

# Peter Schneider's Curriculum Vitae

## Timeline

- 1964** May 27: Born in Hannover, Germany, as the son of Gudrun Schneider and Hans Peter Schneider.
- 1980-** Learned BASIC, ALGOL68, FORTRAN and C by self study and evening classes.
- 1983** German Abitur (general university admission) with main subjects Physics, Latin, German.
- 1983-1989** Studied Physics in Hannover and Political Science at the *Freie Universität Berlin*
- 1987-1994** Div. student jobs; 1990-1994 proof reader at the taz newspaper, and evening taz dealer in bars and pubs.
- 1992** Six months internship at the taz IT department; learned and programmed C++.
- 1993-1998** Studied general computer science (*Allgemeine Informatik*) at the *Technische Fachhochschule Berlin*, finished with a *Diplom (FH)*, similar to a BA. The course focused on software engineering and software development. For my thesis I built a Linux box which buffered and multiplied the output of RIPs for large plotters. It included Linux kernel hacking to enable more than three parallel ports, as well as a simple controlling web frontend.
- 1994-1997** Student's supervisor in the programming and software engineering lab; in 1995 tutor for software engineering, introduced and tutored new users to Debis' CASE tool Promod.
- 1996-1998** Part time student job: system support, FTP server maintenance, and finally building and programming the Linux box for my thesis.
- 1998-2006** Working as a software engineer and QA engineer mostly on a freelance basis.  
Major assignments:
- 1998: Java frontend and some backend programming in a three tier application for a magazine lending company
  - 1999: Java frontend programming in a database application for pricing and billing phone calls
  - 2000-2002: C++ frontend and some backend programming of a car navigation device under VxWorks, using VisualStudio for developing against a PC simulation
  - 2003-2004: C/C++ system programming under Linux for a PCL raster image processor. Emphasis on fast PCL parsing, TCP/IP, interprocess communication and PCL internals
  - 2004-2006: Test engineer for embedded automotive systems. Developed automated tests e.g. for power moding; in-depth anaylsis and simulation of MOST and CAN communication. Mostly CANoe/CAPL.
- 2006-2008** Sabbatical; One year trip through the United States, visited most of the National Parks with my family after the birth of our son Robin.
- 2008-2009** Developing system software for an embedded system in the electronics/broadcasting environment, Linux, PowerPC architecture. Extensive cross compiling experience, including building cross compilers and setting up cross compiling environments. Debugging/adapting of open source software (gtk/directfb).

## Hard Skills

### As software developer:

- Extensive experience as software engineer in diverse environments:

- Good understanding and use of the OO paradigm in program design and implementation
- Good understanding and use of design patterns
- Good experience in system programming
- 3+ years of experience in programming embedded systems
- Experience in cross compiling, including building cross compilers and setting up cross compiling environments
- Excellent C/C++ knowledge, including the language's strengths and pitfalls.
- Good Java knowledge.
- Experience with managing and using version control systems (CVS and ClearCase), Doors.
- Experience with large software frameworks (GhostPCL, proprietary class libraries), including the build process.

**As tester in the automotive environment:**

- Excellent hands-on experience in testing:
  - Assembling documentation; designing, implementing and performing significant, reproducible tests
  - Assessing and communicating test results, qualifying software versions
  - Reviewing code.
- Excellent knowledge of automotive communication via CAN and MOST bus.
- Good/excellent knowledge of the Vector tools, especially CANoe with CAPL, extensive experience in CAPL programming, controlling devices via GPIB.

**Other:**

- Some Linux system administration.
- Some SQL, basic database management.

**Soft Skills**

- I actually do read documentation.
- I actually do write documentation (I like doxygen/javadoc).
- Thorough and persistent.
- Excellent English, both written and spoken (yes, German too).
- Good communication skills, team player, integrative personality, can cooperate with difficult characters.

*Last update May 2009*