

# Learning Erlang and developing a SIP server/stack with 30k potential users

Fredrik Thulin <sip:ft@it.su.se>

Enheten för IT och media

Stockholms universitet

EUC2004

# Stockholm university telephony in 1997 :

- Ericsson MD110
- 5000 subscribers

## 2000 :

- Ericsson MD110 – 4200 subscribers
- Cisco CallManager – 800 subscribers
- Early VoIP adopters, but still no open standard

# SIP

- Session Initiation Protocol
- IETF proposed standard (RFC3261, 2, 3, 4 and 5)
- Does not care about audio/video/whatever
- Instant messaging and presence
- Parsing is hard, even though (?) it's plain text
- Transactions are complex
- Few open source implementations as to date

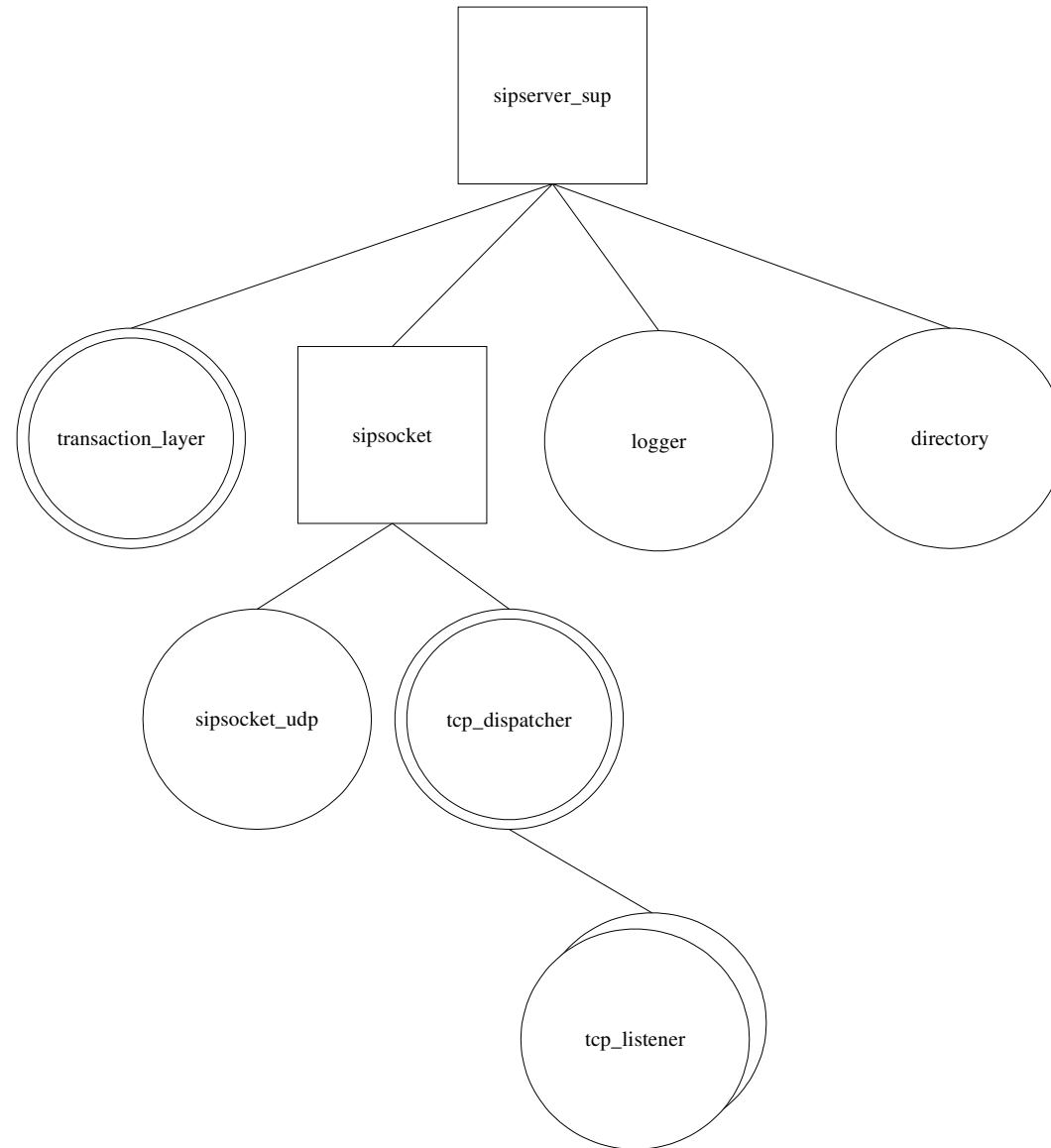
# Personal Erlang experience

- No prior knowledge about functional programming
- Hard time understanding some syntax (strings are lists,  $[H | T] = \text{“string”}$ )
- Assign once
- Really like cheap processes and the IPC

# Implementation

- Magnus Ahltop, KTH made a couple of implementations in Perl, Python, C, ...
- Erlang implementation was the most viable one
- First Yxa snapshot released 2003-10-07
- OTP supervisor model
- 15000 lines of code, 5000 lines of comments

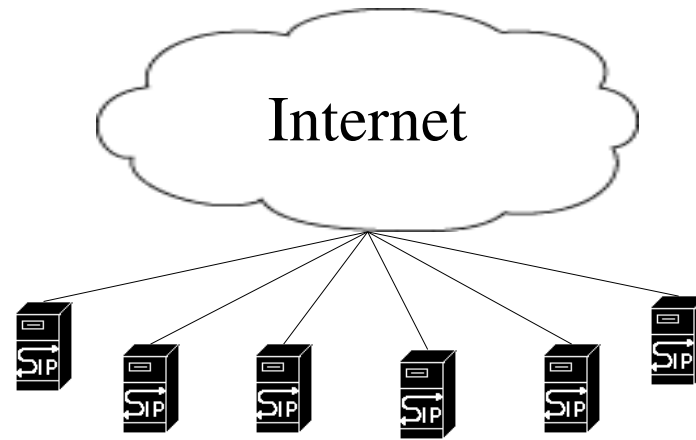
# OTP supervisor model



# Plans

- Build distributed SIP servers
  - for routing
  - for students (free VoIP/SIP-service, no PSTN)
  - for basic call service? Evaluating.
- Distributed policy control (perimeter defense - rate limiting etc.)
- External event logging with call context
- RFC compliant stack, easy to write applications

# Perimeter defense



Servers running Yxa incomingproxy

- Rate limiting number of ongoing calls per host, class C network, domain, SSL certificate etc.



# Project

- <http://www.stacken.kth.se/projekt/yxa/>