

Interview with JavaFX Developer Fabrizio Giudici

[Reviews Interactive](#) recently spoke with [JavaFX](#) developer Fabrizio Giudici, a prominent and highly-respected blogger in the JavaFX community who writes regularly for both [Java.net](#) and [DZone](#). The complete interview with Fabrizio follows. A shorter version and the podcast can be found [here](#).

Reviews Interactive: Tell me about your involvement with JavaFX. Where and how do you use it, on which platforms?

Fabrizio Giudici: I think that JavaFX is great for many targets, but I'm particularly interested in using it for mobile phones and similar devices. I've been developing with Java Micro Edition since when it was released in 1999, and I know that it's not an easy business as with regular Java. Furthermore, applications on mobile devices are mostly clients to remote services, must look smart and with all the bells and whistles – things that are really easier with JavaFX than with Java.

RI: How do you think JavaFX has changed or will change the way developers create RIA applications?

FG: It all depends on what we will see in the upcoming months in the field of tooling. In my opinion JavaFX could fill a gap between regular programmers and graphic designers – this would make it possible to create better looking applications and/or reduce the costs for developing a good looking application. The involvement of graphic designers is particular important; often excellent software developers, even with a developed taste for fine looking stuff, don't have themselves the skills to create fine looking stuff – but they believe they do, and the results are...well, not so good.

This has contributed to creating for years a bad reputation for Java clients, beyond the technical limits that Java had in the past. This is not a problem for industrial applications (I've had and still have a good deal of industrial customers working with Swing, Applets and feeling OK with them), but it is a big limit for end-user markets. Look at Apple – they have successfully entered the mobile phone market with a product that was (and probably still is) beyond many competitors for what concerns core technologies and features; but it's so cool and usable that many people believe Apple has invented smart phones. So far I'd say that only Apple has found the magic mix between the two worlds, but of course it's done in their own, proprietary, non-replicable way. Adobe Flex/Air has been the first to create a bridge in a more open way, and JavaFX can make the blend even better.

RI: Where do you use JavaFX in your work and/or daily life?

FG: My primary open source involvement is the [blueMarine project](#), started a few years ago. It encompasses a series of tools for managing photos and media, but I also

want to create a complete set of tools supporting a photographer in a broader sense. Taking photos is not only a matter of imaging: if you're serious (pro or not), you get engaged in the study of your subjects – they can be places, buildings, animals, nature, landscape...all of these things have bounds with history, geography, and so many aspects of human culture. These activities involve things such as taking and browsing notes, possibly geotagged notes, and documents, and using the related tools. For those things today you can use a generic notepad (paper or electronic makes no big difference), while I'm interested in more specific tools. That's why a few years ago I've developed [windRose](#), an expandable geotagging tool upon which I want to develop my ideas. Thanks to JavaFX I've developed, and am still testing [blueBill Mobile](#), an application to manage birder's observations, which is expanding windRose in a specific direction (birdwatching). I've got a working prototype of blueBill Mobile in roughly one week, a thing that would have been impossible with Java Micro Edition. And it wouldn't look cool.

RI: What new features do you find the most helpful in the new JavaFX 1.2 release?

FG: Well, above all, the new widgets, of course. Besides them, given that I'm interested in mobile development, the addition of portable classes to access local storage is important too.

RI: Have you already, or do you plan to develop JavaFX apps for the Java Store?

FG: Not yet. My current projects are all open sourced. I rather believe in the business that open source can induce, at least for the things I've in mind. But I'm not a fundamentalist of open source, so maybe in future you could find something by me in the Store.

RI: What do you see as the future of JavaFX?

FG: I can't speak for JavaFX TV – partly because it's a market I don't know, partly because it's related to activities that, as a potential user, I'm not interested too (I don't watch much TV actually). But I know I'm not a mainstream person, so I think it's a market with very high growth potential. The future of JavaFX tools interests me more, definitely. The JavaFX Authoring Tool is clearly one of the most important ones, but I also hope to see developments in tools for quality assurance, such as testing, static code analysis, coverage reporting and so on – the kind of stuff we got used to in the Java world.

RI: How do you think JavaFX compares to other RIA application development programs?

FG: I'm clearly biased here: it can be superior because it's Java based. One of the reasons I like the Java ecosystem is that with a relatively moderate investment in learning I can use it in such a variety of places that I surely didn't imagine thirteen years ago when I started working with it: devices, mobile phones, desktop, server side,

and real time too. Clearly, Adobe Flex/Air has got a big advantage since it has been available for a longer time. It all depends on how things go in the next year – and for the mobile stuff I think it's critical how many manufacturers will natively endorse it (thinking of big ones such as Nokia and Motorola) and how many existing devices will be able to run it by installing a runtime once for all. For the desktop, probably JavaFX has already passed its hardest times.

RI: What kinds of projects are you working on now and/or plan to work on in the future with JavaFX?

FG: I've already talked about blueBill Mobile. For the rest... who knows? In the latest two years the study and research about new technologies has proven not only to be fun itself, but it also brought me more business in new areas – it's good, so I don't get bored! Who knows, maybe I'll be working on a paid JavaFX project sooner or later.

RI: What do you like most about JavaFX?

FG: From the JavaFX Script perspective, binding is great. The declarative aspect is good, but I could live without it. Binding, on the other hand, can really simplify not only the implementation, but the design of an application too. Also, for the mobile profile, I like to have – at last – an unified, reference API for the user interface, and the guarantee I have a set of JSRs available below JavaFX – it's a way to fight the fragmentation of the platform.

RI: Where can people follow you online? (Blog? Twitter?)

FG: It might sound strange for an engineer working on leading edge technologies, but I'm pretty conservative when it comes to the user relationship with them. I'm a strict follower of “less is more” strategy, so I don't use a thing if it doesn't do something useful for me (this might sound obvious, but I think most people just use things because they are fashionable). Furthermore I don't like “real time” communications and I have got a strong sense of privacy. This clearly excludes Twitter and basically all the social networks such as FaceBook (I only enjoy a LinkedIn subscription that I see as a business service). So, beyond the web sites of my projects, I've got basically only two blogs: Java.net for the Java stuff (but I also regularly write for DZone) and Stoppingdown.net for the photographic stuff (mostly about photos and photo technique so far, but expect to see the involvement of my software applications as soon as they get ready).

###