RESUME

PERSONAL INFORMATION

Fernando Luis Cacciola Carballal

Born February 22, 1971 in Capital Federal, Buenos Aires, Argentina.
Married with 2 children.

Personal web page: http://fcacciola.50webs.com

Personal email addresses: fernando.cacciola@ieee.org
fernando.cacciola@computer.org

EDUCATION

FORMAL:

- **1991-1995**
  Biochemistry
  University John F. Kennedy

  Degree incomplete after 9 semesters (4 1/2 years) of continuous coursework, with 22 courses approved up front (all via final examinations), only 1 failed final exam (at first try) but approved in the second, and only1 failed and never approved (course title: business administration).

  Biochemistry is a 7-year degree here in Argentina (6 of coursework + 1 of residence), with no undergraduate diploma, so I ended up with no higher education diploma at all because I had to drop due to my full-time day job (see professional experience for details).

  Nevertheless, having fully completed the first 4 years of this degree I consider fair to relate my education to that of a Bachelor in Science.

- **1984-1989**
  Chemistry Technician,
  Technical High School E.N.E.T 27

COMPLIMENTARY

- **Computer Graphics**: 1999, Universidad de Buenos Aires (UBA), CS department. (1 week)
- **Cryptology**: 1999, Universidad de Buenos Aires (UBA), CS department. (1 week)
- **English**: Interlab Institute, 1995-1997. **TOEFL: 637** (in 1998).
- **Basic Electronics**: CEPA Institute, 1989.
- **BASIC programming**: IAEP Institute. 1984-1985
- **Assembler (6809) programming**: IAEP Institute. 1986.

SELF STUDY

From age 15 I have always been self-studying programming, general computer science topics, basic software engineering and some specific fields like Computer Graphics, Image Processing, Numerical Methods, Computational Geometry and Geometric Computing.

SKILLSET

- Master Level:
    - C++, C and C# (dotnet 3.0) programming.
    - Geometric Computing.
    - Data structure and algorithm design.
    - Generic and Generative framework architecture and design.
- Advanced Level:
    - General Numerical Computing.
    - Image Processing.
    - Computer Graphics.
    - Concurrency (Thread safety and synchronization)
    - Networking (TCP/IP [sockets], HTTP)
    - Design and Code Patterns.
    - SVN/Trac.
    - Unit Testing.
    - Technical Writing.
    - Cmake

- Intermediate Level:
    - Linux.
    - Scripting: Bash, JavaScript, Python and Ruby
    - Latex.
    - UML
- Introductory Level:
    - Agile methodologies (Scrum)
    - Functional programming.
    - Web technology (HTML,DTD,XHTML,XML,XSLT,XSD,XQuery/Link/Pointer).

CAREER

CONSULTANT

September 2003 – present
Founder of SciSoft (www.scisoft-consulting.com)

- Developed from scratch, in C# (with bits in C++) , the *entire* back end of a large ”vector drawing application” for a US customer (ongoing).
- Ported the CGAL (www.cga.org) bash-based build and testsuite system to CMake (www.cmake.org).
- Extended the CGAL Halfedge Data Structure to support Holes in faces and multiple boundaries.
- Designed, implemented and submitted to CGAL an algorithm for Triangulated Surface Mesh Simplification based on a paper by Peter Lindstrom and Greg Turk: “Fast and memory efficient polygonal simplification”. As a derivative work, designed, implemented and submitted to CGAL the BGL (Boost Graph Library) interface.
- Designed, implemented and submitted to CGAL an algorithm for Polygon Offsetting based on the Straight Skeleton.

2004
CGAL

The CGAL project (www.cgal.org) (Computational Geometry Algorithms Library) is a C++ library that captures the research work on geometric computing carried on in various European-based research centers and universities. As a result, CGAL represents the state-of-the-art in geometric computing. Usually, it is in CGAL where the advanced techniques of the Exact Computation paradigm, like floating-point filters, restricted predicates, lazy constructors, etc... are first implemented.

Early 2004 I was invited by the CGAL head to integrate into CGAL my Straight Skeleton and Polygon Offsetting code (which I had been advertising on the Web those days). This involves creating and submitting for review to the CGAL Editorial Board a Formal Specification of the package consisting of the code and the user/reference manual.

To do that I:

- Learned Linux and Latex from scratch.
- Learned the state of the art robustness techniques for geometric computing.
- Re-designed and re-implemented from scratch my earlier Straight Skeleton and Polygon Offsetting code.

After some work my code was accepted and integrated into CGAL and I was also invited to present it on the 2nd CGAL User's Workshop.

May 2003
Consultant Milestones that gestated SciSoft

- Wrote the following report for a US company: http://fcacciola.50webs.com/Offseting%20Methods.htm

- Designed, implemented from scratch and sold a Polygon Offsetting C++ library based on the Straight Skeleton using the CGAL library (www.cgal.org) (this is an earlier version of the one I developed for CGAL)

- Designed, implemented and sold a C++ library to produce a free-form deformation of a vector art-work based on a bezier-patch (like the one found in Corel Draw or Above Illustrator).

- Designed, implemented and sold a basic Image Processing Toolkit in C#.

EMPLOYEED

1995-2003
Sierra S.R.L (www.gosoierra.com)
Junior Programmer
Senior Programmer
Project Architect and Lead Developer of the C++ subsystem.

Company profile:

World-leading Argentine company (with offices in the USA) offering hardware and software solutions for the *Embroidery Industry*.

Entry position:

Assist them on rewriting from scratch, using Borland C++ 3.1 for Windows 3.0, their first Embroidery Design Software.

Junior milestones:

Adapted and implemented an early version of a "embroidery filling" algorithm they had developed but for a different kind of input shape, for which I self-thought, entirely on my own, the basics of **geometric computing**.

Started the design and implement ion of the ever-evolving geometric library.

Senior milestones:

As the R&D lead developer, came up with a filling algorithm to fill an arbitrarily-shaped polygon with holes via a constant-direction region-based zig-zag pattern. Learned about advanced data structures and algorithms, graph theory, discrete math and basic mathematical programming.

Guided and mentored the other five programmers in the team.

C++ architect/designer/programmer milestones:

With all the front-end development moved to Delphi and being now the only C++ programmer in the team, got completely in charge of the back-end layered framework offering general, numerical, geometric, topological, imaging and interfacing services to the front-end. Most of my contributions to the Boost library, and some of my earlier publications, originated in this engine.

Failed to implement an algorithm to fill a complex curvilinear shape with a single spiral path but learned about NC machining, offsetting, voronoi diagrams,computational geometry, and most importantly: *robustness issues in geometric computing*.

Implemented a Vectorization (or Autotracing) algorithm to turn a simple cartoon-like bitmap into vector art-work. Learned basic image processing. Developed and image processing toolkit including basic filters, morphological operators and color-reduction.

Learned about image-based Skeletonization and Medial-Axis-Transformation. Learned about the Straight Skeleton geometric structure. Implemented an algorithm to construct a straight skeleton and partition a polygon based on it. Developed and implemented an algorithm for filling TrueType glyphs by partitioning the glyph into columns that could be filled with a zigzag (using my straight skeleton based partitioning algorithm).

Designed and implemented the automated conversion process from a carton-like image into a complete vector embroidery design.

1993
Solydine S.R.L.
Junior Programmer

Entry position:

This time as an independent contractor develop a graphical audio-program scheduler to manage and play named audio-file lists

Milestones:

After delivering the product and during beta testing, detected and FIXED a bug in the 6809-based DSP firmware.

1991-1992
Solidyne S.R.L.
Junior Programmer

Employer profile:

Small Argentina company selling electronic equipment and software for the radio broadcast industry.

Entry position:

Develop, from scratch and in C, a graphical sound file editor application (similar to the wave-editors you get with a sound card these days) using the API to the card driver and a DOS-based GUI framework written in C++.

Milestones:

Under the mentorship of the lead programmer, learned about basic data structures and algorithms, memory management, application lifecycle, architecture and design methodologies, and testing strategies.

HOBBYST

1989

With my first PC, an 8086 PS/2, and having self-taught the C programming language, developed a DOS-based windowing system (ala Borland's OWL) with a clean separation between interface and implementation, then based on it, designed and implemented an “Insurance Management Application” for a local customer.

1988

Got my first PC, an 8086 IBM PS/2, and self-learned the C programming language.
Designed and Implemented a very simple “Wages and Salary System” for "Industrias Camporesi S.R.L" (which is my Dad’s Factory), for the CoCoII, in BASIC.

1987

Fully conceived, designed and implemented from scratch a program named HAL WRITER for the Tandy Color Computer II (CoCoII) consisting of a system extension, written entirely in Assembly language for the Motorola 6809, hosting the BASIC interpreter within a graphic-mode environment via a custom character generator.

PUBLICATIONS

- August 2006
  “CODEF/CML - A novel type of serialization framework in C#”
  Published in ACCU's Overload magazine #74
  (http://accu.org/index.php/journals/1374)

- August 2005
  Proposals to the C++ standardization committee: JTC1/SC22/WG21 - C++
  (www.open-std.org/jtc1/sc22/wg21/)

  - “A proposal to add an utility class to represent optional objects”
    (www.open-std.org/jtc1/sc22/wg21/docs/papers/2005/n1878.htm)

  - “A proposal to add a general purpose range-checked numeric_cast<>”
    (www.open-std.org/jtc1/sc22/wg21/docs/papers/2005/n1879.htm)

  - “A proposal to extend numeric_limits<> for consistent range query”
    (www.open-std.org/jtc1/sc22/wg21/docs/papers/2005/n1880.htm)

- June 2004
  "**A CGAL implementation of the Straight Skeleton of a Simple 2D Polygon with Holes.**"
  2nd CGAL Users Workshop, 12 June 2004, New York, USA
  (www.cgal.org/UserWorkshop/2004/straight_skeleton.pdf)
  Also presented also at: GENEPI (Associated Team Géométrica - Polytechnic University of Brooklyn)
  (geometry.poly.edu/genepi/juin/)

- October 2001
  "An improved variant type based on member templates"
  C/C++ Users Journal
  (www.cuj.com/documents/s=8034/cuj0010cacciola/cacciola.htm)

- September 2000
  C++ Tip #2: **"Generic constants for generic programming"**
  C/C++ Users Journal

- May 1999
  "A point-in-curve algorithm for polyarcs"
  WICC99, Universidad de San Juan, Argentina.

PROFESSIONAL AFFILIATIONS

Member of:

- **ACCU** – *Professionalism in Programming (Formerly the Association of C and C++ Users)*.
  (accu.org)

- **ACM** - *Association of Computing Machinery*.
  (www.acm.org).
  Affiliated to **SIGGRAPH** - *Special Interest Group on Graphics and Interactive Techniques*.
  (www.siggraph.org)

- **IEEE** – *Institute of Electrical and Electronic Engineers*.
  (www.ieee.org).
  Affiliated to **IEEE Computer Society**.
  (www.computer.org)

OPEN SOURCE COMMUNITY

Active developer of the **Boost C++ Project**.
(www.boost.org/people/fernando_cacciola.html)

Mentor for Boost on the Google Summer of Code 2008

Author of:

- **Optional** library.
(www.boost.org/libs/optional/doc/optional.html)

- **Numeric Conversions** library.
(www.boost.org/libs/numeric/conversion/doc/index.html)

- **Value Initialization** utility
(www.boost.org/libs/utility/value_init.htm)

Active developer of the **Computational Geometry Algorithms Library – CGAL Project.**
(www.cgal.org)

Author of:

- **2D Straight Line Skeleton and Polygon Offsetting** package.
(www.cgal.org/Manual/3.3/doc_html/cgal_manual/Straight_skeleton_2/Chapter_main.html)

- **3D Surface Mesh Simplification** package.
(www.cgal.org/Manual/3.3/doc_html/cgal_manual/Surface_mesh_simplification/Chapter_main.html)

- **Boost Graph Library and CGAL** package (contributing author).
(www.cgal.org/Manual/3.3/doc_html/cgal_manual/BGL/Chapter_main.html)