

# Adam Bergmark

## Curriculum Vitae

### Contact information

E-mail address: adam@bergmark.nl

Skype username: adam\_bergmark

### Introduction

Born June 18th 1986 in Skellefteå, Sweden.

I'm a computer scientist who has been programming since around 1998, starting out with QBasic and web related languages such as JavaScript and PHP.

My main interest the last 10 years has been statically typed functional programming, mostly in Haskell but also in Scala, PureScript, OCaml, and TypeScript.

Some of my other interests within computer science are (microservice) application APIs, library design, programming language theory, compilers, computability, free software, category theory, dependent types, and data type generic programming.

### Education

*Masters program in Computer Science, 2013*

Linköping Institute of Technology

Linköping, Sweden

Orientation: Software engineering, Theoretical datalogy

*Secondary school, 2005*

Holavedsgymnasiet

Tranås, Sweden

Orientation: Technnology, Data/IT

### Work experience

References are available on request.

2017–(ongoing) Senior Software Engineer at Magine TV (Scala)  
Stockholm, Sweden

- Part of a team maintaining 60 scala micro services
- Responsible for video playback
- Certified for Google Widevine DRM (CWIP)
- Desgined an API for secure 3rd party access to parts of our platform
- Built a system to allow customizable product offerings
- Unified the library infrastructure to simplify maintenance of existing services and speed up new feature development.
- Wrote tooling for automated testing of API documentation.
- Invented and implemented guidelines to simplify changes made to service APIs (for HTTP and Kafka)

2016–2017 Software Engineer at Klarna (Scala)  
Stockholm, Sweden

- Designed API guidelines for the payments domain
- Lead designer for a unified Direct Debit REST API
- Holding monthly workshops teaching functional programming
- Wrote a Scala library for type-safe conversions
- Wrote a DSL for compile-time derivation of Swagger schemas for HTTP APIs
- Worked on unifying the build system of our microservices to SBT and simplifying dependency management
- Redesigned our continuous delivery pipelines in Jenkins
- Designed and participated in interviews for consultants

2012–2016 Software Engineer at Silk (Haskell & TypeScript)  
Amsterdam, The Netherlands

- Authored the initial version of Silk’s embeddable visualizations
- Designed and implemented Silk’s server side CSV importer including its public API  
API spec: <https://github.com/silkapp/silk-api-client/wiki/Silk-Import-API>
- Wrote a bidirectional routing library based on the DOM History API
- Migrated our PostgreSQL interfacing Haskell code to the type safe SQL-generating DSL “Opaleye”  
by writing a general purpose framework  
Source code: <https://github.com/silkapp/girella>
- Wrote a Haskell test-suite/framework for black box testing our public API
- Major contributor to a client side multipart DSL à la Haxl (<https://github.com/facebook/Haxl>)
- Substantial work on our open source packages  
Repositories: <https://github.com/silkapp?type=source>
- Wrote a microservice responsible for semantic diffing of our XML document format
- Re-wrote our e-mail notification service to aggregate e-mail updates to users

2007–2012 Software development freelancer  
Linköping, Sweden

- Wrote a general purpose GUI based markdown editor in JavaScript
- Created a helicopter pilot schedule tracker for the swedish coast guard

2010-2011 Laboratory assistant at Linköping Institute of Technology  
Linköping, Sweden

- Web Programming (Introduction to HTML, CSS, JavaScript, and PHP)
- Object-Oriented Programming (Using Java, C++, and Python)

2008–2009 Web developer, Java and front-end at Dimest AB  
Stockholm, Sweden

- Head of client side development

2005–2008 Web development freelancer

I created several websites from scratch using PHP and JavaScript.

## Other merits

I'm currently active in Haskell community where I:

- Maintain "aeson", the de-facto library for interfacing with JSON.  
<https://hackage.haskell.org/package/aeson>
- Co-authored the Haskell-to-JavaScript compiler "Fay"  
<https://github.com/faylang/fay/wiki>
- Co-authored the Hackage Matrix Builder  
A build bot for checking package compatibility across dependency and compiler versions  
<https://matrix.hackage.haskell.org/>
- Maintain a number of other packages  
<https://hackage.haskell.org/user/AdamBergmark>
- Curate the hackage package database as a hackage trustee  
[https://wiki.haskell.org/Hackage\\_trustees](https://wiki.haskell.org/Hackage_trustees)
- Curate the Stackage package collections  
<http://www.stackage.org/>

I've held presentations and workshops mostly focusing on functional programming. Here are the longer ones (at least 30 minutes)

- Haskell eXchange 2013 - Compiling Haskell to Javascript with Fay
- Ghent Functional Programming Group 2013 - Compiling Haskell to JavaScript
- Significant Bit 2013 - An introduction to functional programming with Haskell
- Functional Programming Amsterdam (FP AMS) 2014 - JavaScript as a Compilation Target
- FP AMS 2014 - Learn you a Haskell Workshops

Up until 2011 I authored a general purpose JavaScript framework "Cactus" which I used in various projects. It's about 10 000 lines of code. It leverages many design and architectural patterns such as Template View, Active Record, and Key Value Observing.

<https://github.com/bergmark/Cactus>

Profiles on related websites:

- GitHub: <https://github.com/bergmark/>
- Stack Overflow: <https://stackoverflow.com/users/182603/adam-bergmark>

## Natural Language Skills

### Proficient

English, Swedish (First language)

### Beginner

Dutch, German

## Technical skills

### Proficient

Programming languages: Haskell, Scala, TypeScript, Java, JavaScript, Ruby  
Frameworks/libraries: Finagle/Finch, Opaleye, React, Servant, Slick, Servant  
Databases/storage/messaging: PostgreSQL, Cassandra, S3, Kafka, MySQL  
IDE:s and text editing: Emacs, IntelliJ IDEA  
Version control: git, Mercurial, Subversion  
Operating systems: Mac OS X, GNU/Linux  
Build systems: Cabal/Stack, sbt  
Continuous Integration: Jenkins, Travis CI  
Deployment: AWS

### Familiar

Programming languages: Assembler, C, Common Lisp, Emacs Lisp, Idris, OCaml, Python, Rust, Scheme  
Operating systems: Windows

## Miscellaneous

Citizenship: Swedish

Driver's Licence: Yes, Swedish

November 30, 2019