



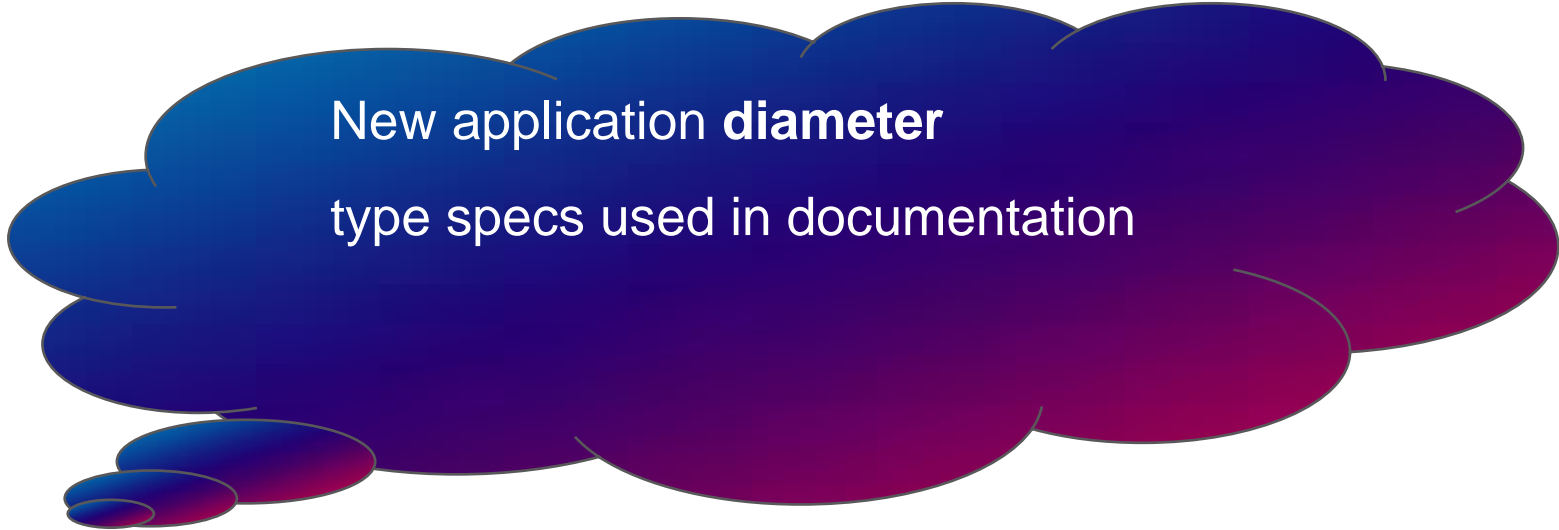
LATEST NEWS  
FROM  
THE ERLANG GROUP  
AT ERICSSON  
ERLANG FACTORY LONDON 2011



# HIGHLIGHTS IN R14B03

---

- › R14B03 released May 25:th
- › Mainly error corrections and smaller user contributions
- › But also some things worth highlighting like:



New application **diameter**  
type specs used in documentation

# DIAMETER

---

- › An implementation of the Diameter protocol as defined by RFC 3588. A successor to Radius.
- › Supports arbitrary Diameter applications by way of a **dictionary** interface that allows messages and AVP's to be defined and input into diameter as configuration.
- › Examples of Diameter applications:
  - Diameter Mobile IPv4 Application (MobileIP, [RFC 4004](#))
  - Diameter Extensible Authentication Protocol Application ([RFC 4072](#))
  - [Diameter Credit-Control Application](#) (DCCA, [RFC 4006](#))
  - Both the [HSS](#) and the [SLF](#) communicate using the Diameter protocol.
- › Support for all roles defined in the RFC: client, server and agent.
- › Plugin system for transport modules (SCTP and TCP provided as standard)

# AT LEAST 3 DIFFERENT TRACKS

---



# PRELIMINARY ROADMAP

---

R14B04 October 5:th



Mainly error corrections and user contributions

# PRELIMINARY ROADMAP CONTINUED

---

R15B December 14:th

Line number info in crash reports

64bit Windows version

SMP performance improvements

ASN.1 encode/decode performance

Inviso will be deprecated

Cleanup regarding typespecs

Support for parallel make

# PRELIMINARY ROADMAP CONTINUED

---

## Longer term

Improve performance with JIT-  
compilation (using HiPE/LLVM)

Experiments and EEP for new  
datatype Hashes

Native processes (maybe some steps  
already in R15B)

More multicore scalability  
improvements

# LINE NUMBER INFO

---

- › Generated as default by compiler
- › Loader puts info in internal tables used when an exception or call to `erlang:get_stacktrace` occurs
- › Size of .beam file increases with ~ 5 %

- › Size of loaded code increases with 10% in a 32 bit VM (less percentage in a 64 bit VM)
- › No extra cost in runtime.
- › New EEP describes in detail how it works.
- › EEP and code will be available on github soon.



# LINE NUMBER INFO CONTINUED

---

Introducing a small "harmless" incompatibility

A stack trace will change from the format:

```
[ {Module,Function,Arity}, ... ]
```

To the new format:

```
[ {Module,Function,Arity,LocationInfo}, ... ]
```

Where `LocationInfo` is a property list :

```
[ {file,FilenameString}, {line,LineNumber} ]
```

# LINE NUMBER INFO, DEMO

## example.erl

```
-module(example).  
  -export([m/1]).  
  -include("header.hrl").  
  
m(L) ->  
    {ok,lists:map(fun f/1, L)}. %Line 6
```

## header.hrl

```
f(X) ->  
    abs(X) + 1. %Line 2
```

```
Eshell V5.8.4 (abort with ^G)  
1> c(example).  
{ok,example}  
2> example:m([a]).  
** exception error: bad argument  
   in function abs/1  
   called as abs(a)  
   in call from example:f/1 (header.hrl, line 2)  
   in call from lists:map/2 (lists.erl, line 948)  
   in call from example:m/1 (example.erl, line 6)  
3>
```

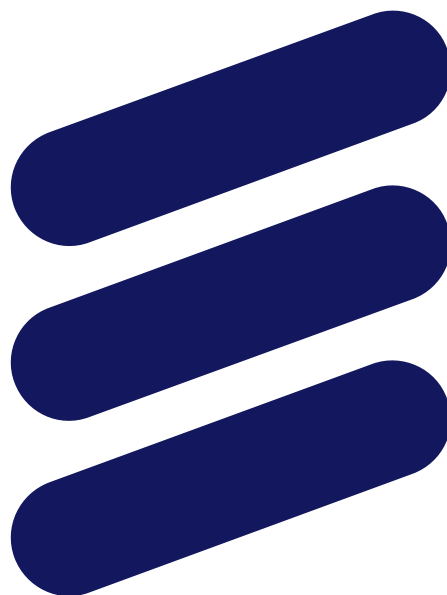
# LINE NUMBER INFO, DEMO

---

```
Eshell V5.8.4 (abort with ^G)
2> process_info(Pid,current_location).
{current_location,{gen_server,loop,6,
                  [{file,"gen_server.erl"},{line,331}]}}

3> process_info(Pid,current_stacktrace).
{current_stacktrace,[{gen_server,loop,6,
                    [{file,"gen_server.erl"},{line,331}]},
                    {proc_lib,init_p_do_apply,3,
                    [{file,"proc_lib.erl"},{line,184}]}]}

4>
```



**ERICSSON**