

Background

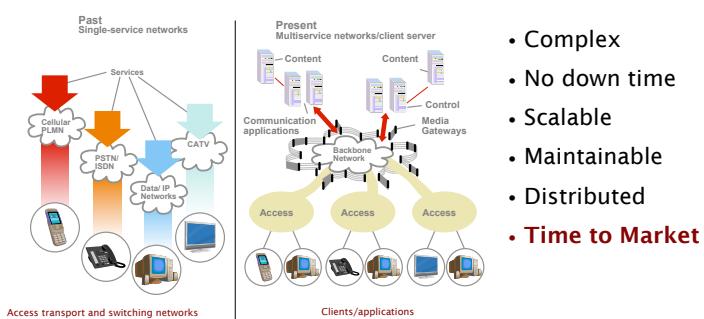


Background

- Telecom Issues
- History
- Erlang Properties
- Products
- Research
- Open Source Projects
- Download Statistics
- What's in a name?

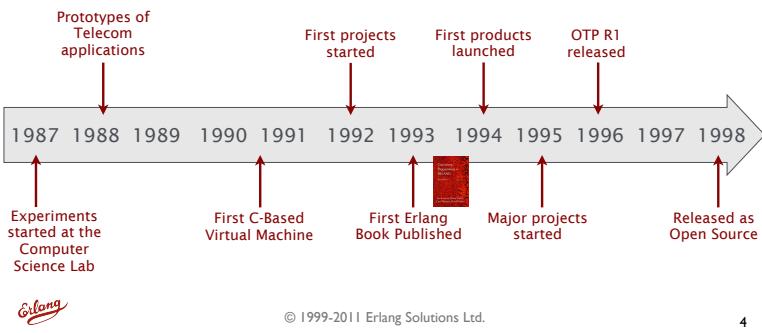


Telecom Applications: issues



Erlang History

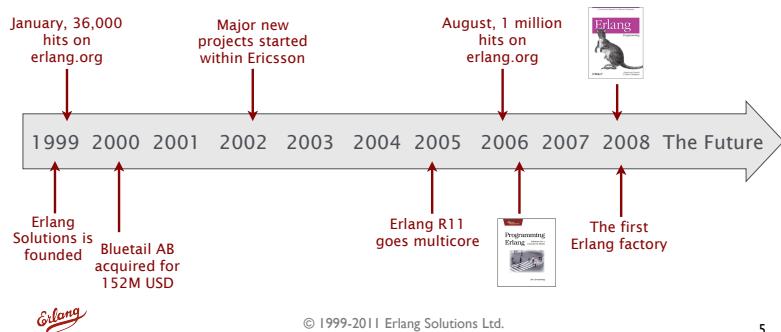
And the rest
is history...



4

Erlang History

And this is just
the beginning



5

Erlang: properties

Declarative

Functional programming language, high abstraction level, pattern matching and concise readable programs

Concurrency

Either transparent or explicit concurrency, lightweight processes and highly scalable

Soft Real-Time

Response times in the order of milliseconds, per-process garbage collection



Erlang: properties

Robustness

Simple and consistent error recovery, supervision hierarchies and "Programming for the correct case"

Distribution

Explicit or transparent distribution. Network-aware runtime system

Hot code loading

Easily change code in a running system. Enables non-stop operations. Simplifies testing.



© 1999-2011 Erlang Solutions Ltd.

7

Erlang: properties

External interfaces

"Ports" to the outside world behave as Erlang processes

Portability

Erlang runs on any UNIX, UNIX-like, Windows and OSX operating system, and also runs on Android, VX works. Supports heterogenous networks

SMP Support

Symmetric multiprocessing support. Takes full advantages of multicore architectures.



© 1999-2011 Erlang Solutions Ltd.

8

Example Products



- Yahoo!
- Delicious



- Amazon
- Simple DB



- T-Mobile (UK)
- WAP, SMS, IN services



- Facebook (USA)
- Chat channel servers



© 1999-2011 Erlang Solutions Ltd.

9

Example Products



- Ericsson
Switches, Servers,
Frameworks, etc.



- Klarna (Sweden & Europe)
e-commerce



- Telia (Sweden)
Call Center Applications



- 37 Signals
Backend Servers



© 1999-2011 Erlang Solutions Ltd.

10

Example Products: research



- HiPE
(Uppsala University, Sweden)
- Telecom Programming
(Heriot-Watt University, Scotland)
- Major Projects
(University of Coruña, Spain)
- Code Refactoring
(University of Kent, UK)
- Code Refactoring
(Eötvös Loránd University,
Hungary)
- Robots & Agents
(University of Catania, Italy)



© 1999-2011 Erlang Solutions Ltd.

11

Example Products: open source



- Disco
Large-scale data analysis
platform



- Wings 3D
3D modeller based on Nendo



- YAWS
Yet Another Web Server



- RabbitMQ
high performance messaging



- ejabberd
IM server



- Riak
Fault-tolerant distributed DB



© 1999-2011 Erlang Solutions Ltd.

12

Example Products: open source

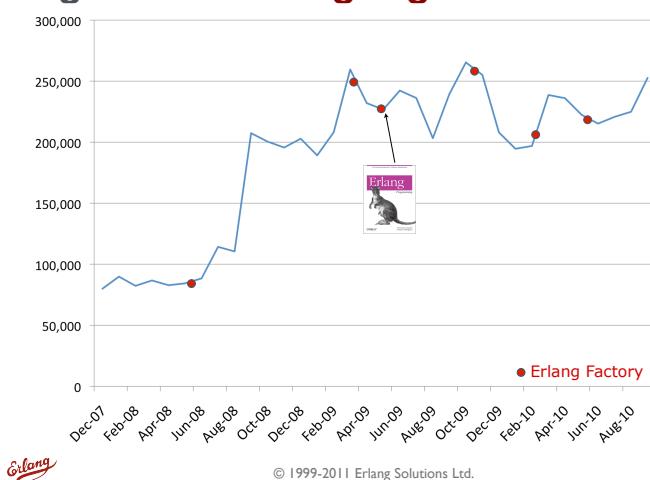


- CouchDB
Document-oriented DB
- Mochiweb
Lightweight HTTP Server
- Webmachine
RESTful toolkit/framework
- Nitrogen
Event-based web framework
- Github
BERT-RPC server in Erlang

© 1999-2011 Erlang Solutions Ltd.

13

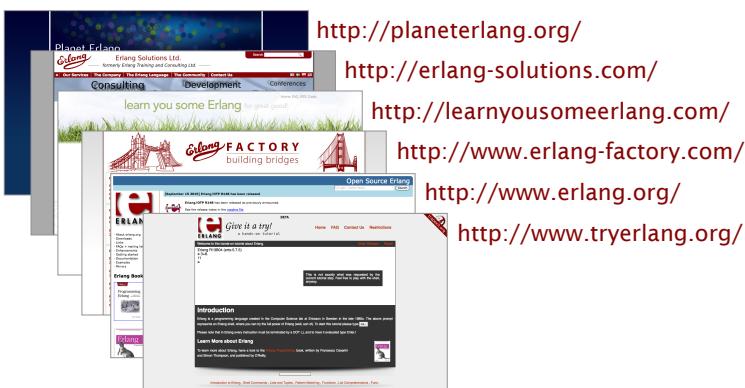
Pageviews: erlang.org



© 1999-2011 Erlang Solutions Ltd.

14

Websites and Communities



© 1999-2011 Erlang Solutions Ltd.

15

Blogs

The image shows a collage of various Erlang-related blog websites. It includes:

- Yariv's Blog**: A blog by Yariv Shermer about Open Source Erlang.
- planet.trapexit.org**: A blog by Steve Vinoski about Erlang/OTP News feeds.
- prog21.dadgum.com**: A blog by Steve Vinoski titled "Advice to Almost, Excited Programmers".
- steve.vinoski.net/blog**: Another blog by Steve Vinoski.
- erlanginside.com**: A blog by Erlang Solutions Ltd. featuring "ERLANG INSIDE" news.
- blog.basho.com**: A blog by Basho Technologies.

© 1999-2011 Erlang Solutions Ltd.

16

What's in a name?



- **Erlang, Agner Krarup, 1878-1829**
 - Danish mathematician. One of the founders of the theory of tele-traffic and queue theory used for dimensioning telephone exchanges and networks
- **An/n!**
1 + A + A²/2! + ... + An/n!
 - Erlang's formula. This formula computes the probability that all lines are busy in a system with n lines and a total traffic intensity A (measured in erlang)
- **erlang**
 - Unit of measurement for traffic intensity, e.g. the average number of simultaneous calls through a telephone exchange.

© 1999-2011 Erlang Solutions Ltd.

17