration from the GNU General Public License. The first stage was the release of a set of copyright licenses free for public use. Following this CC has developed a Web application that helps people dedicate their creative works to the public domain or retain their copyright while licensing them as free for certain uses, under certain conditions.

The success of CC has led to its spread over many different legal jurisdictions and its application in many new areas. The purpose of this talk will be to briefly explain the CC licensing system and the use of meta-data in the licensing. In addition to this the internationalisation of CC will be presented from the viewpoint of the work of CC in Sweden.

Electronic Measurement Automation

Presenter: Guy Dalberto
Track: Business
Day: Tuesday, June 28
Time: 10:00 - 10:30
Location: VB

Zip-formatted files hold complete instruments configurations, measured data and events journal. Data can be exported in Excel format directly used from preformatted results report. Windows-explorer context menu starts file sub-trees automatic measure sequences. Meta-configuration files specify which parts of those configurations are displayed by the the simple common GUI interface used to load/run/save the measurements. Sweeps specify single parameter ranges (i.e from 2.4 to 2.6 Volts with 0.0001 Volt step). The interface process controls via TCP/IP the measurement process. All Python code is validated by UnitTests.



Licensing panel

Presenters: Mathias Klang, Henrik Sandklef
Track: Social skills
Day: Tuesday, June 28
Time: 10:00 - 10:30
Location: VF

-

2 + 3 = ? -- A hands-on approach to slow migration

Presenter: Philipp von Weitershausen
Track: Zope
Day: Tuesday, June 28
Time: 11:00 - 11:30
Location: VB

Zope X3.0 was released more than half a year ago, Zope 3.1 is ante portas and the number of Zope3-based projects is increasing. But what to do with all that Zope 2/CMF-based code that's successfully running? Throwing away is not an option, but neither is falling behind.

So it might be a good time to look for a migration strategy. One that exists for nearly a year now has materialized itself in the Five product, written by Martijn Faassen, Sidnei da Silva, the speaker and many others. This drop-in product for Zope 2 brings a lot of Zope 3 technology to good old Zope 2, allowing an explicit, bit-by-bit code migration.

This talk will first introduce the Five product and explain its installation and usage, then cover different migration scenarios using hands-on code examples. Target audience are Zope 2/CMF developers with little or no Zope 3 knowledge.

Visualization of three- and four-dimensional microscopy images with BiolumeXD

Presenter: Kalle Pahajoki
Track: Science
Day: Tuesday, June 28
Time: 11:00 - 11:30
Location: VF

Pasi Kankaanpää(1,2), Kalle Pahajoki(1), Varpu Marjomäki(1), Jyrki Heino(2) and Daniel White(1) (1) Department of Biological and Environmental Science, University of Jyväskylä, Finland (2) MediCity Research Laboratory and the Department of Biochemistry and Food Chemistry, University of Turku, Finland

Imaging of cells is critical for biological and medical research. It is essential to find out how proteins act in living cells, which is possible by fluorescence imaging in three or four dimensions. Consequently, confocal laser scanning fluorescence microscopy has emerged as a powerful method.

Processing confocal microscopy data is technically challenging and even institutions that possess equipment for confocal imaging often lack proper software for processing the data. Available commercial software is expensive, and often limited in features and flexibility. Algorithms for data processing and 3D rendering employed in commercial software are often treated as trade secrets, making it impossible for the user to validate the results.



Our project is developing free and open source software, BioImageXD, for the visualization and analysis of 3D and 4D microscopy images. BioImageXD will be a free tool for confocal microscopy and related methods that can rival or outperform commercial applications.

Most of BioImageXD is implemented in Python, with speed requiring processes like rendering and data processing implemented as C++ modules for The Visualization Toolkit. VTK is an open source toolkit for 3D graphics, image processing and visualization that integrates well with Python. In this presentation I will outline the structure of BioImageXD and describe how it uses the visualization pipeline model employed by VTK for the processing and rendering of very large microscopy image datasets. The functions of the software will be presented through practical examples and compared with the state of the art in commercial offerings.

GUI programming in Python using Gtk+

Presenter: Johan Dahlin
Track: Tutorials
Day: Tuesday, June 28
Time: 11:00 - 12:30
Location: VR

An introduction to Gtk+/PyGTK will be given in a presentation. Features and advantages will be discussed. The audience will not need to know anything about development of graphical applications, neither with PyGTK or any other toolkits. Basic concepts will be explained and demonstrated.

Several applications will be written from scratch to show the general workflow of a PyGTK application:
- Hello World - Web browser using Mozilla

The UI designer Gazpacho will be shown and a small couple of demos will be shown: - Hello world using Gazpacho - Web browser - Full featured text editor

If time is given, additional GNOME technologies such as virtual filesystem, file manager, CORBA will be demonstrated.

Portability, easy of use, documentation, licensing will also be noted so the audience have a good understand of the platform.

Open Space

Track: Misfits
Day: Tuesday, June 28
Time: 11:00 - 12:30
Location: VV

If you want to have a place to have a BOF (Birds of a Feather session) or have some other informal session, this is where you can do it. The Cafeteria is also available for informal sessions, but a bit noisier.

Complex security with Zope 3 and an RDB

Presenter: Steve Alexander
Track: Python Frameworks
Day: Tuesday, June 28
Time: 11:00 - 12:00
Location: VA



A Laboratory Information Management System for High-throughput, High-content Screening

Presenter: Aaron Bingham

Track: Science

Day: Tuesday, June 28

Time: 11:30 - 12:00

Location: VF

RNA interference (RNAi) is a naturally occurring process in which messenger RNA (mRNA) is degraded by a cellular mechanism regulated by double-stranded RNA. Short synthetic double-stranded RNA, called siRNA, may be designed and produced to suppress the expression of specific genes. The inhibition of individual genes in cultured cells results in changes in cellular phenotype, allowing conclusions about the cellular function of the inhibited gene.

Cenix BioScience GmbH specializes in large-scale RNAi experiments in which thousands of genes are inhibited one at a time. The effects on cell phenotypes are observed using microscopic imaging and computational image analysis.

Our laboratory information management system (LIMS) -- a cross-platform system developed using PostgreSQL, Pyro, and wxPython -- supports this work by:

- * Tracking siRNA sample stock and associated genomics data
- * Supporting preparation of experimental microtiter plates from stock with a variety of liquid handling robots
- * Controlling the image-acquisition process using automated microscopes and a robotic arm
- * Controlling the image-analysis process on a farm of CPU servers
- * Managing the image store and image-analysis results

Five in action

Presenter: Martijn Faassen

Track: Zope

Day: Tuesday, June 28

Time: 11:30 - 12:00

Location: VB

Five is the integration framework allowing the use of Zope 3 technology in Zope 2. Five has come a long way since its initial "coming out" at last year's EuroPython. Five 1.0 is now used in several production-level projects and is integrated in Zope 2.8.

This talk will give a brief introduction to Five, a status update about what has happened with Five over the last year, and show several pieces of software that use Five.

Modular toolkit for Data Processing (MDP)

Presenters: Pietro Berkes, Tiziano Zito

Track: Science

Day: Tuesday, June 28

Time: 12:00 - 12:30

Location: VF



EuroPython Conference

We present the Modular toolkit for Data Processing (MDP), a Python library to implement data processing elements (nodes) and combine them into data processing flows. Each node corresponds to a learning algorithm or to a generic data processing unit. A node can be trained and executed. MDP is designed to make the implementation of new algorithms easy and intuitive. Sequences of nodes (flows) can be defined, allowing for automatic training and execution. The nodes are designed to process arbitrarily long data sets, since training can be done online or in batch-mode.

MDP is flawlessly integrated in SciPy (the scientific Python library), giving to the scientific programmer the full power of well-known C and FORTRAN data processing libraries. MDP helps the programmer to exploit Python object oriented design with C and FORTRAN efficiency.

Already implemented nodes include Principal Component Analysis (PCA), Independent Component Analysis (ICA), Slow Feature Analysis (SFA), and Growing Neural Gas Network.

MDP has been written for research in neuroscience, but it has been designed to be helpful in any context where trainable data processing algorithms are used. Its simplicity on the user side together with the reusability of the implemented nodes could make it also a valid educational tool.

<http://mdp-toolkit.sourceforge.net>

Twisted news

Presenter: Tommi Virtanen
Track: Python Frameworks
Day: Tuesday, June 28
Time: 12:00 - 12:30
Location: VA

The Twisted codebase is changing quite fast. Keeping up with the changes can be troublesome, especially if you are still in climbing the learning curve. You just grokked Deferreds, and thought twisted.flow was something great, and then you get hit by a deferredGenerator.

I've seen the time before Deferreds, .tac or twisted.application; I've written code with twisted.web, woven and nevow. Hey, I even gave the name to woven. Let me bring you up to date with the latest and greatest of Twisted.

Actual contents of the talk will depend on last-minute minion activity on #twisted, but these things will be included:

- Twisted 2.0, subproject split - web2 - twisted vs. WSGI - conch and why you should care about it - filesystem API to unify FTP, SFTP, WebDAV, HTTP and whatnot - liveevil^H^H^Hpage^WAJAX^Winteractive web thingies - formless is dead, long live formless - atop, Quotient, q2q, shtoom, XapWrap - Idaptor, scalemail, wubwubwub, TouRSSt, watchwatchport, tacit - BuildBot, CIA - PyKon, Weever, Reptile, Backwards - yarn, Hep - Vellum, TwistedSNMP - squishes

SchoolBell -- calendaring in Zope 3

Presenter: Albertas Agejevas
Track: Zope
Day: Tuesday, June 28
Time: 12:00 - 12:30
Location: VB

SchoolBell is a general purpose calendaring component of the SchoolTool school administration software framework. SchoolTool is a project of The Shuttleworth Foundation.



EuroPython Conference

For Zope 3 programmers SchoolBell offers a calendaring library supporting a reasonable subset of the iCalendar specification (RFC 2445), a relationship library for many-to-many relationships.

For the end users SchoolBell is a stand-alone calendaring server for small groups and organizations based on Zope 3. It allows the user to manage her calendar through the web, and to see it overlaid with calendars of other persons and groups. Also, SchoolBell supports iCalendar clients, such as Mozilla Calendar, Apple iCal, and KOrganizer.

The SchoolBell server has a web services interface based on the REST architectural style for automation and integration with other systems.

C-Lab: a portal for Computational Chemistry at AstraZeneca

Presenters: Emily Bache, Anders Egneus, Farzad Faramarzi

Track: Science

Day: Tuesday, June 28

Time: 14:00 - 14:30

Location: VF

This is a success story about Scientists and Software Developers working closely together to achieve more than either group could have done alone. We want to tell you about the factors key to our success: · Close daily communication between scientists and developers · Automated Testing with PyUnit and TextTest. · Planning with index cards · Excellent architecture · Python

Yes, Python has been important for us - enabling us to develop code quickly, deploy easily, and retain flexibility to change our minds. The other things on the list are also important you might find them useful too.

We will also tell you a bit about the architecture we have chosen: · TAL html for web user interface · Webservice interface used by other applications · Sun Grid Engine for job management on a Linux cluster · The PyDrone model and descriptor engine, built by Andrew Dalke. · Calculation Services managed by PyDirector

We hope this success story will encourage you.

Moellus - a soft layer on Durus

Presenter: Mario Ruggier

Track: Python Frameworks

Day: Tuesday, June 28

Time: 14:00 - 14:30

Location: VA

The Moellus framework allows for easy development of object-oriented database applications, while remaining relational-friendly should the need arise to implement and deploy the application with a relational database. Moellus is a layer on top of the Durus Object Persistence System for Python, and notably makes heavy use of the ComputedAttribute class, to handle dynamic derivation of information based on attribute values across objects, e.g. for the automatic calculation of inverse relationships. Moellus organizes objects in dedicated containers, thus conceptually not straying far from a relational model. However, typical relational limitations (such as data types, clumsy object references) do not apply. Moellus comes with a very simple web interface to browse objects in a moellus database instance.

Moellus is being developed by a small python-based company, for the specific task of flexibly prototyping an extensive object model. We feel it may be of interest to the python community.



EuroPython Conference

Please note that Moellus has not yet been publicly released, but we plan to do so sometime in May 2005. Also, should you be interested that I come to Goteborg to present this, we would appreciate any help you may provide as we are really a small struggling auto-financed company (scoopics.com) and currently doing only development, with our first product planned for end of 2005.

Writing XMMS2 clients in Python

Presenter: Tobias Rundström
Track: Tutorials
Day: Tuesday, June 28
Time: 14:00 - 15:00
Location: VV

XMMS2 is the new generation media player. The core of XMMS2 is splitted in two parts a client and server. Server part includes nicely seperated plugin structure, medialib, powerful effect API and futureproof (support for 5.1 channels and resampling). The client library has native bindings to Python and a lot of the clients are written in Python, including the official medialib browser. The whole communication is asynchronous and requires some special considerations when writing a good responsive client. This tutorial will help you understand the PythonAPI and also learn you a lot of asynchronous communication trixs. Speakers include the main programers of XMMS2.

PyPy as a Compiler

Presenters: Carl Friedrich Bolz, Holger Krekel, Armin Rigo, Christian Tismer
Track: Python Language
Day: Tuesday, June 28
Time: 14:00 - 15:00
Location: VR

The PyPy project (<http://codespeak.net/pypy>) aims at producing a simple runtime system for the Python language, expressing the basic abstractions within the Python language itself. Simplicity and Flexibility are the foremost goals.

To reach our goal, we are currently working on a Python source analysis and compilation toolchain -- based on PyPy as well -- and a minimal low-level core that doesn't need CPython any more. The presentation will focus on these tools, which are able to perform automatic static type inference of "static enough" Python programs and then compile them to various lower-level languages (currently, we have experimental back-ends for C, LLVM, Python, Lisp, Pyrex and Java). This analysis works on fully standard Python sources (bytecodes, actually), and the distinguishing feature of the techniques we use -- "abstract interpretation" -- is to allow type inference to be performed for any language for which we have an interpreter. Moreover, it doesn't require explicit type annotations added by the user.

To some extent, and although it is not our primary long-term goal, the tools we have developped could be used generally to check or improve the performance of any program. We will see in particular which "staticness" restrictions must be met for these tools to work, and look under the hood:

- * how we build control flow graphs;
- * how we perform type inference, and what "type inference" precisely means in this context;
- * type inference is not the end of the story: good code must be generated from the source and the inferred types.



EuroPython Conference

The presentation will focus on the latter point, explaining the two major code generators we have at the moment: the C and LLVM back-ends.

The techniques relate to Psyco's, the just-in-time specialized for CPython. We will compare them, and also contrast the "internal" notion of type used as an implementation detail (as in PyPy and Psyco) with the other typical approach of explicit user-specified types. This point of view puts our current efforts in the more general context of our long-term goals, which are about dynamic optimizations rather than static compilation.

Our approach is motivated by the desire of flexibility: it allows issues that normally require early design decisions to be postponed, and addressed later in possibly more than one way, without rewriting everything. Typical example: adding Stackless-style continuations in CPython required a whole-scale rewrite, but has only a "local" impact on PyPy. We will plead for a framework based on interpreters as a way to gain the most flexibility from programming languages -- and the best performances too, for very-high-level languages!

Plone4Artists

Presenter: Nate Aune

Track: Zope

Day: Tuesday, June 28

Time: 14:00 - 14:30

Location: VB

- Explain how the Plone4Artists team has leveraged Plone and additional products to build a turnkey solution for artist communities.
- Describe how the Plone4Artists skin was created which provides a whole new look for the Plone site.
- Demo the iPortfolio product which wraps many community building content types (music, photos, links, events, etc.) into a single easy-to-use tool.
- Discuss how CompositePack was utilized for drag-n-drop editing of page layout. Artists can 'design' their own home page and move elements around on the page.
- Demo the integration of iLife applications within the Plone framework (iPhoto -> ATImage, iTunes -> ATAudio, iCal -> Calendaring, AddressBook -> PloneContacts, Firefox -> ATBookmarks), drag-n-drop addition of content.
- Demonstrate how the Plone4Artists customization policy was created to rapidly deploy new websites
- Discuss how CMFMember was used to create a subscription-based website to generate revenue to sustain the operation of the website.

Z3 ECM Panel

Presenter: Paul Everitt

Track: Zope

Day: Tuesday, June 28

Time: 14:30 - 15:30

Location: VB

This panel discussion, moderated by Tres Seaver, will bring the leaders of the major CMS projects for Zope, together for a panel discussion on their partnership for the next generation of CMS and Zope.



Python as a Tool in Resource Optimization Systems

Presenter: Magnus Lyckå
Track: Business
Day: Tuesday, June 28
Time: 14:30 - 15:00
Location: VA

Carmen Systems develops, markets and implements resource optimization solutions for clients found primarily in the transportation industry. Typical clients are large airlines and railway companies. With our advanced algorithms and extensive experience, we help our clients make their operations more cost-effective, adaptable, reliable, and environmentally friendly.

Python has made it possible for us to add more power and flexibility to our main products at a reasonable cost. We will describe why and how we use Python in our software products, the benefits this gives us and our customers, and the problems we have encountered on the way.

More information about Carmen Systems is available at <http://www.carmensystems.com/>

Python and a few more animals - using Python in science & research at MPI-CBG

Presenter: Michael Volkmer
Track: Science
Day: Tuesday, June 28
Time: 14:30 - 15:00
Location: VF

BioSimGrid : An Environment to Archive and Analyse Biosimulation Data

Presenter: Stuart Murdock
Track: Science
Day: Tuesday, June 28
Time: 15:00 - 15:30
Location: VF

The comparison of biosimulations carried out by different software packages or in different labs is logistically difficult due to high volumes of data, proprietary data formats, variety of tools available to analyse the data, and lack of sharing of the data. However, comparative analysis of molecular dynamics simulations of biomolecules allows for the systematic, biologically relevant events to be disentangled from random observations. Comparative simulations therefore help us to understand experimental results and to provide guidance for future experimental work. The BioSimGrid project is developing a software environment to address these issues. With BioSimGrid we provide not only a distributed repository in which we store these large trajectories but also a common environment where they may be analysed with a range of system-supplied or user-supplied tools.

Analysis tools have been written to study the trajectories and it is expected that a user of BioSimGrid will either use the Analysis tools we provide which could be edited to their specific needs. The language of choice in which to write the analysis tools is Python as this is already used in the community and it provides a powerful object-orientated scripting language which is relatively easy for a user to learn.



We demonstrate the strength of the BioSimGrid environment by analysing different biosimulation trajectories using our standard toolkit. Case specific scripts used to analyse the archived data are simple to construct and transfer between trajectories. The logistical problems associated with the comparison of data obtained from different trajectories are minimal, the results are easily comparable, and may be displayed in a variety of different formats to allow their easiest possible interpretation.

Cyprix - A 'Swiss army knife' for operating SQL databases

Presenter: André Deparade

Track: Business

Day: Tuesday, June 28

Time: 15:00 - 15:30

Location: VA

Cyprès IT-Consulting operates a network of different databases for the strategic procurement departments of our customer. Database servers and clients are distributed worldwide.

In the last two years we have been developing 'Cyprix' for internal use. It is a suite of framework-based Python programs for automation of server-side operations on these databases.

Applications: Cyprix is used for different tasks: - controlling and scheduling of timed jobs - internal monitoring and controlling (jobs, time, space) - import and export to / from other data sources (flat file, CSV, XML) - archiving of data - mirroring / one-way replication of databases - (kind of) two-way replication - generating SQL scripts - configuration management, deployment, and setup - development, logging, testing

Advantages of Python: Cyprix has been realized with an extremely tight budget. It has evolved from a 70-line prototype to currently > 15 kLOC. This was only possible due to Python's strengths in productivity: - scalability - highly dynamic code - high-level language - 'run the prototype' - well formatted code - easily readable code

Business model: Time to market and incremental implementation is everything. This requires a flexible, powerful language with fast return on investment, i.e. the possibility to produce a usable program fast.

Development tools: Emacs, vi and Subversion. For some tasks, we have used Hyperbole, but it is not as good as one would like it to be. We would like to use a real class browser / editor like the one in Smalltalk and an-easy-to use tool for refactoring. This would speed up developing and increase flexibility. In order to realize a GUI, we need an easy-to-use database grid. Currently we have been testing Knoda.

Recoverable Exceptions In Python

Presenter: Michael Hudson

Track: Python Language

Day: Tuesday, June 28

Time: 15:00 - 15:30

Location: VR



Plone used for semantic web in the construction industry

Presenter: Reinout van Rees

Track: Science

Day: Tuesday, June 28

Time: 16:00 - 16:30

Location: VB

The construction industry consists of lots and lots of small companies, without big parties that prescribe certain solutions. Likewise, there are lots and lots of different computer programs around with very limited, mostly text-based, exchange.

Construction ICT research's main effort is aimed at big, complex standards that are in no way suited to the majority of small companies. (Semantic) web-based solutions offer a way out. The research being presented offers a simple, industry-oriented, web-based way of data-level cooperation in the construction industry (with very probable applicability in other sectors, btw.). Core to the approach is a simple ontology.

For the ontology, a management application was made in Plone, based on UML code generation. This way, multiple researchers could cooperate in creating the ontology - in contrast to many other research efforts. This, and other applications made for demonstration purposes, exported their data in xml or rdf formats for use in the other applications. Semantic webservices in action.

To get RDF working, the python rdflib had to be ported to zope, something that worked, but barely. A much better port is available to zope3, but some of the experiences might prove interesting.

The presentation is based on my almost-finished PhD research at Delft University of Technology. The almost-final thesis version can be found at <http://vanrees.org/proefschrift.pdf> with most of the python/zope/plone content in chapter 7.

The presentation might be interesting from several viewpoints:

- Web-based collaboration in a research setting.
- UML code generation, resulting in a great user interface (relative to the effort put into it).
- Use of ontologies as a basis for data exchange.
- Experiences with data openness.

Pythonic Mathematics

Presenters: Kirby Urner, Kirby Urner

Track: Education

Day: Tuesday, June 28

Time: 16:00 - 17:00

Location: VF

For several years I've been brainstorming with colleagues about ways to better integrate math learning and computer programming. Currently, the calculator has made deep inroads into pre-college curricula, but the computer, if present, is mostly used as a CD player for canned materials, such as educational games.

Programming is not commonly featured, perhaps in part because the number of languages has grown and teachers are unaware of their options. In particular, they don't know about Python and its potential to contribute to the math learning process. We who use Python are in a position to change that. One approach is to offer strong curriculum writing which shows by example what is possible.



In addition to writing on this theme quite a bit, I've had several opportunities to test these ideas in the field, including with my target age groups. My talk will draw on these experiences, even as I sketch my somewhat grandiose plans for the future.

Introduction to regression testing using TextTest

Presenters: Emily Bache, Geoff Bache

Track: Tutorials

Day: Tuesday, June 28

Time: 16:00 - 17:30

Location: VV

Lots of python scripts are disposable: short and only get run once. But then you find another similar problem and this disposable code evolves into a little script that gets run now and again. In fact, you find this little script so useful it turns into a little tool that might even get used by other people. Before you know it you're supporting a full blown application, and someone is pestering you about a bug they found in some feature you added weeks ago. Then when you fix it you discover that three other things break. Its sometime about then that you realize how useful it would be to have some automated regression tests to lean on while you clean up your code.

Automated regression tests are tests that confirm that your program still does what it did before, at the click of a button. They allow you to change the internal structure of your code (by refactoring) while ensuring that the behaviour of your program stays the same. There are many ways of achieving that, but one of the easiest ways we know of is to use TextTest.

About TextTest TextTest is a testing framework (written in python) that lets you define tests in terms of if I give my program this input, I expect this output. TextTest runs your program with the input you specified, and checks against expected output. If the output is different, the test fails, and TextTest warns you that you program is not behaving as you expected it to.

The main area where TextTest has a proven track record is when working on the command line with a batch program that does some kind of computation intensive work, before returning results in plain text. It has been used successfully at Carmen Systems for testing optimization products, and at AstraZeneca for testing tools for computational chemistry. TextTest, being a fairly large python application, is of course also used to test itself.

TextTest is available on sourceforge: <http://sourceforge.net/projects/texttest> and you can find out more information here: <http://www.texttest.org>

Intended audience You dont need to be a python expert or primarily a software developer to find TextTest useful. Well be pitching this so that for example an interested scientist or mathematician or manager or sysadmin who writes scripts on the side will be able to learn how to write and run useful regression tests. Python and testing experts are of course also welcome.

Instructions for Exercise If you plan to do the exercise in this tutorial you will need to have a laptop with you with texttest installed on it. You can find instructions for how to do that here: <http://texttest.carmen.se/TextTest/docs/install.html> The exercise materials will be available on texttest.org shortly.

Greenlets: coroutines aren't stranger than generators

Presenter: Armin Rigo

Track: Python Language

Day: Tuesday, June 28

Time: 16:00 - 16:30

Location: VR



EuroPython Conference

The "greenlet" package is a spin-off of Stackless (<http://www.stackless.com>), a version of CPython that supports micro-threads called "tasklets". Tasklets run pseudo-concurrently (typically in a single or a few OS-level threads) and are synchronized with data exchanges on "channels".

A "greenlet", on the other hand, is a still more primitive notion of micro-thread with no implicit scheduling; coroutines, in other words. This is useful when you want to control exactly when your code runs. You can build custom scheduled micro-threads on top of greenlet; however, it seems that greenlets are useful on their own as a way to make advanced control flow structures. For example, we can recreate generators; the difference with Python's own generators is that our generators can call nested functions and the nested functions can yield values too. (Additionally, you don't need a "yield" keyword. A "yield" function would do just as well). One could say that greenlets enable a kind of "generator-like" coding style that doesn't necessarily impose an asymmetric consumer-producer model.

Greenlets are provided as a C extension module for the regular unmodified interpreter, as part of the 'py' lib (<http://codespeak.net/py>).

The talk will be focused on a few examples of usage of greenlets, hint at what kind of higher-level mechanisms they could be used to build, and also point at some features that can only be provided by the full Stackless Python.

Where metaclasses surpass decorators

Presenter: Theo F. de Ridder
Track: Python Language
Day: Tuesday, June 28
Time: 16:30 - 17:00
Location: VR

Decorators offer compact specifications for signatures of functions and methods without any need to change existing code. Being itself a function with a function to function signature a decorator is very flexible and powerful. However decorators become clumsy when any reflection plays a role. Then it hurts that decorators are not operators nor types, and that one needs source-code inspection to check if a function is decorated. Metaclasses become superior to decorators in cases where pre- and post-evaluations do more than simple transformations and type-checking or contain declarative elements like getters and setters in properties. A reasonable mix of both mechanisms can be obtained by making decorators instances of metamodelled classes that adapt decorations to a wide range of context dependencies. But real pythonic declarative and reflective power will be enabled when decorators are types and overloadable meta-operators. A small implementation in Python 2.4.1 will demonstrate surprising benefits, like lazy evaluation, of enhancing the syntax and semantics of decorators just a little.

Storage Resource Broker (SRB), Large scientific data and Python

Presenter: Steven Johnston
Track: Science
Day: Tuesday, June 28
Time: 16:30 - 17:00
Location: VB

Computer simulations play a vital role in biochemical research. By simulating the interactions of all atoms within a molecule or protein, the biochemical properties of the structure can be revealed. One important application of such Molecular Dynamics and Monte-Carlo simulations is predictive modelling in drug discovery, where the motion of proteins are important. These simulations are computationally demanding and they produce huge amount of data which is analysed by a variety of methods in order to obtain biochemical properties. Generally, these data are stored at the laboratory



where they have been computed in a proprietary format which is unique to the simulation code that has been used. This constrains the sharing of data and results within the biochemistry community therefore the data can generally not be compared easily with post processing tools due to the varying data formats. This talk covers how the BioSimGrid project utilises python to produce a framework which stores the data as well as allows users to submit processing scripts to process the data. The data are structured into two key areas, the trajectory data and the metadata. As the metadata is small and relational, it is stored in a database, currently Oracle 10g. This database is replicated across our 6 sites and accessed using the Python interfaces. The trajectory data is 5-20GB and not very suited to a relational database, so for storage and processing requirements we have produced a framework which stores and retrieves the data into a Storage Resource Broker (SRB) using python.

SRB enables us to deposit data at any site/location and have it available to all sites thus eliminating the need to transport or record the location of the data. There are currently many interfaces to SRB for other languages as well as a Windows based application and a very basic python interface. This talk will also look into the existing API as well as ongoing work, it will cover the advantages and disadvantages of SRB, providing a practical example showing how it can be used to store extremely large volumes of data.

RhoZeta and Phi² - master the challenge of forms with the power of Python

Presenter: Harald Armin Massa

Track: Business

Day: Tuesday, June 28

Time: 16:45 - 17:30

Location: VA

"forms" within office-applications like Excel and Word are the successors of bureaucratic paperforms within many companies. Problems arise in the long run and are dangerous, for example: different versions of templates AND filled out forms float around; reporting ranges from "very expensive" to "extremely expensive".

Implementing applications within corporate settings often involves much training and many seminars; on the other hand applications like ebay, amazon and google / gmail are used almost without training.

So we combined those approaches: database driven generation of webforms, which store data in central databases; replication techniques to use them on the laptop. A "core data entry application" with the ease of use of an WebBased application.

Solving puzzles with Python

Presenter: Gustavo Niemeyer

Track: Python Language

Day: Tuesday, June 28

Time: 17:00 - 17:30

Location: VR

Many of the classical logic puzzles may be defined in terms of constraints on the domains of given variables. These kinds of problems are usually interesting and challenging, and are defined in the theory as Constraint Solving Problems (CSPs).

On this talk, some funny and well known CSP problems will be presented, and a new Python library for gracefully solving these problems will be exposed.



A Python Experiment in a Brazilian University

Presenter: Rodrigo Senra
Track: Education
Day: Tuesday, June 28
Time: 17:00 - 17:30
Location: VF

After contextualizing the status of CS courses in Brazilian Universities, I'll reports my experience while teaching three different Computer Science subjects in PUC University in Brazil from 2002 to 2004.

The way Python was used by both -- students and the teacher -- is explained to the audience, as well as the immediate results of this pedagogical experiment.

To complement the presentation, I'll expose some language-orthogonal strategies like: The Big Brother experiment, coding mixed with art, teacher as a cheer leader, and other non-orthodox approaches that caused a positive response from the students.

A Simple Python Framework for Introducing Component Principles

Presenter: Raphael Marvie
Track: Refereed Papers
Day: Tuesday, June 28
Time: 17:00 - 17:30
Location: VB

The framework is available (GPL) at:

<http://www.lifl.fr/~marvie/Software.html#picolo>

iKaaro Case Study: Traveluni

Presenters: Luis Belmar-Letelier, Norman Khine
Track: Zope
Day: Wednesday, June 29
Time: 09:00 - 09:30
Location: VF

Smart Package Manager

Presenter: Gustavo Niemeyer
Track: Misfits
Day: Wednesday, June 29
Time: 09:00 - 09:30
Location: VR

The Smart Package Manager (<http://smartpm.org>) project has the ambitious objective of creating smart and portable algorithms for solving adequately the problem of managing software upgrading and installation. It's currently being used in most major distributions (Conectiva Linux, Debian, Mandrake Linux, Fedora, CentOS, and more), presenting great compatibility with different package managing systems (RPM, DPKG, Slackware).

The following list presents some of the project features:



EuroPython Conference

* Smart transactions * Multiple interfaces (gtk, shell, command line, command line+gtk) * Support for several channel formats * Priority handling * Autobalancing mirror system * Parallel downloading mechanism * Flexible removable media support

These features and a good part of the project success were achieved thanks to the Python language, which was used to build the project.

This talk will present the software, it's features, advantages, important aspects of the Python language that have helped the project, proves that the language speed is not an issue even for complex real world projects, and more.

Stupidity and laser cat toys: Indexing the US Patent Database with Xapian and Twisted

Presenter: Michael Salib

Track: Business

Day: Wednesday, June 29

Time: 09:00 - 09:30

Location: VB

I recently downloaded the full text of the US Patent Database. It seemed like a good idea at the time. This talk will show how I built a cool web search engine for it using Xapian, an open source full text indexing library and Twisted, the ultimate network coffee maker. While these tools are very powerful, they all have their own pitfalls. I'll show how to put them all together and get something useful as a result.

Fun matters

Presenter: Benno Luthiger

Track: Social skills

Day: Wednesday, June 29

Time: 09:00 - 09:30

Location: VA

The aims of the FASD study are twofold: First, the study aims to make a quantitative estimation of the importance of fun in order to explain the open source developers' commitment. Second, it aims to verify the hypothesis that open source developers have more fun when programming than developers of commercial software. If this hypothesis can be confirmed, the FASD study aims to identify those elements which are responsible for the fact that open source developers have more fun with their work than commercial developers. The analysis of the data gathered proves that fun plays an important role when software developers decide to get engaged in an open source project. Moreover, the comparison of the answers gives evidence for the hypothesis that programming in an open source project is significantly more fun compared to the same activity under commercial conditions. The reasons for this fact are that open source projects are able to attract software developers with a credible project vision and that they can offer them an optimal challenge.

Localizer 1.1, and Case Studies

Presenters: Juan David Ibáñez Palomar, Mikel Larreategi

Track: Zope

Day: Wednesday, June 29

Time: 09:30 - 10:00

Location: VF



The Python revolution in the publishing industry

Presenter: Stefano Masini

Track: Misfits

Day: Wednesday, June 29

Time: 09:30 - 10:00

Location: VR

Publishing is a task that traditionally is done by graphic designers. Their usual way of doing things is by hand, meaning that the tools currently available to do the job, like Adobe InDesign, FrameMaker or Quark Xpress, simply mimic a sheet of paper and provide a set of tools that perform like very smart pencils and brushes.

Publishing a lot of content or publishing very often becomes a nightmare because it means a lot of manual work, with a huge potential for errors and typos, and very poor integration inside the company workflow.

Pragma 2000 is pioneering the field of publishing automation. We provide higher level tools, that are meant to be used by the end user, not the graphic designer. Tools that work with the user data, integrate with his database, and don't require manual cut&paste.

The traditional DeskTop Publishing software (like Adobe InDesign) is used as a simple backend that is entirely driven by a program. The manual publishing step is skipped entirely in the day-by-day workflow.

We provide workflow management tools for tracking the evolution of the publication, simplify the communication and exchange data.

Of course... everything is done in Python! Data representation is done in XML. Adobe InDesign win32 scripting through COM. Network management and distributed systems implemented in Twisted. Graphical user interfaces with wxWidgets. Database management with Dabo.

Python as an innovative glue for traditional tools: a true revolution in the publishing industry!

PyPy and sprint-driven development

Presenter: Beatrice During

Track: Social skills

Day: Wednesday, June 29

Time: 09:30 - 10:00

Location: VA

The Pypy project has successfully used sprints as a driving factor in the development work. We will talk about how and why this works, and give some useful hints about what to do and what to avoid.

Python in Pension Funds

Presenter: mathias foehr

Track: Business

Day: Wednesday, June 29

Time: 09:30 - 10:30

Location: VB

Using Python as a back-end tool to enable actuaries to define or adapt their financial calculations in a user-friendly way while keeping the data in a secure and integrated environment. Strategic objectives:



EuroPython Conference

Adaptability, Reliability and Perenniality (> 50 years). Technical aspects: Objects in financial calculations, Integration with Java, Other aspects discussed: Evangelization and Marketing

Having given a second thought to this presentation, I think I could easily speak for an hour. It would also be possible to split this talk in two - hopefully consecutive - sessions of half an hour each, one about management and related aspects, a second one being more technical.

Pimp my site!

Presenter: arjen Voogt
Track: Zope
Day: Wednesday, June 29
Time: 10:00 - 10:30
Location: VF

This talk will guide you through changing the looks of your out-of-the-box Plone site. The 5 steps are best practices derived from Plone development experience. During the talk we will interactively build a good looking Plone site. The talk will be a mixture of slides and real code. The talk is primarily aimed at Plone newbies or anyone interested in setting up a basic Plone site.

ItsATree - creating a multimedia editor

Presenter: Volker Gietz
Track: Misfits
Day: Wednesday, June 29
Time: 10:00 - 10:30
Location: VR

This is an overview of a long-running project of mine which aims at providing what could be called a "generic timeline editor". The long-term goal is to enable the user to mix and arrange many different types of media while keeping the number of UI metaphors in the system at a minimum. Python is in charge of providing the GUI and data management, also setting up the proper C++ object graphs for playback. Python's dynamic nature has actually inspired the system design quite a bit since its introduction -- for instance, all necessary conversions between inhomogeneous plug-ins are handled with compilation techniques such as type inference. A track can handle any mixture of media objects, proper conversion is attempted for each of these objects depending on the type of the filter following that track.

It is quite interesting to see that the application feels more and more like a functional programming environment with increasing development time. Lists are central to the data model and internally, setting up rendering is nothing more than building subsets of transformed list elements. Since this is an experimental project, I have decided that even the user interface should reflect the minimalistic nature of the implementation -- in fact, at the moment there is only a list, a tracks and a decimal control.

Technically I am using a blend of home-grown SWIG-wrapped C++ libraries along with pygame for the UI. Given the current pace of development it should be possible to give a quick demonstration of the application itself -- with a heavy slant towards music production though :-)



Choosing Good Names

Presenter: Stefan H. Holek
Track: Social skills
Day: Wednesday, June 29
Time: 10:00 - 10:30
Location: VA

As software programmers, we choose names for modules, classes, methods, constants, and variables as part of our daily routines. Quite often we do not expend much thought on these choices, picking whatever seems to fit as we go along, our minds being focused on the programming problem to solve, rather than the names we introduce in the process.

Now, why would the seemingly mundane task of selecting identifiers be the subject of a talk?

By naming a thing, we put it into existence. A name appearing in a piece of code does not simply signify an address in the computer's main memory, but ultimately defines our ability to talk and think about the program and the problem it purports to solve. Good names make it easy to comprehend the purpose of a software program and the characteristics of the problem-space. Bad names on the other hand obscure the programmer's intentions, making his creation hard to understand and use.

Badly chosen names fail to convey meaning; and without the right words we are - in the end - unable to communicate.

This talk attempts to raise awareness of the importance of naming and to give hints on how to tell the good names from the bad ones. After all: Readability counts!

Python and Money: Discussion

Presenters: Facundo Batista, John Pinner
Track: Business
Day: Wednesday, June 29
Time: 11:00 - 11:30
Location: VF

There are two major problems when manipulating currency in business applications:

1. Loss of accuracy and rounding errors when using floating point numbers.
2. handling currency conversions and exchange rates.

The discussion will centre around these and other issues with the aim of arriving at a clear set of goals for the sprint.

Web Application Testing with Selenium

Presenters: Maik Roeder, Maik Roeder
Track: Zope
Day: Wednesday, June 29
Time: 11:00 - 11:30
Location: VB

Selenium is a test tool for web applications. Its automation engine is embedded in the browser via JavaScript and Iframes. Selenium can be of use in two main areas, browser compatibility testing and system functional testing. In browser compatibility testing, you test your application to see if it works



correctly on different browsers and operating systems. In system functional testing, the focus of this talk, you create regression tests to verify application functionality and user acceptance.

py.test - making testing fun

Presenter: holger krekel
Track: Social skills
Day: Wednesday, June 29
Time: 11:00 - 11:30
Location: VA

py.test ('std.utest' in a former brief incarnation) has extensive documentation and is actively used and improved in the PyPy project, among others. We talk about features such as the new tkinter-frontend as well as other upcoming test features such as doctest support.

Testing is a perfect match to control and make full use of Python's flexibility. Unfortunately, the testing mechanisms of most python projects are more or less entangled hacks around unittest.py. We think that we can and should do better than this.

Python Logo and Website Redesign

Presenter: Tim Parkin
Track: Misfits
Day: Wednesday, June 29
Time: 11:30 - 12:00
Location: VR

A talk about the redesign of the Python logo and website and the creation of a templating system to allow the simple management of site data. The talk will cover

- the process of reaching the current design and various mis-steps on the way.
- the requirements for the system that would maintain the python site - using nevow, yaml and rest to create text based templating system
- how the current system could be deployed
- future plans for distributed management of the Python site's content
- how to get involved

Money module

Presenter: Facundo Batista
Track: Business
Day: Wednesday, June 29
Time: 11:30 - 12:00
Location: VF

The Money module will have a Currency data type, some useful functions to work with money (Time Value of Money, etc.) and some structures with useful information.

This is being developed right now, and the functional core is already designed, but the idea is to reach consensus through the community for the use cases and behaviour that we'll need to achieve.

In the talk I'll show in which state we are, and the decisions that are to be made. And I'll search for mind power for the sprint and further development of the module.



Using tests for team motivation

Presenters: Johan Andersson, Geoff Bache

Track: Social skills

Day: Wednesday, June 29

Time: 11:30 - 12:00

Location: VA

Last year we gave a talk that focused on the technological aspects of the acceptance testing framework, called TextTest, that we use on a daily basis in our development work.

This year we hope to focus more on the methodology aspect of how it affects our work and how it fits as one important piece in our agile development methodology.

We hope to show that a task that has long been regarded as a tedious and boring part of development work, a task that as much as possible has been off-loaded to a separate test department, can be turned around and actually even be used and perceived as a team motivator.

A rough outline of the talk is:

- Introduction, similar to the short description above.
- A short description of the team and the project start in 2001
- Methodology choices, strategies and initial decisions
- The battle of unit tests vs. acceptance tests
- The insight that people's time is more expensive than CPU's, and the consequence for distributed testing.
- A quick look at the technological support tools that can make automated and distributed acceptance tests feasible. (Python with Texttest, LSF and SGE as batch tools)
- How does automated and distributed acceptance tests fit as one piece in an agile development methodology.
- Some thoughts on how to go stepwise from ad-hoc development to something similar to how we work today.

WYSIWYG interface design with CPSSkins and CPSPortlets

Presenters: Julien Anguenot, Jean-Marc Orliaguet

Track: Zope

Day: Wednesday, June 29

Time: 11:30 - 12:00

Location: VB

CPSSkins is a WYSIWYG theme editor for CMF/CPS/Plone. CPSPortlets is a portlet manager for CPS3. The two products have been closely integrated together to give users the ability to create interfaces without having to figure out the intrinsics of Zope, CPS or CMF.

- A description of the overall architecture will be presented
- A live demo explaining how to create fully functional website in matter of minutes will be given.
- Programmers will be shown how to create custom portlets.
- A sneak preview of the upcoming Zope3 version will be presented.



The Python Virus - Using Python for ten years in ePublishing business

Presenter: Andreas Jung
Track: Business
Day: Wednesday, June 29
Time: 12:00 - 12:30
Location: VF

Lately I read in an online article about Python coming to enterprise. I thought big enterprises? Coming? Over the last ten years of my professional life we have been using Python in lots of successful projects also in larger companies (up to 1000 employees). So Python is already in the enterprises since years!!

Python has been widely used in lots of ePublishing projects e.g. in some large-scale projects for the European Union by the end of the 90s. In some projects we managed a multi-lingual document environment with eleven languages and a million documents through a Python-based system working with SGML documents.

Nowadays the trend in ePublishing shifts away from SGML to XML. XML based solutions are state of the art.

This talk will tell some success stories how Python and Python-based solutions have been used for over ten years to support companies and publishers in producing content - both printed and electronically. The time has shown that Python was always flexible, scalable and development-friendly to build, maintain and extend systems in the field of ePublishing.

The Personal Internet Endpoint: Using Python and Twisted to write Reliable Peer-To-Peer Programs

Presenter: Michael Salib
Track: Misfits
Day: Wednesday, June 29
Time: 12:00 - 12:30
Location: VR

The internet is designed to facilitate symmetric connectivity between peers. With the advent of personal internet access, however, several features of today's internet, including network address translation, corporate firewalls, and dynamically assigned IP addresses, those connections are now far from symmetric, and in many cases not even possible.

This paper will explore several solutions to this problem which use Python's inherent flexibility and Twisted's networking infrastructure to solve it in a general manner. Several code examples will be presented where, using Twisted's separation of protocol and transport layer, existing applications are made to work by modifying only one or two lines of code.

It will cover each of the obstacles to connectivity in order of difficulty: dynamic IPs, NAT, and finally firewalls. It will also discuss the security, user interface, and application ramifications of being able to provide connectable services on each user's PC and/or other IP connected device. Covered networking techniques will include:

- * Detecting the ability to open a port on a static, internet-visible IP, by providing a callback server.
- * Using stateful UDP filters on most modern firewalls to provide a temporary, virtual connection.



EuroPython Conference

* Load-balancing so that users with truly broken connections (such as those made through a corporate firewall using the "CONNECT" HTTP command through a proxy) can piggyback on users with more open connections without causing them an undue amount of load.

Existing internet standards will be covered as well:

* SIP * STUN * UPnP

Finally, the paper will propose a standard set of techniques and an accompanying protocol for gradually and gracefully falling back to progressively less efficient techniques so that transport-level concerns don't interfere with an application's functionality until it is actually impossible to establish any kind of connection.

The presentation will cover a several of the techniques explored in the paper and will focus on real-world implementation and deployment of these techniques. It will present code examples, and include several demonstrations, hopefully using the wireless internet access provided by the conference, but with a small virtual network if that one is unavailable for some reason.

Method panel - you don't get paid to have fun!

Presenters: Johan Andersson, Beatrice Doring, Stefan H. Holek, holger krekel, Benno Luthiger

Track: Social skills

Day: Wednesday, June 29

Time: 12:00 - 12:30

Location: VA

-

Zope 3 Common Criteria Certification

Presenter: Aroldo Souza-Leite

Track: Zope

Day: Wednesday, June 29

Time: 12:00 - 12:30

Location: VB

Zope and Zope Foundation

Presenter: Rob Page

Track: Zope Lightning Talks

Day: Wednesday, June 29

Time: 14:00 - 14:45

Location: VB

Zope Corporation has begun the process of creating a Zope Foundation. The Zope Foundation will provide a mechanism for managing the various Zope projects (e.g., Zope 2, the Zope CMF, Zope 3, and other Zope community projects) in a vendor neutral way.



EuroPython Conference

Lightning talks

Track: Lightning Talks
Day: Wednesday, June 29
Time: 14:00 - 15:30
Location: VR

You can register your lightning talks on-line at the Europython website, <http://www.europython.org>

Zope Lightning talks 1

Track: Zope Lightning Talks
Day: Wednesday, June 29
Time: 14:45 - 15:30
Location: VB

Lightning Talks are short (5 minutes) focussed talks about any Python- or Zope-related subject you fancy! Some Lightning Talks will be arranged at the conference, but you can sign up for one in advance too.

Zope Lightning talks 2

Track: Zope Lightning Talks
Day: Wednesday, June 29
Time: 16:00 - 17:30
Location: VB

Lightning Talks are short (5 minutes) focussed talks about any Python- or Zope-related subject you fancy! Some Lightning Talks will be arranged at the conference, but you can sign up for one in advance too.

Lightning talks

Track: Lightning Talks
Day: Wednesday, June 29
Time: 16:00 - 17:30
Location: VR

You can register your lightning talks on-line at the Europython website, <http://www.europython.org>

Presenter Listings

Albertas Agejevas

Albertas is a Python hacker at Programmers of Vilnius. He works on a team that was one of the earliest adopters of Zope 3 for production use, and contributes to the Zope 3 development. Albertas has an MSc in Computer Science and teaches Extreme Programming and Unix Systems Programming courses at the Vilnius University.

Andrea Agili

Born on 07/24/1979, I'm completing studies in Computer Science Engineering at the University of Florence. I know several programming languages: Python, Java, C++... I made a brief presentation of Python language and Plone at a programming conference at the university of Florence in 03/15/2005. I've developed several Plone based sites for both private and public sector. Since October 2004 I work for DrWolf srl. A company wich delivers open source content and knowledge management systems.

Steve Alexander

Zope 3 developer, currently working for Canonical (who make the Ubuntu linux distro) developing systems using Python and Zope 3, along with a great team of python programmers.

Johan Andersson

Johan has been working with Agile methodologies for five years. He has experience from multiple fields in software development and is currently working with optimization software at Carmen Systems. Johan lives in Gothenburg with his wife and two Pumi dogs.

Julien Anguenot

Core Zope and CPS developer. R&D engineer at Nuxeo.

Martin Aspeli

Martin Aspeli is presently a student at the London School of Economics, completing his MSc in Analysis, Design and Management of Information Systems. His dissertation will detail the community processes surrounding the Plone project.

Martin got involved with Plone during another academic project, creating a knowledge management system for small businesses as his final year project at the University of Warwick, where he studied Computer and Business Studies. Climbing the Plone learning curve, he began to contribute, first by writing documentation, then by taking some responsibility for managing the documentation effort, and working on the PloneHelpCenter and PloneSoftwareCenter community support modules. More recently, he has become a core Plone developer, helping to push the upcoming 2.1 release out the door.

Through this involvement, Martin made contacts with several small businesses and non-profits, and began to offer his services as a Plone consultant, both on his own and in conjunction with PloneSolutions in Norway. He has built large-scale systems for the NanoBusiness Alliance in the USA, consulted for the National Health Service in the UK, and created various web sites and content management solutions for other customers.

Nate Aune

With a formal education in computer science, Nate has always been intrigued with utilizing cutting edge technology to find solutions to everyday problems. He has worked for many years in various IT companies as a web developer, and now assists small businesses through his company Jazkarta Consulting.

Growing out of his collaboration with other musicians and technologists, Nate decided to fuse his IT skills together with his love of music, and built Bostonjazz.net, an online community website to network jazz musicians in the Boston area and provide web-based tools for promoting their music. This is one example of a website built on top of the Plone4Artists framework. Nate combines the adaptability of an improvising musician with the analytical mind of a scientist. He is a technician, entrepreneur and innovator with an incessant need to tinker, create and evolve.

Emily Bache

Emily currently works for AstraZeneca in Mölndal, Sweden, writing python applications to help scientists with their research. In 2004 she was on maternity leave for most of the year, but still managed an appearance at EuroPython to give a lightning talk. Emily has presented posters about automated testing and writing software in collaboration with scientists respectively at two european XP conferences. She is tutorials co-chair for the XP2005 conference and is married to Geoff Bache.

Geoff Bache

Geoff has pioneered automated regression testing techniques at Carmen Systems and is the creator and driving force behind TextTest. He has published papers about automated testing at several of the european XP conferences. Geoff is tutorials co-chair for XP2005. He graduated from Cambridge University in 1997 and is married to Emily Bache.

Facundo Batista

Facundo Batista

Male, 29 years old.

Python developer. PSF member. Foundational member of PyAr - Python Argentina.

Actually working in a mobile services company, Telefonica Moviles from Argentina.

Luis Belmar-Letelier

Luis Belmar-Letelier has a PhD in Physics from the University of Franche-Comté. He moved to the Computer field and today he is at the head of itaapy, a company where Python and Zope are at the core of its business. With five years of experience with Python and Zope, he is an active member of the french-speaking open source community.

Pietro Berkes

- since Sept. 2001: PhD student at the Neural Computation Research Group of the Institute for Theoretical Biology (ITB), Berlin, with Dr. Laurenz Wiskott. Research subject: a computational model of self-organization of receptive fields in the visual cortex based on the temporal slowness principle.

- May/Aug. 2001: Diploma thesis in Neuroinformatics at the Institute for Neuroinformatics (INI), Zuerich, with Dr. Konrad Koering and Prof. Peter Koenig. Title: Learning of disparity selective neurons from natural images.

- 20002001: Part-time work as a research assistant at the Institute of Virtual Manufacturing (IVP), Zuerich, with Dr. Andreas Unterkircher and Prof. J. Reissner. Research subject: implementation of an efficient algorithm for parallel stiffness matrix assembling in a finite elements software environment.
- 19962001: Studies of Mathematics at the Swiss Federal Institute of Technology (ETH), Zuerich.

Alberto Berti

Alberto Berti works as a software developer since 1996, and as an independant consultant since 2001. Starts developing database applications and then specializes in working with Python and Zope for system integration and web development.

Aaron Bingham

A native of western Canada, Aaron got his introduction to Python while developing the Komodo IDE at ActiveState in Vancouver. After completing his Bachelor's degree in Computer Engineering at Simon Fraser University in 2002, Aaron moved to Stuttgart, Germany, where he worked as a Python consultant until last summer, when he joined Cenix BioScience in Dresden. At Cenix, Aaron is one of 3 developers working on Cenix's Laboratory Information Management System (LIMS).

kit blake

Kit Blake steers the business course of Infrae, a software development company located in Rotterdam, the Netherlands, which he co-founded in 2001 with Martijn Faassen. The company builds open source tools for content, including the Silva CMS, the Kupu in-browser editor, and the Railroad Repository. Kit Blake is active in the Zope Europe Association, OSCOM, and V2_Organization. Before starting Infrae he managed a series of now defunct companies.

Carl Friedrich Bolz

Duncan Booth

Duncan read Computer Science at Cambridge graduating in 1983 and has worked in many areas including compiler development, foreign exchange dealing, real-time financial data, healthcare systems and website development. He has programmed in many languages (Algol68, BCPL, C/C++, ...) but in 1997 he reached the letter P and got hooked on Python. He currently works for Oxfam doing Plone development.

Pravir Chawdhry

Pravir Chawdhry is system architect and project manager of the EEJNet project. He has been leading Zope applications in real life projects since 2002.

Since April 2001, he has been working at the Institute for the Protection and Security of the Citizen (IPSC) of the Joint Research Centre of the European Commission.

Previously he has been full-time research and academic staff at the universities of Leeds, Newcastle and Bath in the United Kingdon.

Pravir obtained his PhD in system modelling and control in 1985 from the Queen's University of Belfast.

Rob Collins

+44 1295 812176

Richard Cooper

Software Developer Arts Alliance Media Ltd London, United Kingdom

Johan Dahlin

Guy Dalberto

French, born 1951, married, two grown sons, living and working in Ardèche, 100km south of Lyon
fluent english, INPG ingénieur 1974 in fundamental physics wrote software since 1976 + 1976-1983
medical lab automatisaton, database tools, asm, fortran, cobol, pascal + 1983-1990 network
analysers (ISDN, X21, X25), software manager of a start-up (C, asm68000, realtime OS and Unix) +
1990-1992, private radio-telephony networks, software manager of small company (realtime OS, C,
68302, 68HC11) + 1993-1999 half-time university professor in Valence, industrial systems and Java.
Since 2002 I teach only a small course : OOP in Python. + 1992-2005 freelance, ISDN
analyzers,T&M systems, (C++, VB, Python since 2000)

Theo F. de Ridder

Theo de Ridder walked around in the software landscape for more than 35 years. At this moment is
he is lector (~associate professor) in software-engineering for real-time and embedded systems at the
Saxion Hogeschool in Enschede (NL). He is interested in reducing complexity by using Python at the
highest and lowest architectural levels of systems integrating physical and virtual realities.

Remi Delon

Remi has been using Python for more than 7 years. He is the author of CherryPy, a pythonic web
development framework and runs Python-Hosting.com, which offers commercial Python web hosting
and provides free Trac/Subversion hosting to more than 50 open-source Python projects.

Remi Delon

Software Developer Arts Alliance Media Ltd London, United Kingdom

André Deparade

Born on 1969-11-18. MSc. / Dipl.-Inform. in Computer Science at University of Dortmund in 1996;
topics including programming languages, software architecture, modelling and simulation.

Worked at RWTH Aachen (Technical University) until 1998 on software architecture and hypertext.
Worked at University of Dortmund / Process Control Laboratory until 2002 on software architecture,
modelling and simulation of hybrid systems.

Working at Cyprès IT-Consulting / systems department. Topics include operating, management, and
automation of hosts and database servers.

Vincenzo di Somma

Riccardo Lemmi is a computer science engineer particularly interested in object oriented
programming and software engineering methodologies. He has more than 4 years of experience in
java and python web applications programming.

Beatrice During

Anders Egneus

Anders graduated in biology and worked as a teacher for several years before going back to university and taking a degree in computer engineering. He has been working with C-Lab since 2003.

Paul Everitt

Martijn Faassen

Martijn Faassen is a well-known Python and Zope developer. He is co-founder of Infrae. He has had his hand in projects like the Silva CMS, Five, and much XML related software, most recently lxml.

Hans Fangohr

Hans is a computational physicist and lecturer in Computational Methods at the University of Southampton (UK). He obtained his undergraduate degree in Physics at the University of Hamburg (Germany) before he did his PhD in the Computer Science department at the University of Southampton. Apart from working on a number of research projects, he is teaching Engineering students the basics of numerical methods and software design.

Farzad Faramarzi

Farzad graduated in molecular biology and computer engineering, and has been working at AstraZeneca since 2000. He has been working with C-Lab since 2004.

Stefane Fermigier

PhD in mathematics. Python developer since 1996. Zope developer since 1999. Core CPS developer. CEO of Nuxeo.

mathias foehr

Mathias Foehr, the project manager, is an independant actuary and IT consultant. An adept of object oriented technologies since the times of Simula, Mathias has managed various IT projects related to the finance sector such as trading automation, life assurance and investment funds. Mathias is the president of the Luxembourg Actuarial Association.

Volker Gietz

I am 27 and have been interested in audio programming, computer science and electronic music for over a decade. Things started off as sound chip programming, along with a small MIDI-studio and have gradually emerged into creating my own audio effects and instruments. I also did desktop video for a while, so I got in contact with different A/V editing tools and learned about their editing concepts. Over the time I began to see similarities between and within these applications which could possibly be "refactored" as to create a generic editing tool. Two years ago, Python was added to the mix which not only allowed for easy experimentation, but also rapid implementation of a basic editing framework. Defeating my C++ reflexes, I moved more and more program logic into the Python domain which sped up development and made the environment a lot more "humane" to work with. Python surely has been of great use in bringing this project forward.

Katherine Goodwin

Kat Goodwin is a software developer at Clockwork Software Systems, with an academic background in teaching and educational software, but an aversion to actual schoolchildren. She specialises in system design and professional development training. In her spare time, she works on the Linux InfoPoint Project, an organisation which peddles Ubuntu CDs through the harassment of innocent members of the public at computer fairs.

Florent Guillaume

Core Zope2 and Zope3 developer. Lead CPS architect. R&D director of Nuxeo.

Jacob Hallén

Jacob Hallén is the head organiser for Europython 2005 and CTO of AB Strakt. He is the main designer of the registration application for Europython. Jacob has a background in teaching various computer and programming subjects. He has also been the CEO of one of the first Internet oriented companies in Sweden and a standardisation expert for the Royal Library; the National Library of Sweden.

Jacob dislikes all Content Management Systems and will happily tell you why, if you have enough time to spare.

Tero Hasu

Astrid Hellwig

Educational background: Economist (Diplom-Volkswirtin); Professional experience: 5 years of experience as consultant in economies of transition; 7 years of experience as trainer for business and social skills, Some preferred topics: management, leadership and presentational skills. Age: 35 Hobbies: meeting people around the world; books, drinks and sports.

Stefan H. Holek

Stefan is a well known figure in the Zope and Plone communities. He is the author of the ZopeTestCase framework for Zope2 and the current release manager of Plone.

While his talks usually focus on the various aspects of software testing, this year he didn't feel like yet another testing talk at all.

Michael Hudson

Michael Hudson is a mathematics student from the UK.

He is a member of the core Python development team, and is interested in dynamic programming languages and their implementation.

Juan David Ibáñez Palomar

J. David Ibáñez is the CTO of itaapy, a company where Python is at the core of its business. He is the lead developer of the itools framework and the iKaaro Content Management System. He has contributed to the gettext module of the Python Standard Library, and is best known in the Zope community for developing the Localizer product.

Andreas Rübner Johnsen

Developer and project manager at Bouvet - a Norwegian consultancy company with about 150 employees that deliver projects within information and communication technology.

The focus the last four years has been Topic Maps based solution, user centered development and open source.

Steven Johnston

Andreas Jung

- born in 1967 - 1995 MS in computer science and electrical engineering - 1995 - 2001 developer and system analyst for Saarbrücker Zeitung, a large publisher in Germany, working in the ePublishing department and in charge for several large EU projects - 2001-2002 software developer for Zope Corporation, Fredericksburg, USA - 2002-2003 software developer in the ePublishing department for Haufe Mediengruppe, Freiburg, Germany - since 2004: free consultant and developer

other activities: - co-founder of the German Zope User Group - current Zope 2 release manager - regular speaker at Europython, Plone conference, Linux Expo, LinuxTag and DZUG conferences

Norman Khine

Norman Khine has a BEng in Energy Engineering from South Bank University. He has eight years experience of IT systems in a publishing business environment with one year spent working at an Internet Service Provider. Today he works as IT Manager at BMI Publications, where he is at the head of the Traveluni project, leader in the e-learning market within the International Tourist Industry.

Mathias Klang

holger krekel

Holger Krekel, born 1969 began in 1985 to work as a programmer producing games for Electronic Arts. He studied computer science and consulted for Volkswagen, German banks and the chairman of the EU-founded CEN/ISSS workshop who contracted him to prototype integration of OpenSource software. He implemented an open-source transaction service on top of TAO/CORBA and published several articles about Free projects. In 2001 he joined the Python community, published interactive tools, took part in Zope3 development and some of the first coding Sprints in Europe and became one of the initiators of the PyPy project, now funded by the European Union.

Holger Krekel

Mikel Larreategi

Mikel Larreategi is a software engineer at CodeSyntax, with a degree in software engineering by the University of the Basque Country (2004). His final year project was related to workflow systems based in Zope. Once he got his degree, he was immediatly hired by CodeSyntax. He has experience with Python and Zope, and with Localizer particularly. He proudly edits a bilingual blog, as well, localized with Localizer <http://www.eibar.org/blogak/erral>

Jukka Laurila

Research Engineer at the Nokia Research Center Software and Application Technologies Laboratory. A Python enthusiast still amazed he's being paid for having so much fun :)

Marc-André Lemburg

Marc-Andre is the CEO and founder of eGenix.com, a Python-focussed consulting company based in Germany. He has a degree in mathematics from the University of Düsseldorf. His work with and for Python started in Winter 1993/4. In 1997, he became a Python Core Developer and served on the board and as vice-president of the Python Software Foundation (PSF). Marc-Andre is the author of the well-known mx Extensions, e.g. mxTools, mxDateTime and mxODBC, which are now distributed and maintained through eGenix.com. Today he spends most of his time consulting for companies investing in Python and managing projects heavily relying on Python.

Alexander Limi

Alexander Limi has a background from Psychology and Computer Science, and is one of the original Plone Founders. His main area of expertise is User Interface and Interaction Design and he is responsible for most of the interface design of Plone. In addition, he helped pioneer the use of multilingual interfaces in the Zope and Plone worlds, put accessibility on the agenda for open source web frameworks, and has a strong interest in most levels of the technology stack. He is vice president in the non-profit Plone Foundation, and in his spare time he is passionate about music and design.

Johan Lindberg

I work as a Java developer at Nordea Life & Pensions in Sweden. I've been using Python for three years.

Arnar Lundesgaard

Developer and consultant at Bouvet - a Norwegian IT-consulting company with about 150 employees.

Arnar has been developing Topic Map applications on Zope for the past 3 years.

Benno Luthiger

Benno Luthiger first studied theoretical physics and then added a study in cultural anthropology. Since 2001 he concentrates on doing open source. In the context of his Ph.D. research he started the FASD study to investigate the theoretical aspects of the open source development model. As specialist for open source at the IT services at the ETH Zurich he deals with the technological aspect of open source software.

Magnus Lyckå

Magnus studied electronic engineering at Chalmers and got his M.Sc in 1990. After Chalmers, he worked as an electronic design engineer at Saab Ericsson Space, but software was much more fun, so since 1996, he's worked as a software developer and consultant, mainly in his own company

Thinkware AB, with large database systems and object oriented design and programming. He learned Python in 1997, and has found good use for it with every client the last seven years.

Magnus is currently employed by Carmen Systems in Göteborg, where he works in R&D with Carmen's versioned data management system Dave. He spends his time programming C++, Python and SQL on Linux and Unix machines.

Kevin Manley

Lead Software Developer London, United Kingdom

Raphael Marvie

Raphael Marvie is assistant professor at the University of Lille 1 (CS Research Lab). He obtained a MSc in Computer Science in 1999, and a PhD in Computer Science in 2002. His PhD was dealing with the handling of component based software architecture using meta-modeling.

Member of the Object Management Group (OMG) since 1999, he has contributed to CORBA Component Model specification. He is currently contributing to the Model Driven Architecture especially the definition of a standard for model transformation (MOF 2.0 Query / View / Transformation).

Raphael Marvie teaches object oriented programming and design, distributed computing, and component based technologies to students in Bachelors and Masters in computer science. He also teaches these technologies and does consulting for small to large companies.

Finally, after using other scripting languages for a while, Raphael Marvie has started using Python in 2003. Most of his current software developments are performed using Python, both for research prototypes and for industrial development. He teaches Python here and there, and is writing a book related to software engineering using Python as the basis of all experiments.

Stefano Masini

Computer scientist as a vocation, graduated at the University of Bologna, Italy. Then studied in USA, and took a Masters in Computer Science from the University of Texas at Austin. Eager to tackle real world challenges joined Pragma 2000 and dived into Python to discovered a great way to bring his contribution to the publishing industry.

Harald Armin Massa

Harald Armin is self employed with his company GHUM Harald Massa - persuasion python postgresql.

His main objectives are development of per customer specific software and persuasion-seminars concerning process realisation, business economics and persuasion techniques.

He was speaker of the famous "stop promoting python" lightning talk at EP 2003 in Charlesroi, performer of "Selling Pythoners" at EP 2004 Goetheborg and at that time Google Top Result for "Fetish Based Selling".

Gintautas Miliauskas

Currently working at Programmers of Vilnius (www.pov.lt), doing web application development on Zope 3 based projects, including, but not limited to SchoolTool & SchoolBell. Student (2nd year

Bachelor's) in Vilnius University.

Stuart Murdock

Post-Doctoral Fellow - University of Southampton E-Science / Chemistry

Ph.D Computational Chemistry

BSc. Joint Applied Maths and Physics

Research Experience in Ireland, U.K. India, USA

Gustavo Niemeyer

Paul J. Nolan

Professor Paul J. Nolan Department of Mechanical Engineering National University of Ireland, Galway

Jean-Marc Orliaguet

MSc. Lead CPSSkins architect. Core CPS and Zope developer.

Rob Page

Kalle Pahajoki

Kalle Pahajoki is a long time Python user and a fourth year Mathematical Information Technology student from the University of Jyväskylä, Finland. Recently, he has focused on developing a Python based bioimaging software.

Tim Parkin

Tim works for Pollenation Internet Ltd as Managing Director and tried to walk the fine line between design and coding. Our software development is now predominantly Python and we have a company commitment to release our developed tools and software as bsd licensed open source.

John Pinner

Brian Quinlan

Brian Quinlan is a Python developer with Scionics Computer Innovation in Germany. He has been using Python for over five years in both academic and business environments. His activities in the Python community include the development of modules included in the Python Standard Library as well as several external open-source modules. He has also presented at and co-convended Python classes and conferences.

Prabhu Ramachandran

Prabhu received his B.Tech. and Ph.D. in Aerospace Engineering from the Indian Institute of Technology Madras, Chennai, India in 1997 and 2005 respectively.

He has been involved with Open Source Software development and is the developer of MayaVi (<http://mayavi.sf.net>). His research interests include vortex methods, computational fluid dynamics and scientific computing.

He is currently consulting for Enthought Inc. In October he will join IIT Bombay as a member of the faculty in the Department of Aerospace Engineering.

Edward Ream

Edward K. Ream has been programming for over 30 years. He is the creator and project leader for Leo, an OpenSource program written entirely in Python.

<http://webpages.charter.net/edreamleo/front.html>

Reinout van Rees

I'm a civil engineer from the Netherlands. I've got a lovely wife and two kids (girl almost three, boy almost one). I'm now part of Zest Software, making non-profits and government organisations happy with zope and plone.

After graduation I've spend 4.5 years on a PhD in Civil Engineering Informatics: improving the construction industry's information exchange through simple semantic web solutions, greatly helped by python and plone. Aaaaaalmost finished now.

I've got a site and a weblog at <http://vanrees.org/> if you're desperate for more information :-)

Armin Rigo

Ken Rimey

André Roberge

André became interested in Python less than a year ago, as he was looking for a way to introduce his children to computer programming. After having written a few chapters of a Python tutorial designed for children, he came across Guido van Robot (GvR). André had a few ideas about extending GvR and, not realising at the time that he could join the GvR team, he set out to write his own 'improved' version of GvR, which became RUR-PLE.

André obtained a Ph.D. in Physics from the University of British Columbia in 1989, after which he worked as a physicist at Memorial University of Newfoundland and, later, at Laurentian University. Since 2001, he has been President and Vice-Chancellor of Université Sainte-Anne in Nova Scotia, Canada.

Maik Roeder

Maik Roeder

Mario Ruggier

Mario Ruggier has been following the evolution of the Web since its origins at CERN, where he has developed numerous web-based information applications. In 2001 he left CERN to work for a start-up on an extensive web user interface, at which time he discovered python. His primary interests are end-user usability of web applications, as well as architecture and implementation of same. Currently he is co-founder and technical director of a new information company, Scoopics Infotainment, that is using python as its technology base, in the context of which he has conceived and developed Moellus.

Tobias Rundström

Projectlead of XMMS generation 2. Has been working with the XMMS team for over two years. Professionaly works at Netintact AB where I do the PythonAPI for our main product PacketLogic. Other opensource projects include GSTP (Good Secure Transfer Protocol).

Michael Salib

Michael Salib is an engineer at Divmod. He's actually giving this presentation on behalf of Glyph Lefkowitz, another Divmod engineer who is unable to attend.

Michael Salib

Michael Salib is an engineer at Divmod.

Henrik Sandklef

Rodrigo Senra

I'm the Lead Developer/Architect of GPr Systems (Brazilian softhouse) since 1996. I was granted my MSc in Computational Reflection in 2001, and now I'm a part-time PhD student at Unicamp. Since 2000 I've done some **tiny** contributions to several projects: Guaraná, GDK/dejavu, sylpheed-claws, Python, Zope, Localizer and others. I'm also a Python herald in Brazil since 2000.

<http://www.ic.unicamp.br/~921234> <http://www.gpr.com.br>

Michele Simionato

Michele Simionato works as a software consultant and a trainer. His major expertise is in building Web applications with Python.

Michele is known in the Python community for his posts on comp.lang.python and for his articles published on various on-line journals and on the official Python Home-Page.

His interests are in web programming, text processing, and in general in object oriented programming and metaprogramming. Recently he spent some time hacking Scheme and he is always looking to improve his knowledge.

Michele was born a Physicist; he got a Ph. D. from the University of Parme, in Italy and he worked as a post-doctoral researcher in France (University of Paris) and in the U.S. (University of Pittsburgh).

Aroldo Souza-Leite

Michael Sparks

Michael Sparks is a Senior R&D Engineer at the BBC. His work involves researching and creating technologies capable of scaling the delivery of BBC content over the Internet. He's contributed on the squid, linux virtual server, and python projects amongst others, and runs the BBC's Kamaelia open source project -- a sister project to the BBC's Dirac project. In his spare time he likes coding collaboration tools and technologies, and playing with code.

Christian Theune

Christian Tismer

Heikki Toivonen

Heikki Toivonen has been working for the Open Source Applications Foundation (OSAF) since 2003. OSAF is working on a new cross platform Personal Information Manager called Chandler. He has been working on the security aspects of Chandler, including SSL/TLS support. Before joining OSAF, he spent three years hacking Mozilla at Netscape Communications Corporation.

Kirby Urner

Kirby Urner

ditto (I gave two email addresses)

Andi Vajda

I work for the Open Source Applications Foundation (OSAF) where I am responsible for the Chandler repository, PyLucene and PyICU projects.

Guido van Rossum

Tommi Virtanen

Tommi Virtanen runs his own company focusing on medium-size office servers and services, software development and consulting. He's a Debian developer, Twisted person and a kernel hacker. His first encounter with Python (and Twisted) was in 2001.

Michael Volkmer

Philipp von Weitershausen

Philipp studies Physics at the Dresden University of Technology, Germany. He's a Zope 3 developer, Zope 3 book author and a developer of the Zope2-to-Zope3-migration product Five.

arjen Voogt

Grégoire Weber

Gregoire Weber, from Zürich (<http://www.incept.ch>), has a master of electrical engineering working as professional software developer since more than 10 years. He's strong in breaking down complex requirements in concise architectures and is rigorous in squashing out bugs. He discovered Zope in 1999 and wrote since then a half dozens of Products for Zope/Plone.

Joe Wrigley

Joe Wrigley is a software developer at Clockwork Software Systems. Within Clockwork, Joe specialises in graphical software development, and has worked on projects such as an emissions testing system for a large car manufacturer. He has also been actively involved in the implementation of, and development of software using the Thyme framework. In his leisure time, he has an avid interest in DIY.

Tarek Ziade

Core CPS and Zope developer. R&D engineer at Nuxeo.

Manuel Zini

Born 11/1/71, I graduated in Computer Engineering in 2000 at the University of Florence, Italy. I've been working for Lucent Technologies (Swindon, UK) at the UMTS/RNC project in 2001, since then I worked primarily as an information integration consultant mainly for the Italian Highways and for one of the biggest Italian hospitals. Since 2003 I'm designing intranet collaboration systems, mainly implemented in Plone. I'm also currently designing information retrieval systems for intranets, and I started the Isobel Project in February 2005.

Tiziano Zito

- since July 2002: PhD student at the Neural Computation Research Group of the Institute for Theoretical Biology (ITB), Berlin, with Dr. Laurenz Wiskott. Research Subject: Integrating temporal slowness principle and independent component analysis.

- Sept. 2001- July 2002: Research assistant at the International Max Planck Research School on Biomimetic Systems in the Max-Planck-Institute for Colloids and Interfaces. Research subject: Ecological networks and models in theoretical ecology.

- Aug. 2001: Summer school "Tools to Simulate Turbulence on Supercomputers" at the Helmholtz-Institute for Supercomputational Physics (Potsdam University - Germany).

- Nov. 2000- June 2001: Laurea thesis written at the Max-Planck-Institute for Colloids and Interfaces (Golm, Potsdam - Germany) in collaboration with Dr. Christian Seidel, supervisor in Italy: Prof. Graziano Servizi. Title: Grand Canonical Monte Carlo Simulation Of Polymer Chains With A Fluctuating Charge Distribution (Annealed Polyelectrolytes).

- 1996-2001: Studies of Theoretical Physics at the University of Bologna, Italy.



ConferenceKit

...or, how we built this personalised schedule

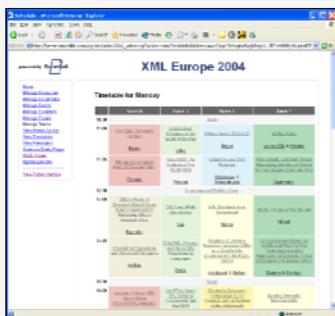
Generate all conference materials for web, download and print...



Personalised programmes for every delegate based on their own track preferences



Graphically rich data-driven timetables with complex spanning for professional or on-site printing



Generate web timetables, event and speaker listings for inclusion in your site

This document was produced using the ReportLab ConferenceKit

Our ConferenceKit is a hosted document production solution for the conference organiser, using our high-speed XML-to-PDF engine to generate personalized materials for print and web. A standard database and web interface store details of speakers, tracks, talks and exhibitors as well as the 'document recipes' which can be configured, reused and extended from event to event. Use it to produce...

- High quality, up-to date timetables and show guides for professional printing
- Web site content with timetables, linked speaker and talk descriptions
- Fully personalised PDF programmes for every delegate
- High quality personalized documents for professional digital printing

The ConferenceKit is designed to make events work better for everyone.

ReportLab - the print button for eBusiness

Reportlab implements reporting solutions which generate PDF documents on demand on web servers from any data source. These build on our PDF and graphics tools, which create sophisticated PDF reports directly. Our solutions are at work in some of the world's largest and most demanding financial firms.

We're doing something new, spanning three traditionally disparate software categories: traditional groups-and-totals database report generators; high-volume print formatting systems used for invoices and statements; and database publishing tools able to produce personalized professionally-printed documents. The existing tools share common limitations: they don't always run on your platform; data has to come from at most one relational database; they were designed with paper output in mind and are slow; there are limitations on formatting; and reuse across a family of reports is difficult or impossible.

ReportLab's solutions are designed to meet the needs of modern e-Business. They can run on any platform, from Unix web servers to Mac or Windows desktops to embedded print engines. They can acquire data from any source – XML, databases, or business objects in your applications. They output PDF, which can be printed, stored, downloaded or emailed as needed. They have no limits on page layouts or graphics. Styles can be shared and reused intelligently across families of reports, exactly as with web sites. And they are FAST!

To learn more about how ReportLab can enhance your business, visit www.reportlab.com