Western Mediterranean
October 10–17, 2004

Perl Whirl • Linux Lunacy • MySQL Swell

SPEAKERS INCLUDE:
Brian Aker
Kaj Arnö
David Axmark
Piers Cawley
Dave Cross
Christopher R. Hertel
Rasmus Lerdorf
Marty Pauley
Steven Pritchard
Allison Randal
Randal Schwartz
Robert Spier
Larry Stefonic
Mauro Tortonesi
Theodore Ts’o
Larry Wall
Michael “Monty” Widenius
Casey West

Escape to the good life.
Allow yourself
to be pampered.
Unwind. Have fun.
Enrich your mind.
And do it all aboard a
luxurious cruise ship.

Don’t miss
this opportunity.
Space is very limited.
Managing Risks: How Parano...
The course is presented as a series of sections on management, and scalability. Topics include: the database, database interaction and abstraction, application logic, testing, session and user interaction and management, session and user abstraction, application logic, testing, and scalability.

When you get off the boat, you will have a functional road map for further exploration, or to get your job done faster when you get off the boat.

Perl is blessed with a great set of simple testing tools. In this class you’ll learn how to use them to incrementally build a test suite that helps to drive your development. Over the course of the day we will develop real software using the deceptively simple test/code/refactor cycle of development that’s at the heart of Test Driven Development.

Because we’re going to be driven by our requirements and tests, it’s hard to say what exactly will be covered on the course, but you should expect to see coverage of:

- Test::More and the Test::Harness family of tools
- Test::Class (my favourite testing tool)
- Other Test::* packages as requirements demand
- Writing better tests
- Refactoring with Perl
- Why Perl is great for refactoring
- Why Perl isn’t great for refactoring
- Why Perl 6 will be better
- Code Smells and what to do about them
- Listening to your code

Test Driven Development with Perl (full day)
Speaker: Piers Cavley
Perl is blessed with a great set of simple testing tools. In this class you’ll learn how to use them to incrementally build a test suite that helps to drive your development. Over the course of the day we will develop real software using the deceptively simple test/code/refactor cycle of development that’s at the heart of Test Driven Development.

Because we’re going to be driven by our requirements and tests, it’s hard to say what exactly will be covered on the course, but you should expect to see coverage of:

- Test::More and the Test::Harness family of tools
- Test::Class (my favourite testing tool)
- Other Test::* packages as requirements demand
- Refactoring with Perl
- Why Perl is great for refactoring
- Why Perl isn’t great for refactoring
- Why Perl 6 will be better
- Code Smells and what to do about them
- Listening to your code

Introduction to Perl 6 (full day)
Speaker: Allison Randal
This talk is an introduction to Perl 6, the next major version of Perl. The primary focus is on the syntax of Perl 6: what’s changed from Perl 5, what’s the same, and what you can do with the language today. Topics include operator changes, the extended importance of closures, smart matching, switches and loops, exceptions, argument binding, aliasing, properties, and exciting new features like junctions and grammars.

Beyond the syntax, we’ll briefly look at some of the reasons behind the changes. What makes one feature more desirable than another? What makes it Perlish? What makes it DWINa? How on earth does Larry decide? You’ll leave with a better understanding of the changes in Perl 6, and the fundamental nature of Perl itself.

Learning Perl Objects (half day)
Speaker: Randal Schwartz
Randal Schwartz presents an introduction to Perl's object system, including creating classes and instances, defining inheritance, overriding, persistence, object destruction, and using AUTO-LOAD. Prior knowledge of other object systems is helpful but not required. Prior knowledge of Perl basics, including packages and references, will be assumed.

Idiomatic Perl (half day)
Speaker: Dave Cross
This tutorial teaches you the bits of Perl that most beginners miss out, enabling you to take your Perl expertise to the next level.

Topics covered include:

- use strict/use warnings
- package and lexical variables
- references
- finding, installing, and using modules
- reusing code
- sorting
- Perl special variables (aka cartoon swearing)
- and a number of other techniques.

Tieing and Overloading Objects in Perl (half day)
Speaker: Dave Cross
Tieing and overloading are two techniques that can make your Perl objects easier to use, but they are often underused. This tutorial explains how to use these powerful techniques in your object code.

Topics covered include:

- Why tie or overload
- Tieing objects: What you can tie; Using tie; Being lazy (using Tie::StdFoo); Easier tie interfaces (Attribute::Handlers); Extended examples
- Overloading objects: Overloaded methods vs. overloaded operators; Overloading operators; Stringification and numericification; Copy constructors; Overloading constants; Extended examples

Writing Better Perl (half day)
Speaker: Marty Pauley
No matter how long you have been programming in Perl, or how great your code is, you can still improve. In this talk I will explain how you, as a beginner or expert, can write better Perl. I also hope to show how you can identify other ways to help yourself improve, because I don’t have all the answers!

Optimize Your Perl Code (half day)
Speaker: Robert Spier
This tutorial is an introduction to optimizing Perl applications. It’s important knowledge for anyone writing time-sensitive applications, from a CGI script to a large application.

It covers the following four major components:

1. A brief introduction to complexity theory: Just enough complexity theory to have a basic language to talk about runtimes of functions. O(1), O(n), O(n^2), etc. Complexity of common Perl idioms like map, sort, nested loops, etc. Why is some code slow? (And it’s not Perl’s fault!)

2. Real world/real code examples and how to make them faster: Examples of O(n^2) or worse functions and how they can be improved by an order of magnitude by rewriting the algorithm, and possibly changing the data structures used to store things. Memoization/Caching/Persistency/Precomputing.

3. Some simple Benchmarking: Devel::TProf and Benchmark.pm can help to isolate problem spots in your code. Learn how to properly interpret data and how to pick the “important” pieces.

4. Is it worth it? Not all time spent optimizing the code is worth it. There are tradeoffs of time, readability, and speed. We will discuss development strategies, pre-optimization, and over-optimization.

Filtering Your Email with Perl (quarter day)
Speaker: Casey West
You will learn how to filter your email with Perl. Taking advantage of Perl’s parsing abilities and CPAN resources, you’ll take back control of your inbox. Topics discussed in this recipe-style course include parsing email, filtering incoming email, sorting email and mailing lists, blacklists and white-lists, handling attachments, auto-responders, reformattign email, virus and spam protection, and more. Basic Perl knowledge is recommended.

Template Toolkit for Non-Web Applications (quarter day)
Speaker: Marty Pauley
www.template-toolkit.org says “The Template Toolkit is a fast, powerful, and extensible template processing system. It is ideally suited to the production and maintenance of web content and other dynamic document systems.”

Template Toolkit is an excellent system for web content, but it is also really useful for many other tasks.

If you’ve thought about using the m4 macro language to help manage your system configuration files, but you didn’t like it, you should try Template Toolkit.

If you need to run a mail merge but you hate Winodze and/or Word, you should try Template Toolkit. If you’re being forced to use Java and your IDE won’t write at least half your code, you should try Template Toolkit. Or in Perl, if h2xs and ModuleMaker don’t do it for you, you should try Template Toolkit.

This tutorial will explain how to use Template Toolkit in the above situations.
IPv6: From Theory To Practice (quarter day)
Speaker: Mauro Tortonesi
IPv6 is short for "Internet Protocol Version 6." IPv6 is the "next generation" protocol designed by the IETF to replace the current version Internet Protocol, IP Version 4 ("IPv4").

Most of today's Internet uses IPv4. But this 20-year old standard is now showing its age. There is a growing shortage of IPv4 addresses (n.n.n.n), which are needed by all new environments.

In this seminar we cover the new features introduced by IPv6 as well as the problems posed by the transition from an IPv4-only Internet to an IPv6-based one.

We'll also discuss:
• how to set up IPv6 connectivity for SOHO environments.
• problems developing IPv6-enabled software.
• application design guidelines and software tools to speed up the porting of networking applications to IPv6.

---

The MySQL Grand Tour (4 hours)
Speakers: Brian Aker, David Axmark, Michael "Monty" Widenius
Your MySQL tour guides Brian, David, and Monty will take you on a scenic afternoon tour of MySQL land. Highlights of the tour will include the lush green forests of stable functionality, the cool forests of little-known functionality, the ripening fruit of new features, and the dragons and fire of SQL standard compliance.

We recommend that you bring a notepad or laptop, a towel for mopping off the sweat and a few cool drinks with a parasol in them.

---

The Case for MySQL in the Enterprise (1 hour)
Speaker: Larry Stefonic
Using a combination of case studies, solid industry knowledge, and glib wit Larry Stefonic will outline the case for MySQL’s use in the enterprise world.

---

Extending MySQL with C, Perl, and Java (1.5 hours)
Speaker: Brian Aker
Have you ever wanted to extend MySQL in strange and wonderful ways? Many people do not realize that a powerful and flexible API for doing this (via binding new storage engines like InnoDB and MyISAM into MySQL) already exists. In this session, Brian Aker will walk you through the process of writing a simple storage engine and binding it into MySQL.

Along the way, you will also learn some interesting things about the implementation of the existing storage engines and the overall design of MySQL.

This session will be most useful to you if you have strong C/C++ skills.

---

MySQL Database Tuning (1.5 hours)
Speakers: Brian Aker, Michael "Monty" Widenius
Learn how to squeeze the best performance out of your MySQL installations. Topics such as query analysis, buffer tuning, query optimization, and platform issues will be discussed.
MySQL offers UDFs (User Defined Functions) which give you the ability to extend MySQL for data warehousing needs. Using UDFs, you can build functions that do complex data analysis inside of the database. The functionality of UDFs is not limited to traditional character and number data; exotic binary data like images and music can be analyzed as well.

Examples in C, Perl, and Java will show how you can take well known open-source libraries like Image-Magick and use these inside MySQL. Additional examples will be shown using myPerl. With myPerl, you can even bypass the need to learn the internal structure of MySQL UDF structure and instead use a popular and easy to use scripting language like Perl.

An Introduction to MaxDB (1 hour)
Speaker: Kaj Arnö
MaxDB by MySQL is a re-branded and enhanced version of SAP DB, SAP AG’s open source database. MaxDB is a heavy-duty, SAP-certified open source database that offers high availability, scalability, and a comprehensive feature set. MaxDB complements the MySQL database server, targeted for large mySAP ERP environments and other applications that require maximum enterprise-level database functionality.

This session provides an overview of the major features of MaxDB, along with a discussion of the current state and future plans for the product.

The Internals of the MySQL C API (1 hour)
Speakers: Brian Aker, Michael “Monty” Widenius
The C API for MySQL is the base of most available MySQL APIs — from PHP and Perl, to Lisp and Matlab. This session will take you on a detailed walk through the API, providing the attendee with the detailed knowledge required to get the most out of the C API and other APIs that are based on it.

The History of MySQL (1 hour)
Speakers: David Axmark, Michael “Monty” Widenius
David and Monty, old friends and MySQL co-founders, talk about how the MySQL project came into existence and why they chose to make it Free/Open Source Software. Learn how the project has grown from its roots as a part-time, two-man venture into one of Open Source’s success stories and hear about the challenges they have encountered along the way.

The Future of MySQL (1 hour)
Speakers: Brian Aker, Kaj Arnö, David Axmark, Larry Stefonic, Michael “Monty” Widenius
Over the last six days, the MySQL Swell attendees will have been presented with a complete picture of the current state of and future plans for MySQL. This session provides a public forum for the attendees to give their feedback on MySQL and to help shape MySQL’s future plans, products and policies.

GnuGuerrilla Speaker: Marty Pauley (45 minutes)
Often the best way to convince someone that Free Software systems can solve their problem is to install them before they can stop you! But you must be careful: “one man’s freedom fighter is another man’s terrorist.” This short talk will describe some real world GNU Guerrilla experiences and provide tips for success. If you really want to use Free Software in your workplace but cannot convince your boss, we hope this talk will help.

Kongoogo Speaker: Marty Pauley (45 minutes)
It is well known that Larry Wall stole lots of good bits from other programming languages to create and then extend Perl. But Perl’s position as the most eclectic Programming language has been threatened by Ruby, created by Yukihiro Matsumoto by stealing all the best bits from Perl.

Larry’s responded by stealing all the good bits from Ruby to use in Perl 6, but he soon realized that this would not be enough. To defeat Ruby, Larry will use his impressive linguistic skills to steal from Matz’s native language: Japanese.

This talk starts by examining the existing similarities between Perl 5 and Japanese, then looks at some of the new Perl 6 features that Larry stole from Japanese, Ruby, and the mind of a crazy Australian. If you want to join the cool kids at the front of the Perl crowd, and learn some Japanese, this talk is for you.

RT in a Nutshell Speaker: Robert Spier (90 minutes)
Tracking tasks, issues, and projects is an activity common to all business and technical disciplines, from software development to customer service. RT, the Request Tracker, is a robust, open-source ticketing system that supports efficient, agile workflow by tracking projects, issues and requests, from inception to resolution, in a community of users.

RT is a flexible, lightweight, extensible set of tools and technologies, including Web, email, and command-line interfaces; an SQL database backend; granular access control that allows users to do their work but also protects the system; a flexible ‘keywords’ system which allows a local site to keep track of ticket metadata sensibly; extensible linking that ties tickets to each other and to external databases; and clean, object-oriented design based on Perl and other open source technology.

RT has achieved broad acceptance worldwide in businesses, educational institutions, and development organizations that rely on agile workflow to provide customer support, bug tracking, software development, trouble ticketing, and other activities for which a group of folks need to keep track of a list of tasks. Join Robert to discover how to put this flexible, extensible platform to work in your organization.

Managing Your Own Open Source Project Speaker: Robert Spier (90 minutes)
How to manage an open source project on SourceForge or by hosting it yourself. Covers Mailing Lists, Downloads, Source Control (CVS/SVN), Wikis, chat rooms, and more!
PRICING AND BOOKING INFORMATION

Course Fees: $995 for the Perl Whirl, Linux Lunacy, or MySQL Swell conferences. For a flat fee of $250, attendees may take any and all of the Linux certification exams offered. Only passengers booked through Geek Cruises will be admitted.

Deposit: $400 per person, due at time of booking.

<table>
<thead>
<tr>
<th>Cabin Type</th>
<th>Cruise Rate</th>
</tr>
</thead>
<tbody>
<tr>
<td>Standard Inside</td>
<td>$1,099 (GS* available)</td>
</tr>
<tr>
<td>Better Inside</td>
<td>$1,199 (GS* available)</td>
</tr>
<tr>
<td>Standard Outside</td>
<td>$1,299 (GS* available)</td>
</tr>
<tr>
<td>Outside w/ Balcony</td>
<td>$1,399 (GS* available)</td>
</tr>
<tr>
<td>Mini Suites</td>
<td>$1,599</td>
</tr>
<tr>
<td>Full Suite</td>
<td>$1,899</td>
</tr>
</tbody>
</table>

3rd and/or 4th Person Rate: ages 19 and older, $649; ages 2 – 18, $549; under 2 years old, $499.

Single Occupancy: 150% for inside, outside cabins, as well as the Mini Suites; 200% for the Full Suite.

Port Charges and Taxes: $175 per person (subject to minor change).

Foreign Shipping Fees and Additional Payment Information: There is a shipping charge of $60 per foreign residence ($35 per Canadian residence) to cover the cost of shipping the tickets. There is a $25 charge for returned checks.

Air Add-ons: Airfare from most major U.S. cities is available through the cruise line. You can call our office for this pricing. (These rates include transfers to/from the dock/airport plus transfers to/from your hotel if we’ve booked the hotel as well.)

Pre- and Post-cruise Hotel Stays: Sightsee Venice! The hotel will be close to the dock. If we book your hotel, transfers from the hotel to the dock are included.

Physically challenged available

Guaranteed Share (GS) Fares: This plan is for passengers who are coming on a Geek Cruise by themselves and wish to share a cabin with another Geek Cruises passenger in an inside or outside cabin only. The prices are the same as the per person double occupancy rates. Share Passengers who smoke are not to do so in the cabin, unless okayed by fellow roommates. We try to match passengers with someone close in age, whenever possible. Note: Costa Cruise Lines will not accept any booking unless a fully completed Reservation Form is accompanied with a per-person deposit.

To book for Linux Lunacy go to http://www.geekcruises.com/booking/ll4_booking.htm
To book for Perl Whirl go to http://www.geekcruises.com/booking/pw4_booking.htm
To book for MySQL go to http://www.geekcruises.com/booking/sql_booking.htm

day        port          arrive          depart        conference sessions

Sunday, 10/10 Venice, Italy — 6:00pm 7:15pm, Bon Voyage Cocktail Party
Monday, 10/11 Bari, Italy 11:00am 2:00pm 8:30am – 11:00am & 2:00pm – 6:00pm
Tuesday, 10/12 Katakolon, Greece 8:00am 1:00pm 1:00pm – 5:30pm
Wednesday, 10/13 Kusadasi, Turkey 7:30am 1:00pm 1:00pm – 5:30pm & 6:00pm – 8:00pm
Thursday, 10/14 Istanbul, Turkey 8:00am 6:00pm 6:00pm – 8:00pm
Friday, 10/15 At Sea — — 8:30am – Noon & 1:30pm – 5pm
Saturday, 10/16 Dubrovnik, Croatia Noon 5:30pm 6:00pm – 8:00pm
Sunday, 10/17 Venice, Italy 10:00am —