
University of
Kent

Joe Armstrong

EXERT SYSTEM DEVELOPER, AND ONE OF THE CREATORS OF ERLANG AT ERICSSON

University of Kent

University of Kent

```
for(Max,Max,F) ->
    [F(Max)];
for(I,Max,F) ->
    [F(I)|for(I+1,Max,F)].

> for(1,5,fun(I) -> I*I end).
[1,4,9,16,25]
```

RPC

University of Kent

```
rpc(Pid, Request) ->
    Tag = erlang:make_ref(),
    Pid ! {self(), Tag, Request},
    receive
        {Tag, Response} ->
            Response
    end.
```

RPC

University of Kent

```
rpc(Pid, Request) ->
    Tag = erlang:make_ref(),
    Pid ! {self(), Tag, Request},
    receive
        {Tag, Response} ->
            Response
        after <Time> ->
                …
    end.
```

RPC split in two

University of Kent

```
rpc(Pid, Request) ->
    Tag = erlang:make_ref(),
    Pid ! {self(), Tag, Request},
    Tag.

wait_response(Tag) ->
        receive
            {Tag, Response} ->
                Response
        end.
```

Futures

University of Kent

```
promise(Pid, Request) ->
    Tag = erlang:make_ref(),
    Pid ! {self(), Tag, Request},
    Tag.

yield(Tag) ->
        receive
            {Tag, Response} ->
                Response
        end.
```

University of Kent

```
Tag = promise(Pid, fun() -> … end),
… do some computations …
Val = yield(Tag)
```

University of Kent

```
par begin
    F1,
    F2,
    F3
par end
```

pmap

University of Kent

```
function pmap(L) ->
    S = self(),
    Pids = [do(S,F) || F <- L],
    [receive {Pid,Val} -> Val end || Pid <- Pids].

do(Parent, F) ->
    spawn(fun() ->
            Parent ! {self(), F()}
          end).
```


University of
Kent