



## Joachim Bengtsson

*Code Poet*

**Born** March 1985  
**Online** <http://thirdcog.eu/>  
**Blog** <http://nevyn.tumblr.com/>  
**Email** [joachimb@gmail.com](mailto:joachimb@gmail.com)  
**Phone** +46 (0)70 600 59 43  
**Mail** Kronborgsgränd 10, 3tr  
 16446 Kista, Sweden  
**Twitter** @nevyn

## Work

<b>Ongoing</b> ^ <b>2014</b>	<b>Lookback</b> <b>Founder, CTO, Code Poet</b> Lookback was my hobby project with the ambition to make UX research and QA testing easier at Spotify, that then transformed into a serious product when Jonatan and Carl Littke joined me as cofounders. I do iOS development, support, and architecting our technical infrastructure.
<b>2013</b> ^ <b>2009</b>	<b>Spotify</b> <b>System Owner &amp; Software Engineer, iOS</b> <a href="#">Writing the world's first Spotify client for iOS</a> ( <i>Spot</i> ) convinced Spotify to hire me. I took over their iOS and Mac clients, maintaining them for a few years, slowly switching focus to iOS only. My greatest accomplishment during this period has been helping design and leading development of Spotify's iPad app, focusing on making its navigation effortless and its UI delicious.
<b>2009</b> ^ <b>2007</b>	<b>Blekinge Institute of Technology</b> <b>Project Assistant</b> Project assistant at BTH's research labs. Spent six months as software archaeologist, restoring various old Java simulators. Rest of the time was spent building a distributed multiagent system in Erlang, with a Quartz Composer-like editor in Cocoa, and visualizer in Cocoa + OpenGL, together with three friends and colleagues.
<b>2007</b> ^ <b>2003</b>	<b>Freelancing</b> I wrote networked touch screen kiosks for AB Spelguiden (Ruby on Rails backend and Cocoa frontend), and various smaller jobs (mostly PHP) on the side during my education.

## Education

2008 ^ 2004	<b>Blekinge Institute of Technology (BTH)</b> <b>Undergraduate Program: Game Programming (Unfinished)</b> Three years of computer science with emphasis on performance and 3D. I specialized on network programming, audio programming, and game engine design. My candidate thesis is called ' <i>Real-time acoustics modeling in games</i> '; it was never finished, and thus I don't have a degree.
2004 ^ 2001	<b>Söderportgymnasiet, Kristianstad</b> <b>International Baccalaureate Diploma Programme</b> IB DP is an international secondary education/university preparatory program in English. I chose to study economy, chemistry and physics, in addition to the compulsory languages and math. I excelled (7 points) in physics and English.

## Developer Profile

I love Objective-C beyond words. I can't write a game without making it networked. I will experiment and try to find unorthodox solutions, with the end-goal always being succinctness (which means not only 'brief', but also 'easily understandable').

I have too many open source and Github projects to enumerate, but a few interesting ones (in roughly chronological order, newest first):

- [SPAsync](#): Some experimental async primitives for ObjC, with much inspiration from C#.
- [SPSuccinct](#): Sane, object-oriented APIs for KVO, and also less sane functional APIs for KVO. Used all over the place at Spotify.
- [TCAsyncHashProtocol](#): Send dictionaries over a TCP AsyncSocket. Aims for fast prototyping and succinctness. Built many games on top of this, love it.
- [Deathroid3](#): Deathmatch multiplayer clone of Super Metroid, in Objective-C (with three others)
- [NuRemoting](#): programming.nu over a socket. Introspect and interact wirelessly with your iOS apps. Used all over the place at Spotify as infrastructure for many hacks.
- [cfxr](#): Proper cocoa Mac port of the SDL app "sfxr" to make retro game sounds.
- [pwcblocks](#): The defacto unofficial guide to blocks.
- [Overload](#) ([source](#)): Abstract turn-based strategy game (actually released and on AppStore).
- [Rymdvarelser Mot Soldater](#): 30 week project during the Game Programming programme. Wrote an FPS engine, where I did engine design and a 3d audio engine from scratch (for my thesis).
- [LuaPlayer](#): Joined Frank Buß in making a Sony PSP homebrew app that would let beginners write their own PSP games in Lua. It was very, very rewarding to watch people learn to program using our tool.

Beyond ObjC, I've worked a lot with C, C++, Erlang and Ruby.

