ERLANG

百科解释

1、Erl是话务量单位。话务量等于单位时间的呼叫次数与呼叫的平均占用时长的乘积。单位是ERL（爱尔兰）。 公式：A表示话源话务量，λ表示单位时间内发生的平均呼叫数，S表示呼叫的平均占用时长，根据话源话务量的定义，则A=λ&＃8226;S

2、Erlang是一种通用的面向并发的编程语言，它由瑞典电信设备制造商爱立信所辖的CS-Lab开发，目的是创造一种可以应对大规模并发活动的编程语言和运行环境。Erlang问世于1987年，经过十年的发展，于1998年发布开源版本。Erlang是运行于虚拟机的解释性语言，但是现在也包含有乌普萨拉大学高性能Erlang计划(HiPE)[1]开发的本地代码编译器，自R11B-4版本开始，Erlang也开始支持脚本式解释器。在编程范型上，Erlang属于多重范型编程语言，涵盖函数式、并发式及分布式。

开发及演变历史
Erlang得名于丹麦数学家及统计学家Agner Krarup Erlang，同时Erlang还可以表示Ericsson Language。
发行版本
1998年起，Erlang发布开源版本，采用修改过的Mozilla公共许可证协议进行发放，同时爱立信仍然提供商业版本的技术支持。目前，Erlang最大的商业用户是爱立信，其他知名用户有北电网络、Amazon.com以及T-Mobile等。
当前的语言特征
Fail-fast(中文译为速错),即尽可能快的暴露程序中的错误.
面向并发的编程(COP concurrency-oriented programming).
函数式编程
弱类型
脚本语言

函数式编程
Ering函数大致写法如下，以一个求整数阶乘的模块为例：

```
-module(fact).
-export([fac/1]).
fac(0) -> 1;
fac(N) when N > 0 -> N * fac(N-1).
```

下面是快速排序算法的一个Erlang实现：

```
%% quicksort:qsort(List)
%% Sort a list of items
-module(quicksort).
-export([qsort/1]).
qsort([]) -> [];
qsort([Pivot|Rest]) ->
qsort([ X || X <- Rest, X <= Pivot]) ++ [Pivot] ++ qsort([ Y || Y <- Rest, Y > Pivot]).
```

并发及分布式编程
代码示例如下：

```
% create process and call the function web:start_server(Port, MaxConnections)
ServerProcess = spawn (web, start_server, [Port, MaxConnections]),
% create a remote process and call the function web:start_server(Port, MaxConnections) on machine RemoteNode
RemoteProcess = spawn(RemoteNode, web, start_server, [Port, MaxConnections]),
% send the {pause, 10} message (a tuple with an atom "pause" and a number "10") to ServerProcess (asynchronously)
ServerProcess ! {pause, 10},
% receive messages sent to this process
receive
a_message -> do_something;
{data, DataContent} -> handle(DataContent);
{hello, Text} -> io:format("Got hello message: ~s", [Text]);
{goodbye, Text} -> io:format("Got goodbye message: ~s", [Text])
end.
```

- **通信词典解释**