

Building GSEs with Erlang

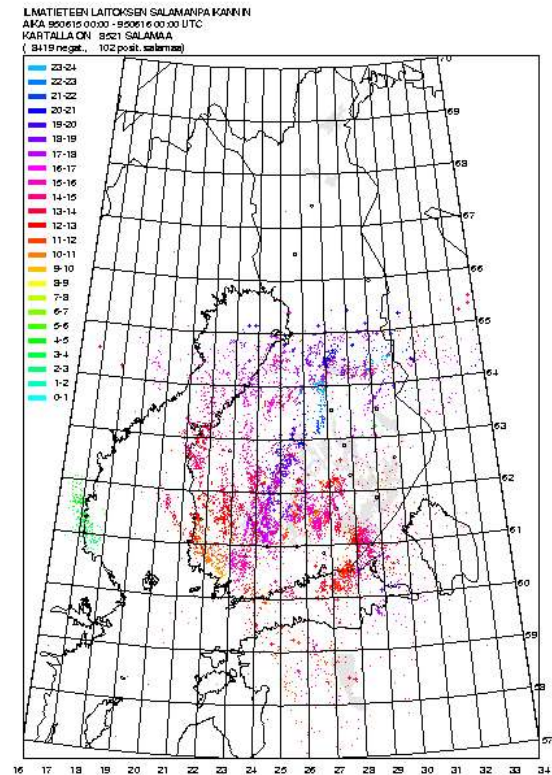
Jouni Rynö, FMI / Space research

- history of Erlang at Finnish Meteorological Institute (FMI)
- CIDA and COSIMA instruments
- Ground Support Equipment (GSE)
 - real-time telecommands (TC) and telemetry (TM)
 - offline telemetry (DB)
 - online telemetry (WWW)



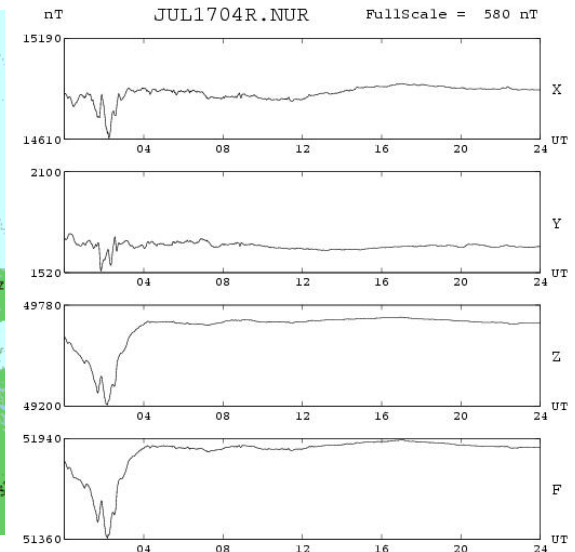
History of Erlang at FMI

- a small world
 - a colleague had been a colleague of Joe in 1980's
 - and the colleague's brother was working at Ericsson
- a broken lightning location system in June 1995
 - from learning Erlang to a realtime graphical lightning display in 3 weeks...



History of Erlang at FMI

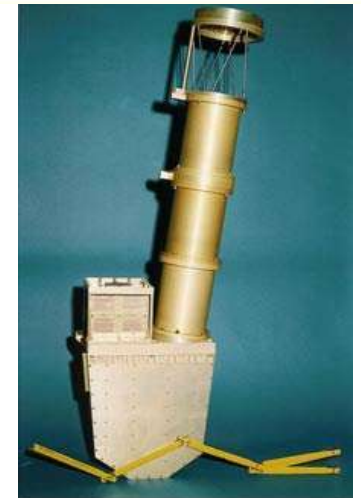
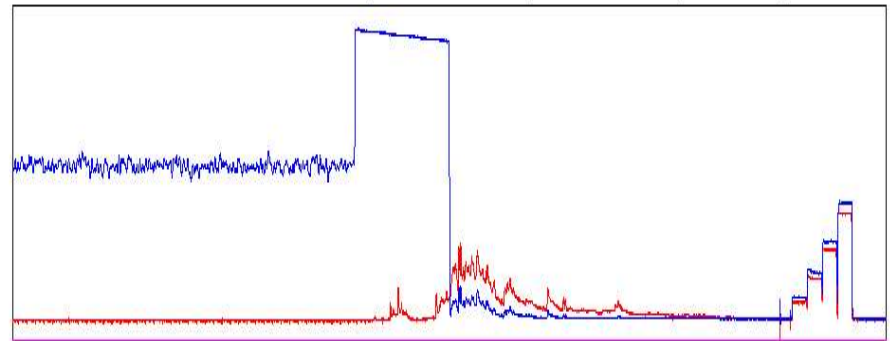
- magnetometer data acquisition systems
 - measures magnetic variation of the Earth
 - 3 components (north, east, vertical)
 - means with 1s, 10s and 60s time resolution



Comet and Interstellar Dust Analyzer (CIDA)

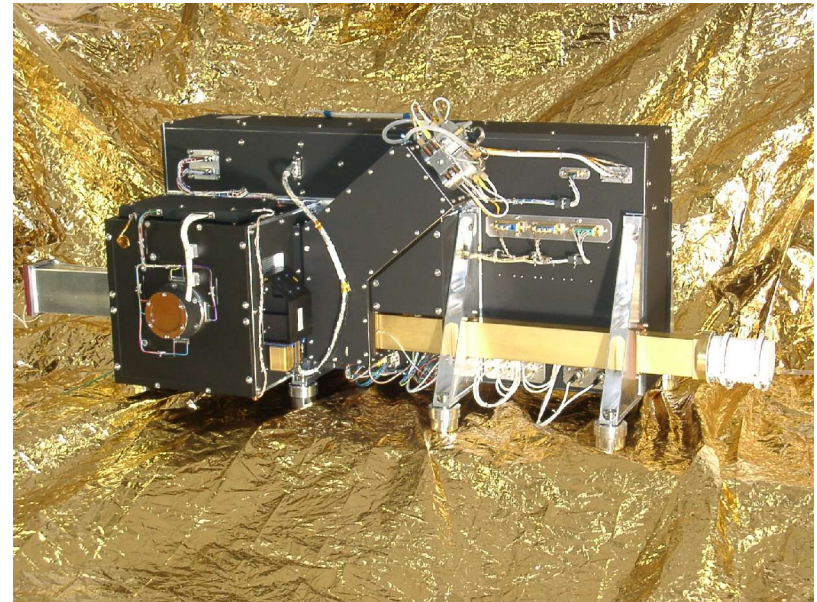
- time of flight mass spectrometer
- made a comet flyby on 02.01.2004 on the Stardust-spacecraft
- development 1996 - 1998

"POSITIVE "81P/WILD 2 (1978 A2)" large

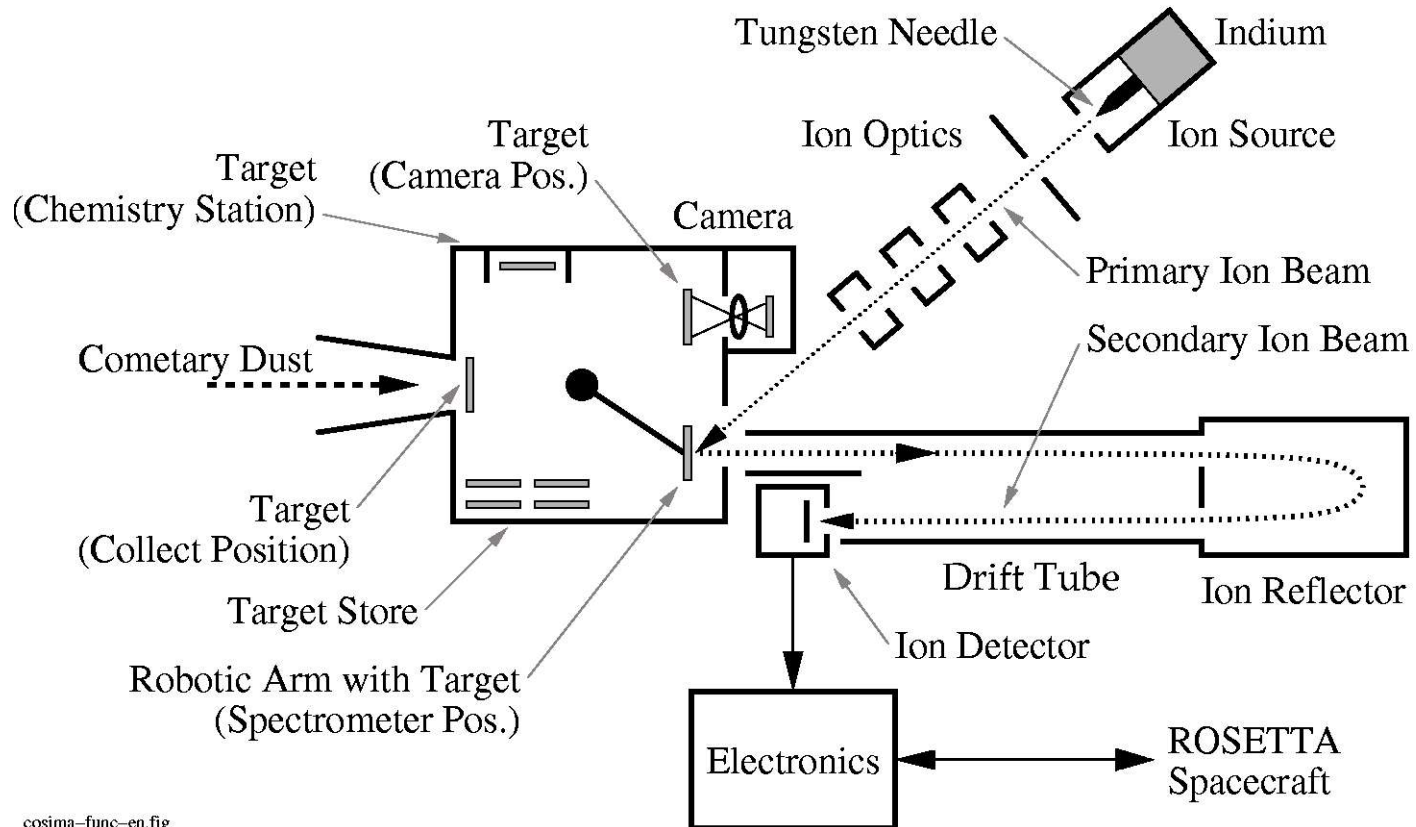


Cometary Secondary Ion Mass Analyzer (COSIMA)

- mass spectrometer
- onboard Rosetta, ESA's spacecraft to study the comet 67 P/Churyumov-Gerasimenko
- measurements 2014 - 2015



Cometary Secondary Ion Mass Analyzer (COSIMA)



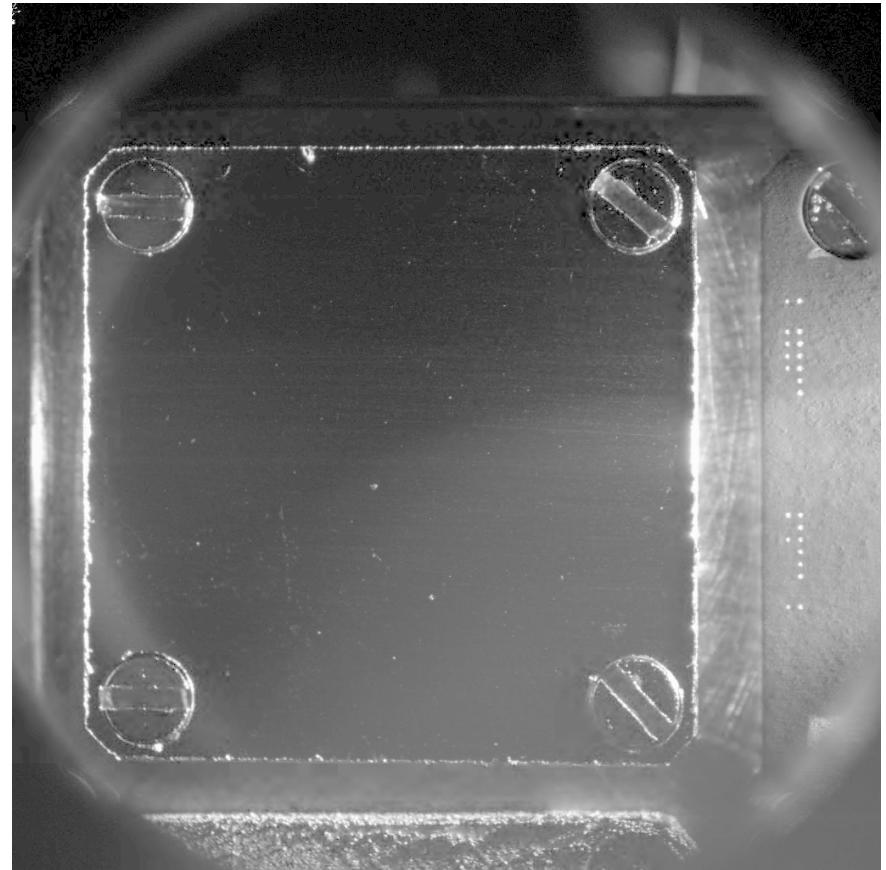
cosima-func-en.fig



ILMATIETEEN LAITOS
METEOROLOGISKA INSTITUTET
FINNISH METEOROLOGICAL INSTITUTE

Cometary Secondary Ion Mass Analyzer (COSIMA)

- 72 dust targets, each 10*10 mm size
- camera to detect 10-100 μm size particles
- motor system to move targets around with 4 μm accuracy
- 80 μm In ion beam



Ground Support Equipment

- simulates S/C electrical interface during the development
- provides telecommand generation and telemetry analysis
- as in the Erlang book!



User interface

Main CIDA instrument operation Check out View

MISSION VARIABLES

SC sim. parameters

Spacecraft: Stardust
 Mission: comet encounter
 Encounter start offset (in seconds): -700
 SW source: the spacecraft Choose SW file
 S/C sends messages every 100 ms
SC can't take data every 10 . time

CIDA parameters

Use simulator
 Particle flux: constant (particles/s) 1 Start 2004 213 00:00:00
 CIDA port: simulation Pick TM file Stop 2004 214 00:00:00

SIMULATION STATUS

No simulation

HOUSEKEEPING VALUES

+5V:	N/A	V	TARGET:	N/A	Å° C
-5V:	N/A	V	MM-1:	N/A	Å° C
+15V:	N/A	V	SCIF:	N/A	Å° C
+5V CPU:	N/A	V	CPU:	N/A	Å° C
+15V HVC:	N/A	V	LVC:	N/A	Å° C
-15V HVC:	N/A	V	HK:	N/A	Å° C
I(+5V):	N/A	A	FDAQ_P:	N/A	Å° C
I(-5V):	N/A	A	FDAQ_A:	N/A	Å° C
I(+15V):	N/A	A	SpareD:	N/A	

ADC status: N/A
 HVSAFE: N/A
 TEST: N/A
 HV power: N/A
 RFB enable: N/A
 MM1 enable: N/A

Target: N/A
 Autocal: N/A
 Protection: N/A

EDF no: N/A
 EDF size:
 EDF time:
 EDF encounter:
 Events: N/A
 Event rate: N/A

Save hk values into file

MAIN LOG

Last EDF(s)

Spectrum Speed Charge
 None km/s fC Log

Detailed panel plot Detailed plot Bell Multiple

CIDA OPERATION

High voltage control

HV on HV off
 Current value: N/A

Ion mode control

Positive Negative
 Current value: N/A

Target size control

Small Large
 Current value: N/A

MM1 HV control

HV (kV) Dec
 N/A
 Set new HV value

EDF timeout

Timeout min.

FDAQ calibration

Manual Staircase Baselines

Offset/trigger configuration

Primary Hi 1 Lo 1
 N/A N/A N/A
 Straight N/A N/A Trigger pulse
 Delayed N/A N/A integral
 N/A N/A N/A
 Configure

CIDA



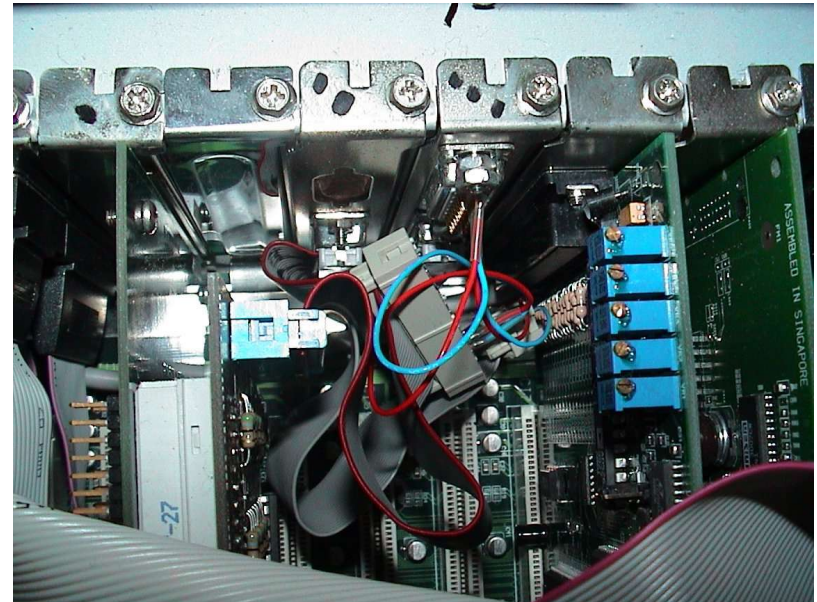
User interface 2

- user interface written in Tcl/Tk
 - button/menu interface fine for simple and often used interactive operations
 - complicated commanding needs a scripting language
 - BLT gives nice 2D graphs
- Erlang talks with Tcl via modified ewish



H/W interface

- selfmade PCI-card for S/C interface (linux driver with IDL c-server)
- power supply
- analog to digital PCI-card for current, temperature and vacuum monitoring



Why not Erlang

- 1996:
 - no binary syntax
 - no hex format
 - not open source
 - not sure about language future
- 2004:
 - few know about the language



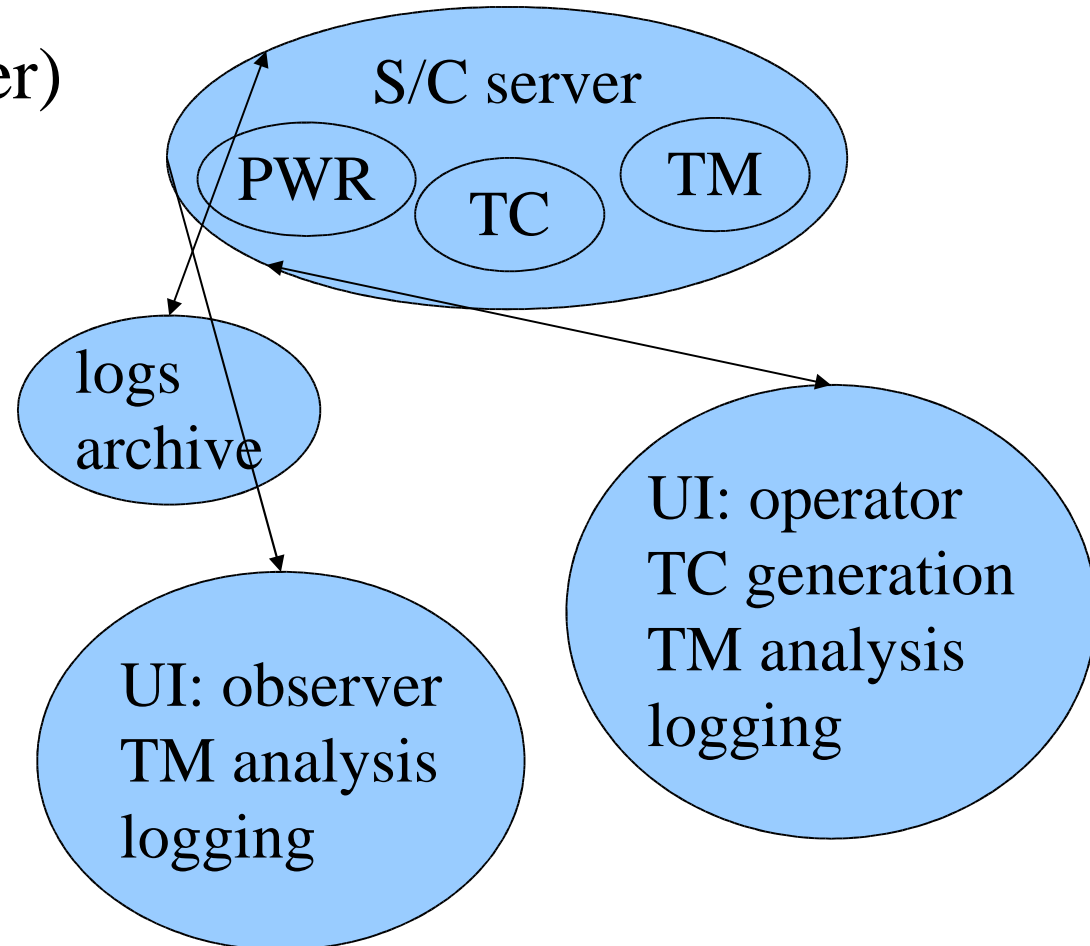
Why with Erlang

- experience with Occam language (in the Transputer-processor)
 - processes, synchronous messages, no pointers
- Erlang:
 - buffering comes for free! (asynch ...)
- new languages are fun to learn



Erlang as middleware

- TC timing (buffer)
- TM buffering
- logging
- HK monitoring
- archiving
- distribution!



TC with Erlang

- formatting and checking
- timing
- buffering
- acceptance and execution
- acknowledgements

```
handle_event({time_out, TC, Timers, TimerKey}, State) ->
  Result = gen_server:call(State#state.owner,
                           {write, TC}, 15000),
  {ok, Resp_timer} = timer:apply_after(?TIMEOUT,
                                       'gen_server', 'cast', [sc_communication_server,
                                                            {no_response, {TimerKey}}]),
  ets:insert(Timers, {TimerKey, Resp_timer}),
  NewState = State#state{queue = tl(State#state.queue)},
  queue(NewState#state.queue),
  {ok, NewState};
```



TM handling

```
split_packets(TMblocks, Length) ->
  if
    Length == 0 ->
      {ok, 0};
    Length < 0 ->
      {error, negative};
    Length < 6 ->
      {error, short};
    size(TMblocks) >= Length ->
      case TMblocks of
        <<0:3,Type:1,Header:1,?COSIMA_PID:7,Category:4,Segment:16,TMLength:16,Rest/binary>> ->
          PacketLength = TMLength + 7,
          if
            PacketLength > size(TMblocks) ->
              comm_log_server:write("TM error: packet block ~w <> length ~w~n", [size(TMblocks), PacketLength]),
              {error, invalid};
            true ->
              {TMpacket, RestOfTM} = split_binary(TMblocks, PacketLength),
              comm_log_server:write("TM cat ~w "size is ~w = ~w~n", [Category, size(TMpacket), PacketLength]),
              gen_server:cast(sc_communication_server, {tm_packet, TMpacket, true}),
              split_packets(RestOfTM, Length-size(TMpacket))
          end;
      end;
```



TM handling, parameters

```
extract_binary(Size, HeadSize, TailSize, Type, Bin) ->
  case Type of
    "unsigned-integer" ->
      <<H:HeadSize, Value:Size/unsigned-integer, T:TailSize, Rest/binary>> = Bin;
    "signed-integer" ->
      <<H:HeadSize, Value:Size/signed-integer, T:TailSize, Rest/binary>> = Bin;
    "float" ->
      <<H:HeadSize, Value:Size/float, T:TailSize, Rest/binary>> = Bin;
    "time" ->
      <<H:HeadSize, Value:Size/unsigned-integer, T:TailSize, Rest/binary>> = Bin
  end,
  <<UH:HeadSize, UValue:Size/unsigned-integer, UT:TailSize, URest/binary>> = Bin,

  {Value, UValue}.
```



Offline TM -> DB

- -record(time_parameter,
 { time, parameter, value }).
- on ground, every 2 seconds
 - ~30 16-bit AD values
 - ~64 single bit status flags
- plus images, spectra
- about 200 Mbytes / working day



Mnesia

- 200 Mbyte / day -> disk_only_copies
- for faster searching, use fragmented tables with a special hash module
 - key is MJD2000, decimal day since 01.01.2000
 - key_to_frag_number(State, Key) when record (State, hash_state) ->
$$\text{IntKey} = \text{trunc}(\text{Key}).$$



Online access -> yaws

- DB frontend for the scientists
 - housekeeping data
 - events, images, spectra
 - target history
 - instrument status
- document repository
- wiki



Yaws

RSDB event (for multiple selection, use shift for a range, control for individual parameters)

```
YCS41666 COSIMA Startup 41666
YCS41665 COSIMA SW error 41665
YCS41661 COSIMA shutdown 41661
YCS41660 COSIMA about to reboot 41660
YCS41655 Cosima shutdown request 41655
YCS41653 Initiate Context request 41653
YCS41652 COSIMA needs SW 41652
YCS41651 COSIMA boot 41651
YCS41612 TOF HV switched off 41612
YCS41611 TOF HV switched on 41611
YCS41610 TOF to be switched on 41610
YCS41602 PIBS HV SWITCH OFF 41602
YCS41601 PIBS HV SWITCH ON 41601
YCS41600 PIBS HV TO BE SWITCHED ON 41600
YCS41586 COSISCOPE COMM-STATUS 41586
YCS41585 COSISCOPE ERROR 41585
YCS41583 COSISCOPE OP COMPLETED 41583
YCS41582 COSISCOPE SWITCH OFF 41582
YCS41581 COSISCOPE SWITCH ON 41581
YCS41580 COSISCOPE to be switched on 41580
YCS41574 PTS ignition failure 41574
```

Start date (YYYY-MM-DDTHH:MM:SS):

Stop date (YYYY-MM-DDTHH:MM:SS):

For start date, the hour defaults to 00, minute to 00 and seconds to 00

For stop date, the hour defaults to 23, minute to 59 and seconds to 59



ILMATIETEEN LAITOS
METEOROLOGISKA INSTITUTET
FINNISH METEOROLOGICAL INSTITUTE

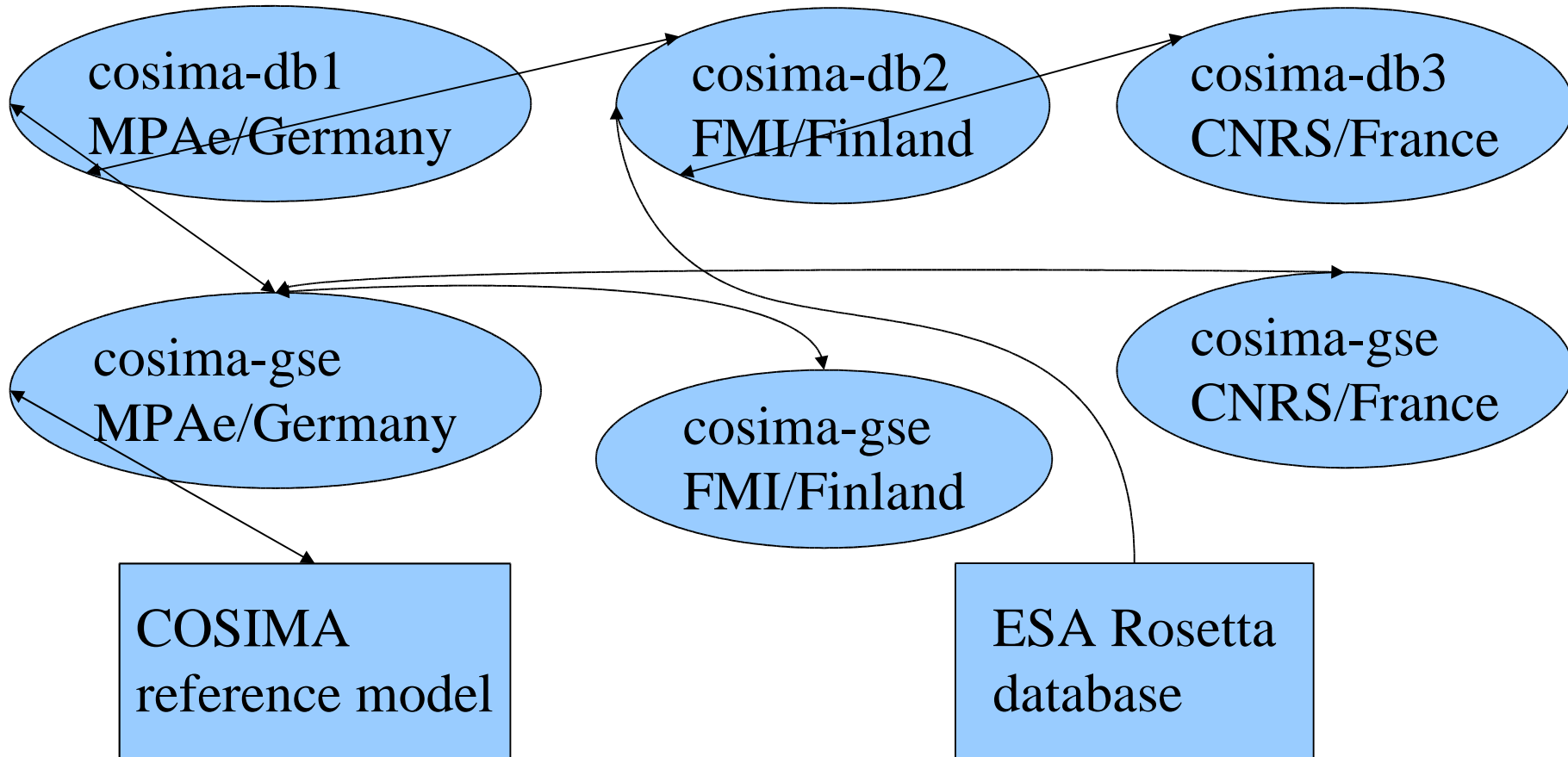
Yaws

- define queries to the DB

```
Selection = [{ {time_parameter,'$1','$2','$3'},  
  [{ '<', {const,1608.00}, '$1'},  
    {'=<', '$1', {const,1609.00}},  
    {'==', '$2', {const,"NCSA1100"}},  
    {'orelse', {'==', {const,4.15650e+4}, '$3'},  
              {'==', {const,4.16000e+4}, '$3'},  
              {'==', {const,4.16010e+4}, '$3'}}}],  
['$ ']]
```



2005: COSIMA net nodes



Future

- EGSE for the BepiColombo/SERENA (ESA Mercury mission)
- the whole meteorological section could benefit from Erlang
 - lots of different weather reports in different formats coming in daily
 - different, tailored services to public
- needs some selling

