Johnny Bock Andersen CV for Software Consultancy [1]

Summary and Introduction

The consultant specializes in development and design of advanced and high-performance software. In the areas of compiler, programming language and computer vision and graphics technologies, he has particularly strong skills that he uses in his company, Hardcore Processing, to turn cutting edge technologies into real-life applications. In doing this, he has also gained considerable experience in implementing numerical algorithms where numerical stability is often a challenge. He has worked both as an employee and even more so as a consultant for several companies over the years using a very broad spectrum of platforms, tools and languages with many of which he has thorough in-depth experience, which he often gains relatively quickly. He is usually in a good mood, well-liked among colleagues and gladly answers questions. He performs best when focusing on larger tasks alone. From his 32 years of experience, out of which 22 are professional, he has a very strong intuition about what the best, or at least near-optimal, solution is to many complex problems, even when it would be very hard and time-consuming to find the optimal solution by thorough analysis. He has worked more than full time over many of the years, which is hard to reflect with this style of CV where the workload is not fully specified.

Year of Birth : 1975	Citizenship : Danish (Denmark)	Current residence : Copenhagen, Denmark

Education

Year	Education	Place
2004-2010	M.Sc. Computer Science *The last 102.5 ECTS points were passed in 1.5 years, i.e. faster than normal, while partly working as a consultant*	University of Copenhagen, Denmark
1998-2004	B.Sc. Computer Science and Mathematics	University of Copenhagen, Denmark
2002-2003	Exchange Student in Mathematics	University of Athens, Greece

Human Languages

Language	Skill level
English	Read, write, understand, speak *fluently*
Danish	Read, write, understand, speak *fluently* - my mother tongue
Northern Jutish (Danish dialect)	Understand, speak *fluently* (not really a written language)
Modern Greek	Read, write, understand, speak *fluently*
French	Read, write to some extent, need practice for serious communication
German	Read, write to some extent, need practice for serious communication
Swedish	Read, understand (it is similar to Danish and I worked a bit in Sweden)
Norwegian	Read, understand (it is similar to Danish)

Software Development Skills (Updated 2013-10-18)

Skill Name / Group	Level	Latest Use	Years Active Use
Programming Languages:			
· Standard ML (a.k.a. SML '97)	Expert	2013	16
· CeXL (language that I created)	Expert	2007	2
· OCaml (Standard ML dialect)	Routine	2002	0.2
· MLFi (OCaml dialect)	Lots of Routine	2011	2.5
· Haskell (lazy Standard ML dialect)	Good Knowledge	2003	0
· Scheme, LISP	Good Knowledge	1999	0.1
· C#	Lots of Routine	2011	2.5
· Python	Lots of Routine	2005	1
· Java	Good Knowledge	2003	0.5
· C	Expert	2012	9-15?
· C++	Lots of Routine	2006	9-12?
· Objective-C ("a mix of C and Smalltalk")	Lots of Routine	1998	2
· Delphi/Pascal	Lots of Routine	2006	4
· APL	Some Knowledge	2011	2.5
· RenderMan Shading Language	Lots of Routine	2007	4
· MEL (Maya Embedded Language)	Good Knowledge	1998	1
· Shell scripts (bash, Unix, DOS)	Lots of Routine	2012	10?
Other Kinds of Languages:			
· Operational Semantics (prog. lang. specification)	Lots of Routine	2010	8
· Denotational Semantics (prog. lang. proofs)	Routine	2007	0.5
· BNF-Grammar	Lots of Routine	2010	7
· Regular Expressions	Lots of Routine	2007	7
· COM/CORBA IDL (Interface Definition Lang.)	Good Knowledge	2001	1
· UML (Unified Modelling Language)	Lots of Routine	2011	5
· GNU Make (and OMake and Microsoft NMake)	Lots of Routine	2011	14?
· HTML	Routine	2012	15?
· LaTeX	Expert	2012	13
APIs, Interfaces, Protocols (Programming):			
· Standard ML Basis Library	Expert	2012	15
· Delphi Visual Component Library	Lots of Routine	2006	4
· Microsoft .NET Framework	Routine	2011	2.5
· OpenStep (MacOS X's API descends from this)	Routine	1998	2
· Swing	Good Knowledge	1997	0.1
· Document Object Model (DOM)	Good Knowledge	2001	0.1
· TCP/IP (Sockets)	Routine	2006	2
· RS232 (PC Serial Port)	Good Knowledge	2005	1
· Gtk+	Good Knowledge	2007	0.1
· XLib (X-Windows API)	Some Knowledge	1999	0.3
· SDL (Simple Direct Media Layer)	Lots of Routine	2007	7
· DirectX	Good Knowledge	2001	0.5
· RenderMan (API and RIB-files)	Expert	2012	6
· OpenGL	Good Knowledge	2007	0
· Maya Plugin API	Good Knowledge	2003	0
· LightWave 3D plugin API	Some Knowledge	2001	0
Databases (Programming Experience):			
· MySQL	Good Knowledge	2006	1
· BTrieve, InterBase	Good Knowledge	1996	2.5
· ODBC (standardized database API)	Good Knowledge	2001	1
· SQL (Structured Query Language)	Routine	2006	3
Operating Systems (Programming Experience):			
· Windows 95/98/CE/NT/XP/Vista/7	Routine	2011	9
· Linux	Routine	2013	15?
· Silicon Graphics IRIX, Sun Solaris	Some Knowledge	2001	0.2
Tools, Programs, IDEs, Version Control (Some Tools Are Covered Above):			
· MLton	Expert	2013	14
· SML/NJ	Expert	2013	16
· MS Visual Studio 2005, 2008, 2010	Lots of Routine	2011	5
· Rational Rose	Good Knowledge	2003	1
· gcc	Lots of Routine	2012	8-14?
· C++ Builder	Routine	2006	1
· Delphi	Lots of Routine	2006	4
· SVN	Expert	2013	6
· Git	Good knowledge	2013	0.1
· CVS	Lots of Routine	2009	10

Software Development Projects (Updated 2018-02-04)

2014 March - Now: Web admin systems, content and rating systems etc.

Employer: Jobindex A/S (www.jobindex.dk).

Project description: Employed as a full-time developer in the Jobindex concern, which encompasses Jobindex (www.jobindex.dk), Jobbsafari Norway (www.jobbsafari.no) and Sweden (www.jobbsafari.se), StepStone (www.stepstone.dk), it-jobbank (www.it-jobbank.dk) and ComputerWorld (www.computerworld.dk). During his employment, the now decommissioned site Jobbsafari Finland (www.jobbsafari.fi) was also live and the consultant worked on all sites mentioned above, except for www.computerworld.dk, though he worked on some of ComputerWorld's internal systems that are shared throughout the concern. Worked on core functionality of the websites as well as the webbased internal administration systems. This includes the frontpages, jobsearch, article and content systems, events/arrangements systems. Most notably, the consultant was one of the central developers in getting www.stepstone.dk and www.it-jobbank.dk online in 2015 after their acquisition by Jobindex. This included completely new frontpage designs and jobsearch rewritten and merged with the one from Jobindex. The consultant is the primary developer of the workplace rating system launced on www.jobindex.dk on 15th of January 2018.

Technologies used: perl, HTML, CSS, less, Bootstrap, Mojolicious, Javascript, jQuery, CGI, Gettext, git, Linux.

2014 July - Now: Image Rectifier (www.imagerectifier.net)

Customer: Hardcore Processing (www.hardcoreprocessing.com).

Project description: Designed and developed a high-tech commercial fully automated website, Image Rectifier (www.imagerectifier.net), that rectifies images such as photographs taken with an ordinary camera. Images are rectified such that parallel and orthogonal lines seen in the image become straight, vertical and horizontal. No prior camera calibration or knowledge about image content is necessary. This particularly requires computer vision technology, but also computer graphics technology, as well as many advanced algorithms all written from scratch.

- All web design, web server setup, software design and development from scratch
- Following, evaluating and further extending the latest cutting-edge scientific research in computer vision and computer graphics
- Automated testing of developed software modules, the core application as a whole and testing of the website
- Online payment integration
- Business development, planning and marketing of the whole project and product

Technologies used: Standard ML, MLton, SML/NJ, C, gcc, GNU Make, SVN, LaTeX, Linux, large amounts of books, scientific research articles and conferences attended.

2009 January - Now: CeX3D Inverse (www.cex3d.net/inverse/)

Customer: Hardcore Processing (www.hardcoreprocessing.com).

Project description: Designed and developed an upcoming and very high-tech commercial software application, CeX3D Inverse, (running on Linux and Windows) for automatic reconstruction of 3D models from photographs taken with an ordinary camera. No prior camera calibration or knowledge about the cameras is necessary. This requires a substantial amount of, in particular, computer vision technology, but also computer graphics technology. As part of these technologies are many numerical optimization algorithms, such as Singular Value Decomposition (SVD), Eigenvalue Decomposition, Cholesky Decomposition, Newton, Levenberg-Marquardt, Second-Order Cone Programming (SOCP) and others that were all written from scratch, which in itself is a substantial undertaking.

- All software design and development from scratch
- Following and further extending the latest cutting-edge scientific research in computer vision and computer graphics
- Automated testing of all developed software modules and the application as a whole
- Development of self-extracting installation programs, copy protection and license registration with online payment integration
- Business development, planning and marketing of the whole project and product
- Patenting, scientific documentation and evaluation of technology

Technologies used: Standard ML, MLton, SML/NJ, C, gcc, MinGW, GNU Make, SVN, LaTeX, Linux, large amounts of books, scientific research articles and conferences attended.

2013 February - 2013 October: Computer Vision Software

Customer: (confidential).

Project description: Consultancy via Hardcore Processing for a customer. Designed and implemented high tech computer vision software capable of detecting 3D objects in images and rectifying their surface into 2D at interactive speeds. Work for this customer was not with a full-time workload.

- Design and implementation of all the delivered software with sparring of ideas from the customer
- The first production version of the software was developed in roughly 3 weeks of work. This first version is able to load an image file of around 100kb, perform advanced computer vision processing, and save an image file of around 1Mb in roughly 150-200ms (including file load and save time) on a 3Ghz 64-bit computer at Hardcore Processing using only one CPU core
- An improved production version was delivered after a couple of months of additional work. The improved version can process the same data using more advanced technology and giving better results in typically 200-400ms on the same computer and using two CPU cores
- It was a customer requirement that no graphics hardware be used
- A Scrum-like development model was used with very fast development being prioritized over test and documentation
- Additional work on this project was done in 2016

Technologies used: Standard ML, MLton, GNU Make, SVN, Git, Linux.

2008 April - 2011 December: Technology for Financial Software

Customer: SimCorp A/S (www.simcorp.com).

Project description: Consultancy via Hardcore Processing for SimCorp on their financial software system, called SimCorp Dimension. Worked on integration of a developer kit into their software, for letting users write their own financial instruments, using technology supplied by a subcontractor. Established a substantial amount of test frameworks and procedures for many development and debugging tasks, including script automation. Made an extensive migration of the team's revision control system, leaving all prior development processes intact, leaving all released software versions retrievable while improving and restructuring the directory layout.

- Integration of a developer kit into their software, using technology supplied by a subcontractor

- Established test frameworks and procedures for many development and debugging tasks, including script automation
- Development environment improvements
- Migration of revision control system (from CVS to SVN)
- Passed SimCorp's C# Framework certification course

Technologies used: OCaml, MLFi, Cygwin, Windows, C#, .NET, APL, CVS, SVN, OMake, GNU Make, OUnit, Microsoft Visual Studio 2005, 2008, 2010.

2008 January - 2010 September: CeXL Compiler Technology (www.cex3d.net/cexl/)

Customer: Hardcore Processing (www.hardcoreprocessing.com).

Project description: Formal proof of type-soundness was made for most of the core calculus, called ksi-Calculus, of the CeXL programming language. A compiler for the language was designed and written with an advanced two-generation mark-region typed memory management system (garbage collector).

- Formal proof of type-soundness for a programming language
- Development and design of a compiler and an advanced memory management system (garbage collector)
- Scientific research

Technologies used: Standard ML, SML/NJ, C, gcc, GNU Make, LaTeX, x86 Assembly language, books and research articles

1997 - Now: Various Websites

Customers: Churchill Communication (www.churchillcommunication.dk), Rebild Vandrerhjem, Hardcore Processing (www.hardcoreprocessing.com), and Anoq Music (www.anoq.net).

Project description: Made a few websites over the years. The websites www.hardcoreprocessing.com, www.anoq.net, and www.cex3d.net are continuously being maintained as necessary with very little effort, since they are almost completely automatically generated, including most PDF and PostScript files, from simple LaTeX code by GNU Makefiles and a small program written at Hardcore Processing around 1999 (and that program *is* in need of an update ;-)).

- www.churchillcommunication.dk: Made in February 2013 for Churchill Communication (given the website texts, design and pictures), using CSS and a subset of HTML5 that is also valid HTML4, to keep it simple and as backwards compatible with browsers as possible. Some of the smaller graphical elements were also made
- www.vandrerhjem.net (no longer online): A website made in 2001 for Rebild Vandrerhjem using plain HTML. All graphical elements were also made
- www.hardcoreprocessing.com, www.anoq.net, and www.cex3d.net: Mostly autogenerated HTML, PDF, and PostScript from LaTeX. All graphical elements were also made. www.anoq.net additionally uses a bit of Javascript. The download and purchase pages of www.cex3d.net also use PHP and integrate with Paypal

Technologies used: HTML, CSS, PHP, MSP, Javascript, LaTeX, Paypal APIs, GNU Make, Standard ML, Photoshop, GIMP.

2006 June - 2006 December: Bluetooth Headset Tests

Customer: Sennheiser Communications (www.sennheisercommunications.com).

Project description: Consultancy project via Prevas A/S. Test of wireless headsets (Bluetooth), including substantial test specification improvements.

- Test specification
- Manual testing and error reporting
- One code-review session (a few hours) of sub-contractor code that uncovered around 8 errors in around 8,000 lines of code and solved several problems

Technologies used: Microsoft Word, C, C++, Visual Source Safe.

2005 October - 2006 June: DECT Product Tests and Development of Software for Test Equipment

Customer: RTX Telecom A/S (www.rtx.dk).

Project description: Consultancy project via Prevas A/S (named Glaze A/S at that time). Development of software, using C++ Builder, for operating custom-built test hardware to perform tests on target units, such as: Software flash load, RF-tests, audio tests, power measurements and software tests. This project involved working remotely from Copenhagen and on-site in Aalborg.

- Development of user interface and software
- Specification with customer cooperation
- Documentation of software and the tests performed by the software

Technologies used: C++ Builder, Microsoft Word, flash loading.

2004 September - 2005 September: Umra Vehicle Detection System

Customer: Exensor (www.exensor.se).

Project description: Consultancy project via Prevas A/S (named Glaze A/S at that time). Sensor-based vehicle detection system (UMRA). Made design, documentation and implementation of basic system interfaces and a multi-threaded radio communication protocol. The system was implemented in Embedded Linux and Windows, in Python and C, using RS232, Radio link and TCP/IP, on PC104 and standard PCs. This project involved working partly in Copenhagen and partly in Malmö, Lund and Stockholm in Sweden.

- Design and documentation of basic system interfaces
- Real-time multithreaded implementation
- Automated unit testing of implementation
- Integration with hardware sub contractor
- Choice of System Architecture
- Definition of project wide documentation standard
- Project specification and planning

Technologies used: Python, C, RS232, Radio link, TCP/IP, PC 104, Embedded Linux, Windows.

2002 February - 2004 September: Internet Payment Systems and Other Projects

Customer: NetGroup A/S (www.netgroup.dk).

Project description: Freelance work via Hardcore Processing. Development and design of several projects, including small tools running on platforms such as SGI IRIX, Sun Solaris, Linux and Windows, but most notably work on some Internet payment systems.

- Development on several projects and of small tools
- Work on Internet payment systems

Technologies used: C, PHP, gcc, ASP, CGI Scripts, COM, IDL, POSIX API, ODBC, Linux, Windows, SGI IRIX, Sun Solaris, ISAPI Plugins, Zeus Webserver, UML, NMake, GNU Make, CVS, Microsoft Visual C++.

1999 August - 2004 July: CeXL Language Specification, Front-end and Interpreter (www.cex3d.net/cexl/)

Customer: Hardcore Processing (www.hardcoreprocessing.com).

Project description: Made a formal specification of a programming language, CeXL, which is a large subset of Standard ML with some extensions allowing polymorphic extensible records with optional fields. This has involved a significant amount of scientific research. Implemented an interpreted version of CeXL, including a full front-end with parser and type-inference.

- Formal semantics and programming language specification
- Implementation of parser, interpreter and type-inference for type-checking
- Scientific research

Technologies used: Standard ML, MLton, SML/NJ, C, gcc, LaTeX, Linux.

1998 September - 2004 January: Various Projects

Customer: Hardcore Processing (www.hardcoreprocessing.com).

Project description: Designed and developed several pieces of software, including:

- AbstractUI::ML and UI::ML: 2 GUI toolkits for Standard ML (Linux and Windows, written and designed from scratch). UI::ML has been revised and improved in 2013 and 2014
- RenderMan and SDL bindings for Standard ML
- Programming (Linux and Windows) of 5 small computer games that were released as advertisements for various products
- Ported the runtime of an ML compiler (MLton, 60,000 lines of code) from Linux to Windows. New patches with minor corrections and improvements to MLton's runtime were made in 2014, to get all features of CeX3D Inverse 0.7.2.0 alpha to work properly on Windows (and partly Linux)

The projects UI::ML, the SDL bindings, and the RenderMan bindings are used in Hardcore Processing's commercial products CeX3D Converter and CeX3D Inverse.

Technologies used: Standard ML, MLton, SML/NJ, C, C++, gcc, MinGW, POSIX API, Win32 API, GNU Make, SVN, Linux, Windows.

2003 October - 2003 November: VueLink Protocol

Customer: GateHouse (www.gatehouse.dk).

Project description: Worked as a consultant for GateHouse on development and design of software for hospital equipment for a customer.

- Development and participation in design of an implementation supporting the VueLink protocol

Technologies used: C++, Rational Rose, Windows, Windows CE, Microsoft Visual Studio.

1999 August - 2001 August: CeX3D Converter (www.cex3d.net/converter/)

Customer: Hardcore Processing (www.hardcoreprocessing.com).

Project description: Designed and developed a commercial software application (running on Linux and Windows) for converting between different 3D graphics file formats: RenderMan, LightWave 3D and the level-editor for the 3D game Unreal.

- All software design and development from scratch
- Reverse engineering undocumented file formats and implementing those and other documented file formats
- Automated testing of the developed software modules and application as a whole

Technologies used: Standard ML, MLton, SML/NJ, C, gcc, MinGW, GNU Make, SVN, Linux, Windows.

1997 May - 1999 December: Open Source projects

Customer: Partly spare-time, partly Hardcore Processing.

Project description: Worked on development and design discussions of a few Open Source projects:

- GNU Step (www.gnustep.org): Implemented some of the Foundation Kit classes in the OpenStep API (now Apple Cocoa), e.g. NSAttributedString and related classes. Also worked on NSXKit, a former attempt at making the OpenStep Application Kit run on X-Windows using XLib. The OpenStep API (now Cocoa) is still the basis for development on MacOS X and iPhone (for those too young to know this ;-))
- Berlin and Warsaw (www.berlin-consortium.org at that time): Worked on design discussions and implementation of the Berlin (desktop environment) and Warsaw (development API) projects. A personal aim was a truly cross-platform and cross-programming language API that eventually led to the discovery of CORBA and its IDL (Interface Definition Language) and that it is possible to make an efficient CORBA ORB (Object Request Broker), such as omniORB. Made a C++ implementation of a part of W3C's DOM (Document Object Model) for omniORB (also works without omniORB). The latter was partly done by writing a tool (at Hardcore Processing) called CodeTransformer for reading IDL and C++ header files and generating C++ files

Technologies used: Objective-C, OpenStep API, C, C++, gcc, GNU Make, Linux, X-Windows protocol, XLib, CORBA, OMG IDL, omniORB, DOM.

1997 January - 1997 May: HK CD-ROM, Molslinien Training CD-ROM

Employer: Visionik A/S (www.visionik.dk).

Project description: Worked as an employed software developer on a training CD-ROM for ship personnel (Molslinien) and a computer game for entertainment.

- Software development
- 2D and 3D graphics work

Technologies used: Delphi, 3D Studio Max, Photoshop, Authorware.

1994 August - 1996 December: ELFO Meldesystem, Optics Shop System and More

Employer: Sandlykke & Leifsgaard A/S, later named TargIT (www.targit.com).

Project description: Worked as an employee software developer at Sandlykke & Leifsgaard A/S, later named TargIT. Development and design of a system for optics shops (Thiele, Profil Optik and others), which is used for customer service, storing human vision test results, making orders etc. Worked in particular on the implementation and design of reading glass databases from various glass manufacturers into a single database. Developed and designed a complete computer system for ELFO (Elinstallatørernes Landsforening). The system is for registering companies performing work on house electricity installations and checking that the Danish law of "licitation" is respected. Educated the customer in using this system and participated in user documentation.

- Software design and development
- Database design
- Customer meetings regarding software design and evolution of features and requirements
- Documentation
- Teaching end-users in system use

Technologies used: Delphi, C, JAM (Jyacc Application Manager), Pascal, ODBC, BTrieve, SQL, Microsoft Word, Works.

Courses and Conferences (Updated 2018-02-04)

Name	Topic	Level	Years	Duration
SimCorp's C# Framework Course	C#, in-house framework	certification	2010	2 weeks
ML Workshop	programming languages	Ph.D./research	2012	1 day
I gave one demo presentation				
EuroGraphics Conference	computer graphics	Ph.D./research	2014-2017	1 week each year
EuroGraphics Conference	computer graphics	Ph.D./research	2007-2012	1 week each year
SIGGRAPH Conference and courses	computer graphics	Ph.D./research	2017-2017	1 week each year
SIGGRAPH Conference and courses	computer graphics	Ph.D./research	2004-2011	1 week each year
SIGGRAPH Conference and courses	computer graphics	Ph.D./research	1999-2001	1 week each year
CVPR Conference and courses	computer vision	Ph.D./research	2013-2017	1 week each year
CVPR Conference and courses	computer vision	Ph.D./research	2011	1 week
ICCV Conference and courses	computer vision	Ph.D./research	2017	1 week
ECCV Conference and courses	computer vision	Ph.D./research	2014, 2016	1 week each year
ICCV Conference and courses	computer vision	Ph.D./research	2011, 2015	1 week each year
VISAPP Conference	computer vision	Ph.D./research	2011	1 week
Visionday Conference	computer vision	Ph.D./research	2012-2016	1 day each year
Visionday Conference	computer vision	Ph.D./research	2009-2011	3 days each year
VgPbg Symposium	volume/point graphics	Ph.D./research	2008	2 days
SGP Symposium	geometry processing	Ph.D./research	2008	3 days