Curriculum Vitae

Name: Mats Cronqvist
Phone (mobile): +46 72 7285755
E-mail: mats@cronqvi.st
URL: http://massema.net
Based in: Stockholm, Sweden (GMT +1)
Languages: Fluent in Swedish and English, basic German
Passport: EU (Swedish)
Education: Ph.D. in Physics
Professional Programming Experience: 32 years

Education

1992: Ph.D. in Nuclear Physics from the University of Goteborg, Sweden

1986: B.Sc. in Physics from the University of Goteborg, Sweden

The subject of my Ph.D. thesis was: "Reaction Mechanisms in Intermediate Energy Heavy-Ion Collisions"

Skills

Cloud:

kubernetes, terraform, AWS stuff (primarily SQS, S3, EC2).

Operating Systems:

Mostly forgotten: VMS, irix, solaris, hp-ux, ultrix.
In daily use: linux (debian, centos), osx.

Programming languages:

Mostly forgotten: BASIC, FORTRAN, pascal, Julia, various CERN languages.
Up to date: C, javascript, SQL.
In daily use: Erlang, bash.

Version Control Systems:

Mostly forgotten: cvs, clearcase, subversion.
In daily use: git.

Architectural Belief System:

Decoupling, Distribution, Asynchronous Message Passing.

Always Installed:

emacs >= 25, docker.

Experience

2018-present: Senior Developer at Working Group Two, Sweden.

Startup doing telephony in the public cloud. My main focus was designing and implementing a gateway between telecommunication protocols (mainly SS7 and DIAMETER) and cloud technologies (mainly

GRPC and kafka). For redundancy and scalability, the system has a two-tiered, distributed, design, with a small part managing the external-facing sockets, backed by an arbitrary number of workers. Also created a system for indexing content of terabytes of packet captures (using wireshark, awk, and AWS Athena).

2017-2018: Senior Technical Director of 247.ai, Sweden.

Managed ~15 developers. Architecture, mostly regarding AWS services; S3, SQS, Glue, Presto. Data analysis using Julia.

2016-2017: Architect at Campanja, Sweden

Developing and architecting in a Campanja's system, mainly Erlang. Microservices running under kubernetes in AWS.

2014-2016: Senior Architect, at Klarna, Sweden

Protected the corporate revenue streams. My team made sure the cash cow system stayed wella-functioning. We paid off technical debt, mainly through overload testing with intense profiling, judicious refactoring, and introducing sensible deployment and testing pipelines.

2012-2013: Chief Architect, at Klarna, Sweden

Responsible for the migration of the Klarna legacy business system from a monolithic Erlang application to system of loosely coupled services. Technically successful, but the company decided to go down the Enterprise Java route.

2011: Manager, Core Development, at Klarna, Sweden

Managed a team of ~10 developers (mostly Erlang). Main task was to migrate functionality from the legacy system to isolated services.

2010: Manager, Live Operations, at Klarna, Sweden

Managed the team that operated the Klarna business system. The teams tasks included system administration (Debian), application monitoring, upgrades of hardware and software.

2007-2009: Senior Developer at Klarna (a.k.a. Kreditor), Sweden

Implemented various parts of the Klarna business system (written in Erlang). Profiling and debugging.

2003-2007: System Expert at Ericsson Telecom, Hungary

Implemented a suite of profiling and troubleshooting tools for Erlang applications. Enabled us to identify numerous bottlenecks, significantly improving performance. In order to write the GUI for the profiler I invented and implemented a GTK binding for Erlang. The C-side is a daemon implementing the Erlang distribution protocol. Most of the C code is generated from the GTK header files.

1997-2003: System expert at Ericsson Telecom, Sweden

Implemented the first distributed version of our product. Allowed us to scale out the application on multiple CPUs. Invented the first serious Erlang troubleshooting tools. Still in heavy in-house use.

1993-1997: Post Doctoral Researcher at Lawrence Berkeley Laboratory, CA, USA

Wrote major pieces of the software for the E896 experiment at Brookhaven National Lab, including the on-line monitoring system, the event visualization, and most of the track reconstruction. Wrote most of the statistical analysis code for the TRANSPORT experiment at Lawrence Berkeley Lab.

1986-1992: Graduate school at the University of Goteborg, Sweden

Wrote many pieces of code for several experiments, mostly related to data analysis and visualization. Awarded a grant by the Sweden-America Foundation to spend a year at Michigan State University, MI, USA. Awarded a grant to develop a simple data analysis program for the Mac to be used in undergrad teaching lab. It was still in use when I graduated.

1986: Software contractor

Hired over summer break by the Nuclear Physics group at the University to develop a data visualization program for nuclear physics experiments. Hugely impressed everyone since it used 2-D graphics.

Open Source Programs

At github.com/massemanet

```
gtknode - an Erlang GTK binding
inotify - an Erlang binding to the Linux inotify API
redbug - a tracing debugger for Erlang
```

At github.com/marijnh/CodeMirror/tree/master/mode/erlang

```
An erlang mode for CodeMirror (a code editor javascript component)
```

Maintains distel (Erlang-Emacs interface)