

And now for something completely different - Erlang

PlugFr Février 2012

Fabrice Nourisson



10 février 2012

©Extreme Forge. All rights reserved.

Faisons connaissance

Introduction à Erlang

Le langage plus en détail

Démo

L'écosystème Erlang

Erlang comme ingrédient du cocktail

Faisons connaissance

Introduction à Erlang

Le langage plus en détail

Démo

L'écosystème Erlang

Erlang comme ingrédient du cocktail



Fabrice Nourisson
@duckmole

- ▶ eXtreme Programming
- ▶ Erlang (for fun 2004, professionnellement 2007)
- ▶ Python, javascript
- ▶ trains, USSD (#123#), Internet (118000.fr)
- ▶ ornithorynque, ski, cut-out animation

Extreme Forge

Des logiciels fiables avec une équipe fiable

Nous utilisons des technologies puissantes et des méthodologies efficaces pour développer mieux et plus vite. Nous pouvons vous aider à faire de même.

Un langage simple, sûr et concis adapté aux environnements modernes en réseau et multi-cœur.

Erlang

Une approche pragmatique et centrée sur l'humain, pour construire des logiciels utiles ensemble.

Agile



Un logiciel libre qui rend simple le TDD (développement piloté par les tests) en Erlang.

Active Ring™

[http ://extremeforge.com](http://extremeforge.com)
@extremeforge

Faisons connaissance

Introduction à Erlang

Le langage plus en détail

Démo

L'écosystème Erlang

Erlang comme ingrédient du cocktail



- ▶ Du code *concis* (peu de lignes de code)
- ▶ Un langage *haut niveau*
- ▶ Des systèmes plus *robustes*
- ▶ Des systèmes faciles à *distribuer*
- ▶ Des architectures *scalables*

- ▶ Alonzo Church, λ -calcul, 1930
- ▶ Erlang naît dans les labos d'Ericsson en 1986
 - ▶ Dérivé de Prolog (logique)
 - ▶ Inspiré de ML (fonctionnel)
 - ▶ Plusieurs tentatives de VM
- ▶ Distribué en open source depuis 1998
- ▶ Perce dans le monde Internet en 2007

Utilisé pour l'informatique critique



Ericsson, Goldman Sachs, Mobile Interactive Group, Myriad Group, British Telecom, Klarna, Teba Bank, Ansaldo Signal, Facebook, GitHub, Amazon, ...

Faisons connaissance

Introduction à Erlang

Le langage plus en détail

Démo

L'écosystème Erlang

Erlang comme ingrédient du cocktail

- ▶ Gestion de mémoire automatique (ramasse-miettes)
- ▶ Typage implicite
- ▶ Types haut niveau : listes, tuples, symboles
- ▶ Un modèle de concurrence simple et puissant
- ▶ langage *fonctionnel*
- ▶ Affectation unique (équivalence)
- ▶ Style déclaratif : pattern matching (filtrage)

$$\begin{aligned}\text{fac}(0) &\rightarrow 1; \\ \text{fac}(N) &\rightarrow N * \text{fac}(N-1).\end{aligned}$$

A quoi sert la concurrence ?

- ▶ Systèmes réactifs, multi-tâches
- ▶ Parallélisation de certains algorithmes
- ▶ Tirer parti des processeurs modernes multi-coeurs
- ▶ Modélisation du domaine

- ▶ Basée sur une algèbre (CSP, Tony Hoare, 1978)
- ▶ Processus légers isolés (sans mémoire partagée)
- ▶ Communicants par messages asynchrones
- ▶ Parfaitement déterministe
- ▶ Simple et compréhensible
- ▶ **Automatiquement multi-cœur**

La distribution Erlang :

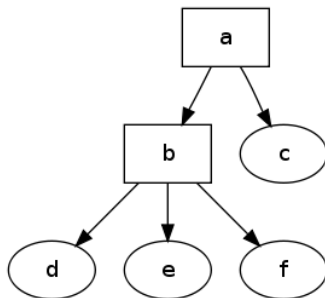
- ▶ les noeuds (VM) Erlang se connectent automatiquement (cluster)
- ▶ il y a un système de nommage distribué
- ▶ les process s'échangent des messages localement ou via réseau de façon transparente
- ▶ la supervision de noeuds et de process est aussi distribuée

Encodages binaires :

- ▶ syntaxe puissante (pattern matching, construction)
- ▶ protocoles réseau (paquets standard ou propriétaires)
- ▶ fichiers binaires

Supervision de process

- ▶ Le langage permet de surveiller l'existence d'autres process
 - ▶ `spawn_link (...)`
 - ▶ `spawn_monitor (...)`
- ▶ La notification est automatique et se fait aussi par messages
- ▶ Séparation en process métier et process de supervision (worker, supervisor)



- ▶ Génération d'environnements cible
- ▶ Versionnement
- ▶ Système d'upgrade/downgrade
- ▶ Upgrade de code à chaud (sans arrêt logiciel)
- ▶ Framework OTP (Open Telecom Platform) inclus dans le langage

Faisons connaissance

Introduction à Erlang

Le langage plus en détail

Démo

L'écosystème Erlang

Erlang comme ingrédient du cocktail

Un nuage de mots temps-réel

- ▶ Un peu de code fonctionnel (en TDD)
- ▶ Intégration dans une appli Web avec Yaws
- ▶ Changement de code à chaud



Faisons connaissance

Introduction à Erlang

Le langage plus en détail

Démo

L'écosystème Erlang

Erlang comme ingrédient du cocktail

L'environnement de développement



- ▶ Debugger
- ▶ Wrangler

- ▶ Dialyzer
- ▶ harnais de tests
- ▶ QuickCheck

- ▶ Modules
- ▶ Make ou rebar
- ▶ Hot upgrade
- ▶ Jenkins ...

Outils de supervision

Interne :

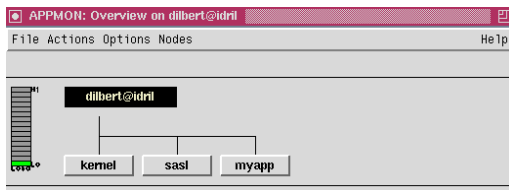
- ▶ Etop
- ▶ os_mon
- ▶ Appmon
- ▶ traces

```
etop@art                                     08:22:12
Load:   cpu      0      Memory: total      4126  binary      418
        procs    34      processes  534      code       1842
        runq     0      atom       300      ets        133
```

Pid	Name or Initial Func	Time	Reds	Memory	MsgQ	Current Function
<0.29.0>	user	"-"	5690	34000	0	group:server_loop/3
<0.39.0>	etop_txt:init/1	"-"	2104	6060	0	etop:update/1
<0.28.0>	user_drv	"-"	75	6940	0	user_drv:server_loop
<0.0.0>	init	"-"	0	9340	0	init:boot_loop/2
<0.2.0>	etop_server	"-"	0	54592	0	etop:data_handler/2
<0.3.0>	erl_prim_loader	"-"	0	13248	0	erl_prim_loader:loop
<0.5.0>	error_logger	"-"	0	3548	0	gen_event:fetch_msg/
<0.6.0>	application_controller	"-"	0	6920	0	gen_server:loop/6
<0.8.0>	proc_lib:init_p/5	"-"	0	3548	0	application_master:in
<0.9.0>	application_master:is	"-"	0	1404	0	application_master:in

Externe (SNMP) :

- ▶ Nagios
- ▶ HP Network Manager
- ▶ IBM Tivoli NetView



Bases de données

- ▶ Base temps-réel distribuée **mnesia**
- ▶ Drivers ODBC, MySQL, PostgreSQL, Oracle
- ▶ Plusieurs bases NoSQL en Erlang
- ▶ Bonne intégration avec autres bases NoSQL

	ETS	Mnesia	Tokyo	CouchDB	Riak	MongoDB	MySQL
RAM	✓	✓					
Disque		✓	✓	✓	✓	✓	✓
Gros volumes			✓	✓	✓	✓	✓
Latence	✓	✓	✓	✓	✓	✓	
Réplication		✓	✓	✓	✓	✓	✓
Répartition		✓		✓	✓	✓	
Cluster dynamique					✓		
Transactions		✓		✓			✓
Requêtes ad hoc		✓		✓		✓	✓

Ericsson :

- ▶ Développement actif du langage, des librairies, documentation
- ▶ Code sur GitHub
- ▶ Mailing list

Communauté :

- ▶ Trapexit.org
- ▶ Blogs
- ▶ Livres

Formation et support :

- ▶ Extreme Forge
- ▶ Erlang Solutions

Outils avec support commercial :

- ▶ QuickCheck
- ▶ RabbitMQ
- ▶ Riak
- ▶ CouchDB
- ▶ Process-One (ejabberd)

Etude de cas : serveur interactif USSD

- ▶ Cellicium/Myriad Group
- ▶ Selfcare téléphonie mobile (suivi conso, rechargement, call-me-back...)
- ▶ Interface core network, SI opérateur et fournisseurs contenu
- ▶ 2 personnes, 9 mois
- ▶ 99,99% dispo
- ▶ 20 millions d'utilisateurs
- ▶ 5000 MPS
- ▶ Leader marché (>30 opérateurs)



Faisons connaissance

Introduction à Erlang

Le langage plus en détail

Démo

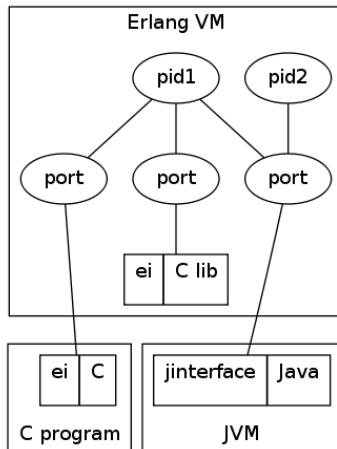
L'écosystème Erlang

Erlang comme ingrédient du cocktail

- ▶ Nombreux produits écrits en Erlang pour du non-Erlang
- ▶ Open source et/ou commerciaux

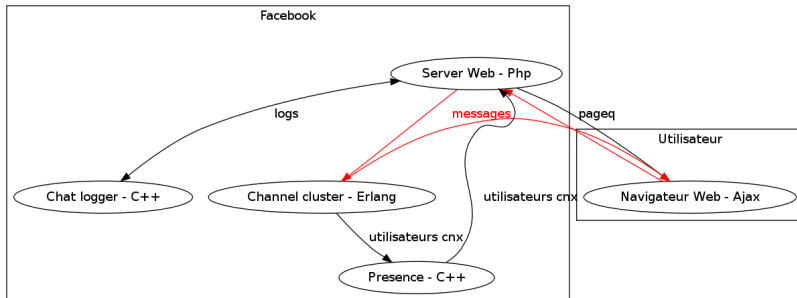
Nom	Description	Société
QuickCheck	Vérification de propriétés	QuviQ
CouchDB	Base de données	Fondation Apache
Riak	Base de données	Basho
RabbitMQ	Middleware AMQP (orienté messages)	VMware
ejabberd	Chat, messaging XMPP	Process One
U.C.Engine	Framework Web temps-réel collaboratif	AF83
Zotonic	CMS	WhatWebWhat
Disco	Map Reduce	Nokia Research

- ▶ Sockets TCP/IP
- ▶ Serveurs et clients HTTP
- ▶ SSH
- ▶ CORBA
- ▶ ASN.1
- ▶ Megaco (H.248)
- ▶ SIP
- ▶ Ports et drivers C
- ▶ Jinterface (Java)



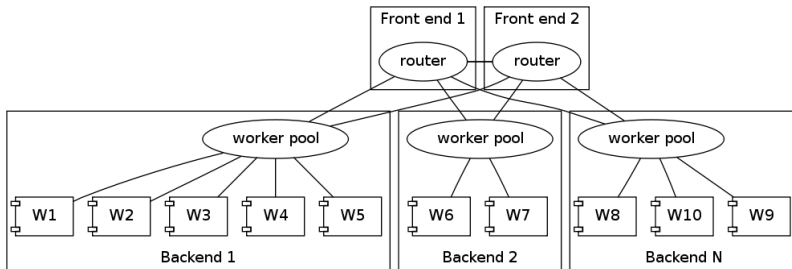
Etude de cas : Chat Facebook

- ▶ Appli de chat en ligne
- ▶ 4 personnes, 7 mois
- ▶ Ajax/PHP/C++/Erlang
- ▶ +800 millions messages/jour
- ▶ 7 millions de canaux actifs
- ▶ 100 machines



Etude de cas : routage et répartition de charge

- ▶ Utiliser la distribution et la supervision d'Erlang
- ▶ Rendre robuste, parallèle et scalable votre code existant





- ▶ Langage simple, concis, sûr et puissant
- ▶ Fait pour le multi-cœur et la scalabilité
- ▶ Facile à intégrer à un existant

Questions ?

Références

Société	Domaine	CA/M€	Utilisation
Goldman Sachs	Banque	27000	Trading
Amazon	E-commerce	24000	Base de données cloud
BT	Télécoms	24000	
Ericsson	Télécoms	22000	Switches, gateways. . .
T-Mobile	Télécoms	21000	
Motorola	Télécoms	15000	
Activision	Jeux vidéos	3100	Middleware jeux réseau
Electronic Arts	Jeux vidéos	2600	Chat
Facebook	Réseau social	1400	Chat
Ansaldo Signal	Ferroviaire	1300	Contrôle-commande signalisation
Rackspace	Hébergement	600	Backup les datacenters
Nortel	Télécoms	438	E-mail, accélérateur SSL
Meetic	Site de rencontres	186	
Myriad Group	Télécoms	71	Portail USSD
Teba Bank	Banque	2	Transactions
37signals	Gestion de projet		Chat
Beebole	Gestion de projet		AppliWeb
Blue Group	E-commerce		Paiement en-ligne
Correlatus	Télécoms		SS7
Engine Yard	Hébergement		Hébergement Ruby
Gemini technologies	Télécoms		Base de données
GitHub	Hébergement		
Heroku	Hébergement		Load balancing
Klarna	Paiement en-ligne		Tout
Smarmets	Pari		Site de paris en-ligne

Etude de cas : Switch AXD301

- ▶ Switch ATM Ericsson
- ▶ 40 personnes, 4 ans
- ▶ 1500 KLOC, 2000 modules Erlang
- ▶ 99,9999999% dispo (31ms/an)
- ▶ 10 à 160 Gbit/s
- ▶ Leader marché (11%)

