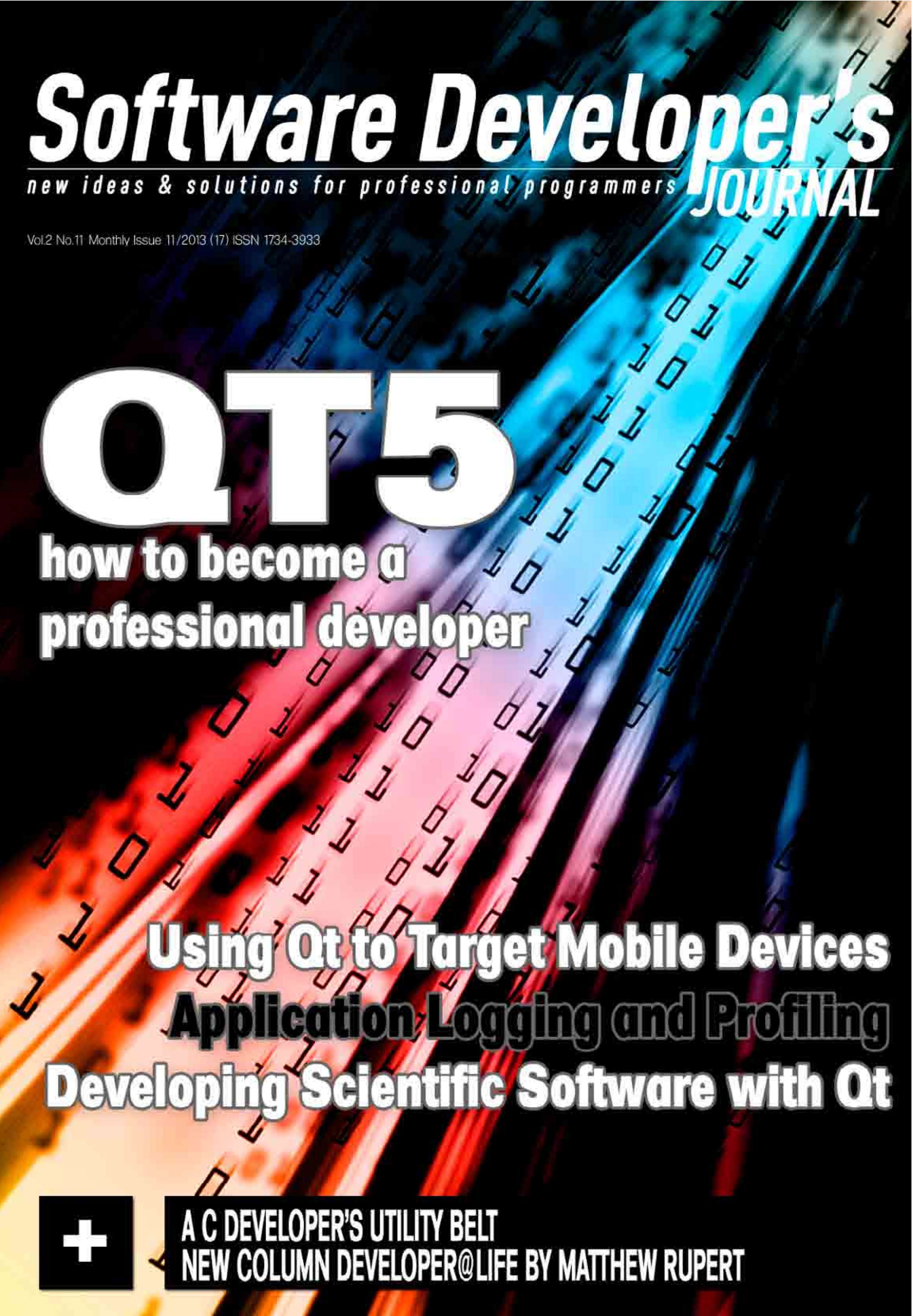
Software Developer's Journal

new ideas & solutions for professional programmers

Vol.2 No.11 Monthly Issue 11/2013 (17) ISSN 1734-3933



QT5

how to become a professional developer

Using Qt to Target Mobile Devices

Application Logging and Profiling

Developing Scientific Software with Qt

+

A C DEVELOPER'S UTILITY BELT
NEW COLUMN DEVELOPER@LIFE BY MATTHEW RUPERT

Dear Readers,

We've decided to start this summer with the topic that remains exotic for the majority of software developers, even those most geeky ones. If you've ever tried to find any support using Qt, you probably already know what I'm talking about.

This leading cross-platform application and user interface (UI) development framework were initially designed by Haavard Nord and Eirik Chambe-Eng from Norwegian company Trolltech in early 90s, lately developed by Qt Project and Digia. Qt uses standard C++ and is mainly used for developing application software with a graphical user interface (GUI). This year Digia released Qt 5.1 and commented: 'Qt 5.1 includes significant improvements to Qt Quick, the Qt-specific declarative UI language making it easier to develop graphically-rich UIs quickly and efficiently, it brings in new APIs, and improves performance and stability with the implementation of over 3000 enhancements to the previous version. The new version also introduces preliminary support for Android and iOS(...)' (Digia,Qt). That's why we're presenting some of priceless tricks used by the best Qt developers. We can assure you, you won't find them anywhere else!

Furthermore, you'll be able to read some +EXTRA articles related to the main theme. We hope you'll find them interesting and useful as always.

Finally, we're delighted to announce that we have a new team member. Matthew Rupert will cover the topic of Developer's@Life in his column. You will find his brilliant cogitations and advices in our up-coming, two- monthly WordPress line. We're also working on a special blog adjusted to your needs so that you could easily get into interaction with Matthew and share your comments with us. More info on our blog soon. Stay tuned.

Last but not least, I'd like to invite all of you to our partner, up-coming conferences. We're attending:

- *DevCon5 (July 23-24; December 9-11)*
- *It Dev Connections (September 30 – October 4)*
- *JAX London (Octoberc28- 30)*
- *Live!360 (November 18- 22)*
- *GOTO Aarhus (September 30 – October 4)*
- *Campus Party London (Septembe 2-7)*
- *CDN World Summit (October 1-3)*

Enjoy your reading!

Anna Lakomy
& SDJ Team

Software Developer's Journal
new ideas & solutions for professional programmers

team

Editor in Chief: Anna Łakomy
anna.lakomy@sdjournal.org

Editorial Advisory Board:
David Schneider, Amit Chugh

Special thanks to our Beta testers and Proofreaders who helped us with this issue. Our magazine would not exist without your assistance and expertise.

Publisher: Paweł Marciniak

Managing Director: Ewa Dudzic

Production Director: Andrzej Kuca
andrzej.kuca@sdjournal.org

Art. Director: Ireneusz Pogroszewski
ireneusz.pogroszewski@sdjournal.org

DTP: Ireneusz Pogroszewski

Marketing Director: Ewa Dudzic

Publisher: Hakin9 Media SK
02-676 Warsaw, Poland
Postepu 17D
Phone: 1 917 338 3631
http://en.sdjournal.org/

Whilst every effort has been made to ensure the highest quality of the magazine, the editors make no warranty, expressed or implied, concerning the results of the content's usage. All trademarks presented in the magazine were used for informative purposes only.

All rights to trademarks presented in the magazine are reserved by the companies which own them.

DISCLAIMER!
The techniques described in our magazine may be used in private, local networks only. The editors hold no responsibility for the misuse of the techniques presented or any data loss.

NEW COLUMN – DEVELOPER@LIFE BY MATTHEW RUPERT

QT'S SECRETS

EXTRA +

MATTHEW RUPERT'S COLUMN – DEVELOPER@LIFE

Hello World

Is it safe to say that all software developers, at some point during their journey, have written a "Hello World" program? The first program I ever wrote, the very first one, went something like this:

```
10 PRINT "MATT RUPERT "
20 GOTO 10
```

That's it. Two lines of Applesoft BASIC, written on an Apple IIe at the Auburn, Indiana public library. The results were immediate: I watched with enthusiasm as my name scrolls down the screen infinitely. I told the computer what to do and it did so! I was hooked. Steve Wozniak would have been so proud.

We weren't supposed to be fooling around with such things on that computer at the library, on no! And perhaps the rules regarding the computer added to the mystery and intrigue. The librarians, seemingly fearful of that yellowish-beige box with a clunky military-grade keyboard and green monitor, warned us repeatedly, "Only use the programs and follow the instructions closely, otherwise you may break it!"

While most of those programs were intended to be educational, the only thing I learned was that oftentimes people died of something called dysentery while travelling to Oregon. I hoped I didn't get dysentery, whatever it was.

I was curious. What if I didn't want to play *Oregon Trail* again? What other things could be done at that flashing prompt? *What if I wanted to make MY OWN Oregon Trail? How will it 'break?' What might I discover? Can I really do things like Matthew Broderick in War Games? (No.)*

The computer generally sat unused in a corner of the small, musty library, available to be used by "checking out" half-hour increments of computer time. As far as I could tell, only my friends and I used that computer. I never once saw a "grown-up" near the thing. This device, as 'breakable' as we were told it was, was something that the adults feared and the children cherished.

So as often as I could, I rode my bike to the library, about 2 miles from my house, in the scorching heat and the bitter cold to tinker on that fascinating machine, full of ideas and programs that I had written on sheets of paper. Oh sure, I did other things. I wasn't *that* strange of a kid. I played baseball and went to the pool and tied GI Joe figured to bottle rockets. But I also read books about Computer Programming and magazines like *Nibble* and *Byte*.

The six or so computer books, on a shelf right there beside, seemed to call my name, enticing me to figure out more. And as I took it upon myself to do more with that computer, I learned to cover the speaker with my palm so as not to alert the librarian that I was, once again, rebooting the computer. I suppose I became one might call a "library hoodlum." Hey, it's better than petty vandalism!

After many visits, the librarians no longer saw a need to hover while this scrawny boy messed around with the computer. Soon enough I was writing more and more programs, saving them on my own floppies, and (gasp), copying programs to play with at my friend's house. Zork was among my favorites.

While most kids begged for an Atari or an NES, I begged for a computer, and somehow, despite my mother's paltry income (paper route money helped), I ended up with one – my very own! It was a Laser 128 – an Apple IIe clone (with a whopping 128k of RAM, a full 64k more than the Apple IIe at the library). It had a color monitor. Eight colors! I was also fortunate enough to have a printer – An Epson 8 pin dot matrix – built like a tank, and probably somewhere in a landfill today, assuredly in working condition if someone with a parallel cable happens to find it.

My BASIC programs became more complex. Soon I was experimenting with other stuff – the bits and bytes that were not well documented in any of the 6 computer books at the public library. "PEEK" and "POKE" commands opened passages to endless fascination. Ask any dedicated computer programmer about his or her first computer and watch his or her eyes light up.

Like others in this field, I could go on and on about my early experiences with the Apple IIe and Laser 128. Even now, as I write about it, I cannot help but smile. And while I'd like to digress further about my beloved Laser 128, I must focus on my task here: An introduction.

Hi there! (Hello World.) I'm Matthew Rupert, Software Developer. Software Architect. Lead Software Engineer. Web Developer. SQA. Technical Writer. Technical Lead. Senior Developer-Level 1. Senior Developer-Level 2. I've had many job titles, and I've learned that a job title means little. I've had more job interviews than I can recall. Whatever the job, whatever the title, I've been doing this for the better part of my life, sometimes paid, often just for fun. Always enjoying it: Software Development.

There are many stereotypes in the non-techie world about *us* (those strange folks who like writing something called 'code' on computers). We're called eccentric. We're called introverts (arg!). We're called geeks (fair enough). We're assumed to be socially awkward (nonsense!). We like strange fantasy novels (so I am told). We don't know how to dress (so goes a stereotype based on evidence that makes it difficult to argue otherwise).

We may be all of these things (or none of them). Too often, by family and friends, we're just considered "the computer guy" (or gal). We are constantly asked the question: "My computer is acting funny, can you help me?" Yep, the endless question. Ours is a strange discipline – one that is difficult to explain to those who have never seen a line of C++ code.

I'll never assert to be the best at anything other than Metroid on NES. I've not the best software engineer. I'm not the best architect or UI designer. I'm far from the best DBA (very far). My specialty is this (and I hope many of you can say the same): A diverse and ever-growing range of interests, from the technical side to the social side.

Sure, we all have our quirks. A while ago I read a book with the title *Everybody's Normal Till You Get to Know Them*. A much simpler title could have been used: *Everybody is Weird*. What's our weirdness? We don't think of it much, I suppose, but it is this: Whereas most people are content to use a computer for email, Facebook, and writing, we want to know how it works – and what *else* we can make it do. We take delight in the bits that others might consider tedious or strange.

When I was offered an opportunity to write a column for SDJ I leapt at it. My interests, just like others in this field of 'work,' extend well beyond software development. I like people. I like interaction. I love going on a job interview (as you'll learn in my first article of this column). I try to play the guitar. I love running long distances (another thing people consider odd). I've watched the movie Office Space countless times, and each time I laugh with more vigor.

One of my greatest loves, aside from the world of software, is writing. Oh how I wish I could be the next Stephen King, sitting in my office dreaming up antagonists and protagonists and all of the struggles between!

Most of all, I love my family. I love spending time with my wife and children. Where do I find the time for all of these interests? I'll write about that at some point in the future. For now let me assure you, if you're wondering: There is time! Plenty of time!

SDJ is a great publication for developers. Why? Because it provides content on a wide range of subjects that should be of interest to all of us. We need to stay up on all the latest and greatest developments in the ever-changing world of software. We must (lest we flounder). While I will certainly enter into some of the technical subjects, my goal in this column is to discuss the other side of things – the *life* side. What goes on when we leave the cubicle? When we leave the building? How do we present ourselves in meetings, on a resume, at an interview, or in life in general (hence the title, Developer@Life). Subject matter of this nature may be called "soft skills." While I understand the description, it isn't one I necessarily like. These skills are not "soft" in any way. They are critical to the livelihood and career development of a software guru. I'm full of ideas, from interview advice to dealing with recruiters to maintaining our health.

And, via my SDJ blog, I am going to ask for your help. I am eager for this not to be a column all about me and my thoughts. I want to interview other developers. I want to respond to your questions and feedback. I want to make this a column interactive, lest it become another bullhorn of opinion from one guy.

So before I being, I'd like to say hello and thank you for reading. I hope you continue to do so, and I sincerely hope to hear from you!

One more thing: As for the *eccentricity* accusation of software developers – let's embrace it!

MATTHEW RUPERT
Family Guy
Software Guy
Writer