# Software Technology

## Periodic Column in IEEE Software

## Editor: Christof Ebert

V11, May 2016

### Summary:

*Software Technology* is a periodic column that provides concise, hands-on information on technology that's just hitting the market or that's still meandering along the hype cycle. It offers guidance for those of you who need to get a quick and unbiased look on a recent technology, but don't have access to the latest analyst reports and neither time nor energy to cover all the relevant journals, conferences, and trade shows. Editor: Christof Ebert, christof.ebert@vector.com

### Vision and Format:

IEEE Software ( www.computer.org/software/ ) has this column on software technology since July 2007 with a self-contained article in each single issue. Its mission is to provide concise, hands-on information on recent software technology. Though it will not try to forecast anything, this column provides guidance for those who are not "blessed" to get all latest consulting and analyst reports on their desks and who don’t have the time to cover the relevant journals, conferences and trade shows.

This is a bimonthly 3-4 page summary on a dedicated software technology, the experiences and lessons learned with selecting the component or product, using it, or maintaining a product with this technology, so that other practitioners can directly benefit. The technology is evaluated in a concrete and not too narrow application context. It is presented in a neutral style.

This column presents new and promising software components, software technologies, or development tools. More than only providing a listing of vendors it answers questions such as: Is this technology already ready to use? What are the experiences from early adopters? Is there a real business case behind a certain technology or is it not yet mature? How to deal with conflicting standards in product development? We call it “Software Technology”, but this title also translates into "Interpreting the Hype Cycle" or more down to earth "Rrelevant software technologies for the leading practitioner".

As product life-cycles get shorter and an exploding number of technologies hits the shows and market, having some evaluation and explanation of worthwhile technologies and some kind of meaningful comparison with state of the practice seems useful. Our target is to engage useful feedback on some recent and useful technologies – not to drag people into hypes or to scientifically explain state of the art.

### Example:

Eclipse would have been something for the column some years ago. Let’s look how such an article would have handled Eclipse. The article would briefly describe why it will become relevant (freely accessible, open environment with plug-in mechanisms, attractive to those who search for a platform to share their tools and let them interoperate, etc.). Then the author would dive into 2-3 key aspects that drive the decision-making and would elaborate on technical issues to consider (e.g., the interfaces, its description, how to use debugging tools, how to use the trigger mechanisms, etc.). This part must be sufficiently detailed and technically sound to avoid marketing messages which would be considered superficial by our readers. Finally the article would rate Eclipse compared to what is already around and give some recommendation, what to do, how to evaluate, how to maybe build an own trial environment, etc.

**Contributors:**

The contributors must have concrete practical background, such as in other columns of IEEE Software. This keeps the contents sufficiently rich and ensures a variety of environmental conditions. It also avoids any biases in the broad software technology scene. They follow with their evaluation a common format (template). Preferably the authors are users of the technology, rather than primary authors or key contributors. They would typically not come from the vendor or inventor of such technology. They must not own the totality or portions of copyrights of the technology or product they describe.

Authors must not be “hostile” technology nerds, as this will bias the evaluation and thus reduce credibility.

**Style:**

Columns should follow our style guide. Here are some guidelines for layout and style.

Layout

- The column should start out with briefly explaining why this specific SW technology is relevant for readers’ development and products.
- Then list the questions practitioners and managers should ask to narrow the field, introduce a comparison and evaluation with other related technologies or solutions. This covers specifically the gap between state of practice and state of available technology. For example, is it a new/controversial/breakthrough/failed technology? Why is this topic important to a professional reader? How much background or technical detail is necessary? Please highlight the main "schools of though" or viewpoints on the topic.
- A section should compare the specific technology with its competition – be it legacy or competing technologies. Use graphics and tables! **This part is mandatory to provide an overview on the underlying technologies as a table with some evaluation criteria, such as platform, usability, access, OSS/Cost etc.**. Practitioners like to find information presented which easily conveys quantitative and qualitative information. Providing an evaluation table here helps for comparing pros and cons. Note that for legal reasons we must be precise in such evaluation what we can compare and what not. Typically the evaluation should either quote from available studies with quantitative results or remain qualitative.
- Then introduce the specific technology and its relevant capabilities, details and constraints. **This should be done hands-on so readers can transfer results to their own settings.**
- A last section must be titled “**Hints for the Practitioner**”. Provide here as much as possible concrete experiences, installation guidance, performance feedback, feedback from the respective community, evolution trends, information on releases, vendors, etc.
- **A side bar with a short case study presents how the technology is practically used**. Just 1-2 short paragraphs will show how the described technology is practically used and with which benefits, and with a literature link for further info on the case study.
- URLs and further references are key in such column. Ensure they are correct and updated frequently. When discussing a particular implementation, the piece should contain a URL where the technology can be obtained, information about platforms that it will compile and run on (and/or executables are available), licensing (GNU, etc.?). Please provide at least three links to previous stories on the topic, including related research projects and companies with related products. Please include a 1-2 sentence description with each URL, explaining its relevance.
- The column finishes with authors’ contact info, email address, and photo.

Style

- We encourage writing in the first person and active voice.
- Style must be neutral, that is no advertising or criticizing. Facts must be presented, no marketing.
- Avoid having anecdotal or persuasive treatments not backed up by authoritative analysis.
- Include some screen shots or code snippets where appropriate. This emphasizes practical usage.
- Have a look to IEEE Software author guidelines at: http://www.computer.org/software/author.htm

Size is 2-3 printed pages in IEEE Software (1300 words plus 1-2 graphs or tables).

Example column for download:
http://vector.com/portal/medien/vector_consulting/publications/LanubileEbert_GroupAwareness_IEEESoftware_2013V30N2.pdf

### Scope

Topics are hot software technologies – early in their life-cycle but ready to use and publicly available.

Scope is either evaluations within a vertical product group (e.g. applications, tools, components, frameworks, etc.) or a review of using a distinct technology (e.g. assuring safety in embedded systems with a suite of tools, implementing a certain quality attribute by using a new methodology and platform, etc.).

It's important that the selected technology segment is not too narrow and sufficiently interesting for a broad range of practitioners. Occasionally one might also want to discuss restrictions and constraints (e.g., where a technology is not working, impacts of specific license schemes).

The scope is not on scientific results, trend interpretation or state of the art. It is on technologies that are about hitting the market and will make a difference for the leading software practitioners and their products.

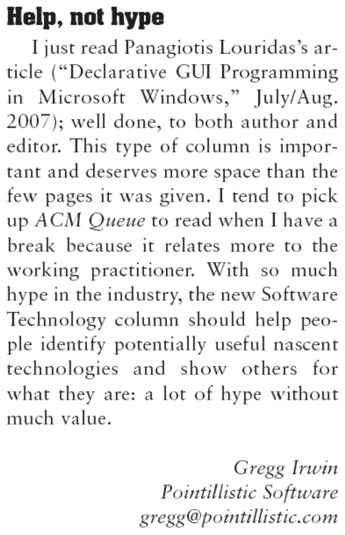
It’s key not to “ride only one horse”. The focus is on evaluation and comparison with related technologies. For this to be useful these evaluations must be in-depth experience reports, rather than superficial, “one weekend” evaluations. The readers expect an authoritative, balanced evaluation, which is fact-based, not opinion-driven.

Timeliness might be important if latest flavors are covered, however given size restrictions of the column format, this should not be a typical issue. People don't use a new technology just because of one new feature or because it’s looking hot.

### Column-Editor

The editor is Dr. Christof Ebert (christofebert@ieee.org , @christofebert ). He is managing director at Vector Consulting Services, a firm specializing on improving clients’ technical product development. Previously he was Alcatel’s Director of R&D responsible for software technology. The editor reviews any incoming submission. He might ask dedicated reviewers to further support technical correctness. He follows together with IEEE Software editorial staff each column from inception to publishing and answering to potential letters to the editor.

### Reader Feedback:



**Help, not hype**

I just read Panagiotis Louridas’s article (“Declarative GUI Programming in Microsoft Windows,” July/Aug. 2007); well done, to both author and editor. This type of column is important and deserves more space than the few pages it was given. I tend to pick up *ACM Queue* to read when I have a break because it relates more to the working practitioner. With so much hype in the industry, the new Software Technology column should help people identify potentially useful nascent technologies and show others for what they are: a lot of hype without much value.

*Gregg Irwin*
*Pointillistic Software*
*gregg@pointillistic.com*