Portfolio of Peter Friebel

Personal Details

Name:
Peter Friebel

Date of birth:
24th of June, 1967

Nationality:
Dutch

Assignments

Teaching System Engineering in Java, YA Sundsvall, 2018

- Taught 2 classes in basic computers science, programming and full stack system development in Java.
- Created a preparatory course for those students that did not met the requirements for the education.
- Totally rewrote the education plan and course plans. The full education consisted of the following courses:
  - Basic computer science
  - Introduction to Java Programming
  - Object Oriented Programming
  - System Architecture
  - Database design
  - Software Development Process
  - LIA 1 (learning by working at an IT company)
  - Enterprise Java Backend
  - Enterprise Java Frontend
  - LIA 2
  - Exam work

Scotty III, 2016, Drenti Games & GameTech

- Simplified PHP version of the Scotty web app framework for easy deployment at mainstream hosting companies.

Radical Retro Entertainment System 1 (RES 1), Irritum Radical, 2015

- RES 1 is a retro gaming console based on the Raspberry Pi 2 running Retropie. Because Retropie and the Raspberry are products tailored towards technically proficient persons with a great deal of knowledge in Linux etc., Irritum Radical packages and configures all required components for various scenarios and personal requirements. This is both a product and a service.

Naturum Fulufjället, NDIG & NQuiz, Irritum Radical, 2015

- Both programs are based on the Java Engine implementation of the Fringe engine.
- NDIG: A digital image grabber. Naturum wants to make a movie from a whole year of daily pics from their web camera.
- NQuiz: A grapical quiz engine. Uses Fringe with the Java 2D rendering engine. This system lets you create graphical quizes. This is running on one of their Linutop systems with touch screens (see below).

Fringe & Fringe Engine, Irritum Radical, 2015 -

- Fringe is a classified game development project. The game runs on mobile devices. I created a new version of Firesteel GDF that is called the Fringe Engine. This game engine can produce 3D and 2D games on many platforms including Android and IOS. In its prototype version Java, Java 2D and Android implementations are created.

Nexum, 2014

- For a French company (Nexum) and its classified new software development system, I designed and implemented TXP, a transactional process manager.

Magento web shop for HuskyGård and NilAlf, 2014

- Implementation of Magento based web shops for HuskyGård and NilAlf

Misc, 2013

- Various web camera assignments
- Maintenance and development for earlier customers

Framework and integration support and development Office Depot, 2012

- Office Depot (formerly Svanströms & Co) required various addition changes and support of the xml based eCommerce platform and integration layer to MOVEX that I designed and implemented.

Idre Golfklubb, 2012

- Design and implementation of a web camera solution. Requires a long distance wireless network bridge and deployment and hosting of an image server at IdreData.

HedData, 2012

- Irritum IdreData started a strategic partnership. IdreData and HedData cooperate in the design and implementation of larger ICT projects in northern Dalarna.

Vattenriket Kristianstad, 2011 - 2012

- For Vattenriket Kristianstad I created a web camera solution to monitor a nesting bird in the wilderness. This solution required a solar panel energy solution to power the camera and 3G Internet router.

Naturum Fulufjället, 2010 - 2011

- For Naturum i Fulufjället National Park I created a low power consumption digital signage and kiosk solution. I also created web cam solution with hosting.

BFD Travel & Event, 2010 - 2011

- Customization and deployment of Irritum Activity booking system.

Irritum IdreData, November 2010

- For my own software company, Irritum, I created an Open Source product called Irritum OE Small Business Tools. SBTools is based on Irritum IdreData's modified content management framework OpenEdit (Irritum OE Core). The system provides web developers with an easy way to implement e-Commerce functions without having to deploy complex back office systems. The system is a J2EE compliant web application.

Confidential destination company in Sweden, October 2010 – 2011

- For a destination company in Sweden, I designed an Internet Kiosk System. The system was intended to provide visiting tourists with terminals at various locations in the destination. The terminals provide access to the destination company's Internet Web site and its on-line booking system. The Challenge was to create a locally supported system that was cheaper than established providers of such systems. I have succeeded in designing such a system using industrial PC's and other components and using Linux.

Strike Games / Irritum Radical, November 2009 – 2011

- I became Lead Programmer and Co-Founder of the US based company Strike Games. I developed FireSteel, Strike Game's revolutionary GDF (Game Development Framework), 3D Graphics and Physics Engine. My company Irritum IdreData owns this project and is planning to diversify the engine to be used in 3D simulation projects, 3D data visualization projects and other games. FireSteel is largely written in Java but has the feature that it supports natively implemented optimized renderers.

Svanströms & Co, 2009

- Svanströms runs an e-Commerce system based on several frameworks that I designed and implemented. In 2009 Svanströms upgraded their MOVEX installation to M3. As I created the integration layer for MOVEX and the e-Commerce framework, I was asked to create a new one that would enable both MOVEX versions to be integrated transparently at the same time.

Cypoint, 2008 - 2009

- To assist Cypoint's projects at Svanströms I created various API's, tools and class libraries for Cypoint such as an import and conversion tool to load up data base information from PostgreSQL to Apache OFBIZ Entities.
- A Java API for performing real-time credit checks of persons and organization using a 3rd party providers web services.
- Various integrations and adaptations of my own application server and development framework Uhura (running at Svanströms) etc.

Irritum IdreData, 2008

- To address the needs of a market of small to medium sized local companies I designed and implemented an Open Source web application suite called Irritum iWeb Suite. The suite provides a framework for implementing simple but very robust e-Commerce and on-line Collaboration Systems. The product is currently in use at various companies and is continuously upgraded. The product is based on the Irritum OE Core a Content Management Framework based on OpenEdit but enhanced to support n-tier distribution of services and web based distribution of updates.

Irritum IdreData, 2008

- Designed and implemented STACI (System Telemetry And Control Interface). STACI is a Java Mobile Agent Framework. This framework is based on and a continuation of IBM's Aglets Framework. STACI was engineered for a different purpose though. STACI can be used to create massively distributed systems with actual moving code. In combination with various levels of artificial intelligence mobile agents can achieve various levels autonomy. Using agents is also the only viable way for certain types of networks that suffer high latency such is the case in controlling space crafts. Using mobile agents instead of the client server model can hugely improve the performance of certain system. STACI is deployed at various sites. Agency has huge potential for e-Commerce applications. At Svanströms STACI provides a continuous guarding and management of the many servers that are involved in their e-Commerce and integration systems.

Svanströms & Co / Irritum IdreData, 2007 – 2008

- Initially a R&D project at Svanströms and later continued in Irritum IdreData, I created the Irritum OpenEdit Core Framework. This framework is Open Source and the bases of many of Irritum's web systems. It is an adaptation of the OpenEdit CMF (Irritum IdreData is an official partner of EnterMedia). This enhanced framework enables the development of J2EE web applications with an n-tier SOA

and configurable distribution sites. The framework uses Spring, Velocity and many more Open Source technologies.

Svanströms & Co / Irritum IdreData, 2007 – 2008

- For the benefit of Svanströms and a few internal projects, I designed and implemented iProx. IProx is an intelligent information proxy. It defines proxy sites and a set of public API's on information resources. IProx sites can communicate with each other over a network and iProx sites manage distributed and long transactions. IProx serves as a PnP registry for information services and it defines an API to embed hooks in information systems. iProx provides leasing services for distributed resources and pre and post processors on the define API's. At Svanströms iProx was set up to provide real-time access to pricing information from MOVEX. Pricing calculations are so complex that iProx was needed to intelligently cache results for customers so they would be able to see their specific prices in the web shop. IProx was also used to create a hook into the web shop service to enable conditional and limited offers. The conditional offers were in fact STACI agents that had a life cycle of a few month. The offer agents docked with the e-Commerce service, used iProx to plugin to the defined hooks for a while and later returned with statistics.

Irritum Radical, 2006 – 2007

- Irritum has a project/department solely for one-off technical R&D projects called Radical. In 2006 I started on MAECAMS (Modular Alternative Energy Controll And Management System). The project's aim was to create a set of modular tools and systems for efficiently controlling and managing alternative energy sources. From this project several products emerged. To name a few:
- CoreNet: RS485 packet transport layer for communicating with micro controllers.
- FoxCore: Linux based micro computer completely designed by bth tec including rewritten device drivers for the CRIS architecture.
- AVRCore: AVR Controller board based on the AVR Mega UPC including operating system and communications. A project that today is still in the concept phase is the technical and functional design of a power supply model comparable to what is used in various satellites. The design uses several redundant main buses and can have multiple power sources such as PV Solar Panels, fuel cells etc. Charging of pluggable accumulation systems can be fully programmed over time. Multiple secure output channels with programmable output specifications. The main idea is to use space technology to create more efficient alternative energy systems here on earth.

Irritum Radical, 2006

- Designed a implemented Saron, a very simple One-Pad Encryption/Decryption package that can be used to communicate proprietary information between parts with absolute certainty of security and non-repudiation. This encryption method can not be cracked by any person or organization. The main application of this package is preventing industrial espionage. The package has since been released under GPL.

Apica / Irritum IdreData, 2006

- Designed and implemented ACE (Apica Control Engine). Apica developed network based management systems. ACE lets you easily configure many concurrent management tasks. An ACE systems consist of daemons and controllers. Irritum owns this framework and uses it under the name ICE. Many parts of ICE have later been incorporated in STACI, an altogether much more advanced management system. ACE/ICE was even used at Svanströms and integrated in JMAPI (Java Management API) using NAGIOS to query server health.

Irritum Radical, 2005

- Designed and implemented SR1. SR1 is a musical software synthesizer. The system supports as many concurrent voices as the host CPU can handle. This system was a show case for doing advanced DSP (Digital Signal Processing) on PC's. The system core was fully written in C and later, parts of the DSP algorithms were optimized in assembly language for various processors. The system supports all features of MIDI and has a plugin that makes it act like an old-school analogue synth. The system runs on Windows and Linux and its control canvas was implemented in Java so only the engine needed to be system specific. Hardware designs were made for embedding the system in a specialized system with dedicated DSP's

Svanströms & Co / Irritum IdreData, 2001 – 2008

- I was contracted to help implement Svanströms e-Commerce platform. Svanströms had bought a skeleton e-Commerce System from Idealock/Kanebiz a company I was one of the founders of. The system was based on my Open Source framework Uhura. Uhura is an optimized and light weight framework that supports vital J2EE standards but without the performance overhead J2EE systems can have. I was lead systems architect and programmer and in that function made sure a holistic view was kept of all the integration challenges. The system has been up and running for many years now without major failures. The Uhura framework has been kept up to date and gradually enhanced over the years. The framework is owned by me in my company Irritum IdreData.
- The most important enhancement that was made during that period was the design and implementation of PMagick. This subsystem allows developers to define server side processes in XML files by defining the service objects they are dependant on, which entity services to use, define those services and entities, create custom queries etc. PMagick takes this file and generates about 2/3 of the server side code for the system. This product brought down the frequency of bugs created in development enormously (much better metrics) and increased performance significantly.
- In this period I also created ePipe. ePipe is the message engine that processes all XML messages coming in and going out from Svanströms. It is in this product the integration with MOVEX and other back office systems is realized. It is also in this product that the external integrations are done by mapping the various XML standards that need to be supported to an internal on (cXML). As an example: ePipe integrates Visma's Proceedo procurement system with Svanströms order process. The systems were all deployed on Linux servers.

Idealock / Kanebiz, 2000 – 2001

- I was Head R&D at Idealock. From 2000 to 2001 I worked full time on gathering the set of tools and frameworks my department had developed into a single coherent Java framework for the development of large web based distributed systems. Idealock had a subsidiary company Kanebiz that needed the framework for their development of a full e-Commerce product for suppliers based on XML e-Commerce standards. The framework is called Uhura. It consists of Scotty which is a content management framework based on XML and XMLT, PStore which is an object - relational mapper, database abstractor and transaction manager with optional internal CLI database engine and an application server.

AGA Sweden & Finland / Idealock, 1998 – 2000

- I was Head R&D at Idealock. The first components of the framework that later got called Uhura were developed. These new components were based on experiences I had gained from the development of Cortex at Entra Data. They were first used to create a large CRM system for AGA.
- In 1999, together with a colleague, I created a web order system for AGA called AGA WebOrder. The whole system was based on the XML e-Commerce standard OBI.
- AGA was unimpressed with the time it took IBM to create bug fixes for WebSphere that we needed to deploy our java servlets. I decided to create our own J2EE servlet container called Uhura Http Daemon and supported the full J2EE specs (including security, a first in Europe at the time). All AGA systems were deployed on AS/400, AIX and Linux systems.
- Also, in this period, I did a research program to ascertain the feasibility of writing 3D game engines and games in Java. A project I later had a lot of benefits from when I got involved with Strike Games and the development of FireSteel.

Entra Finanskonsult / Tieto Enator, 1996 – 1998

- The last two years at Entra I was assigned to Entra Finanskonsult. This subsidiary needed to create a new leasing system with buy in from customers such as ABB. I became Lead Architect for the project and decided that the system should be developed in Java, a language only 1.5 years old at the time. The system was distributed over several services and some sort of application server architecture was needed. It was decided at first that the architecture would be CORBA based. After I visited Java One I realized that my application architecture for CORBA , called Cortex, mapped almost perfectly on the early specifications for EJB (Enterprise Java Beans). I changed Cortex to fully implement the EJB specification and it became the first EJB container with full container controlled persistence and transaction demarcation in Europe (according to Sun). This enabled Entra to develop Enterprise Java Beans, test them in Cortex and later deploy the system at customer sites using conventional application servers such as Weblogic Tengah, IBM WebSphere etc. once they implemented the EJB standard.
- I talked before a number of engineering groups such as from Saab Aeronautics about EJB and its technology and also educated Entra's own OO expert group in how to design large scale systems using EJB.

Entra Data, 1995 - 1998

- In December 1994 I moved from The Netherlands to start working for Entra Data in Sweden. I was hired to work on Entra Data's R&D department. I started working on Entra Data's commercial product called Phantom Hurricane. The product ran on OS/2 and Windows. During the first year at Entra I had created many API's supporting software development on Phantom such as:
- DDE subsystem.
- Dynamic menu system.
- DCOM container for embedding terminal emulators.
- Phantom was fully written in C. I gradually introduced OO concepts to Phantom by introducing C++ to the team.
- Entra also had a few products that were written in SmallTalk. I have created various drivers for use in a SmallTalk environment such as a driver and an API for a SmallTalk product to control a credit card embosser.

Mettler Toledo, 1994

- I took a freelance job at Mettler Toledo to design and implement an elaborate class library to enable developers to create CUA compliant GUI's on Mettler Toledo's scale products. The scales ran OS/2. Once this job was finished I moved to Sweden.

The Frobozz Magic Software Company, 1994

- I was Co-Founder and CEO of Hactar, a small embedded systems company. The Frobozz Magic Software Company commissioned a system from Hactor for their stock investment software to acquire text-tv stock price information. Hactar developed a PC and PS/2 interface card and I wrote the firmware and OS/2 device drivers.

FMT, 1993 - 1994

- I was Co-Founder and CEO of Hactar, a small embedded systems company. FMT commissioned a system to measure energy use, loss and flows in buildings. My colleague and I designed the hardware for this system which consisted of a computer system that was designed from scratch with elaborate I/O features. The computer system was called GUM and was build around a NEC processor that was Intel x86 compatible. I wrote the operating system and device drivers for the system. The system software was written using Borland C/C++. I completely rewrote the C startup library and some standard libraries to be able to cross compile and deploy the system on the embedded system.

Lecombi, 1992 – 1993

- I was Co-Founder and CEO of Hactar, a small embedded systems company. An auction house commissioned Hactar to develop a voting system for their procurement auctions. Hactar designed a twisted pair network technology and I designed and implemented the firmware and device drivers for the server PC. The

system was called HLAN.

Exact Software, 1988 – 1992

- I started my professional career at Exact Software in Delft, one of the largest ISV's in The Netherlands. I was hired to join the two persons that made up the Technical R&D department. Exact Software creates ERP, project planning and financial software. The systems consisted of 6 million lines of C code and only about 300,000 of them were operating system dependent. R&D's job was to design and maintain those 300,000 lines of code. This was The Runtime. Exact Software ran on DOS, Windows, OS/2, Mac and Unix. R&D had developed their own database engine. I created many parts of the runtime such as:
- Data Dictionary System. The Data Dictionary described every table and field in the system not only for the database but also how it was presented on screen and reports. This feature made writing software against the runtime almost like writing code in a 4GL.
- DLL subsystem of the runtime enabling DLL loading and building under DOS. This made DOS compatible with the other systems that all had dynamic library loading.
- Prototype of the Exact Operating System (EOS). The idea was to have a raw OS core.
- Company wide chat application using Novell's service architecture and AWK. The system was called Gonzo. It enabled all people on the local network to chat with each other and report bugs and opload and download files.
- Memory and system expander for the DOS applications. DOS did not take advantage of the protected mode of Intel processors so I made a runtime extension that did.
- A virus to find and remove virus infections.
- A sorting algorithm, used in the database engine, that was faster than Quick sort. It is to this day uncertain if I did not come up with a fully new algorithm.
- In the last 2 years of my time at Exact I was put in charge of major internal quality and automation project to redesign and automate many software development processes. The system ensured that every software developer could take any workspace, check out his project submit build jobs to build servers and get reports about software metrics, and statistics on individual programmers etc. During this project I gained extensive experienced in compiler design and software analysis (metrics).
- At Exact I started studying Object Orientation. I started with the AT&T's Glockenspiel C++ preprocessor and wrote a prototype runtime in C++.
- Some other languages I worked with in this period were FORTH and MODULA2

Fields of Expertise:

Computer science / research and development:

- Software Engineering
- Systems Engineering
- Database engine design and implementation
- Compiler design and implementation

- Embedded Systems
- Client Server Architectures
- SOA
- Systems Integration
- Operating system design and implementation
- Device driver design and implementation
- Artificial Intelligence
- 3D real-time hardware accelerated graphics rendering, OpenGL, OpenCL
- Digital Signal Processing (music industry, software synthesis)
- Development Tools and API design and implementation
- Distributed computing
- Software development frameworks (e-Commerce, web, distributed)
- Application research
- Analysis, profiling, debugging, problem solving
- Security and encryption

Programming languages:

Assembly language for many processors, micro controllers, transputers and DSP's, C, C++, Java, Lisp, Forth, BCPL, Pascal, Basic, SmallTalk, Modula, etc.

Technologies and Tools:

Java, J2EE, EJB, JMQ, LDAP, Spring, Tomcat, Sun Java Systems Web Server, GlassFish, JBoss, OpenEdit, OpenCMS, PostgreSQL, RMI, CORBA, Jini, JMAPI, JAAS, Swing, Javolution, XML, XSLT, CSS, Ajax, Jason, JSP, Velocity, Lex/Yacc, Antler, Ant, Eclipse, Emacs, RTSJ, SDL, OpenGL, OpenCL, GLSL, SOA, Unity, Unreal Engine and many more.

Currently most used operating systems:

Linux, MacOS, Microsoft Windows

Spoken and written Languages (fluent):

English, Swedish and Dutch