Professional Developer Toolkit

Bonus Session

Rick Schummer
White Light Computing, Inc.
42759 Flis Dr.
Sterling Heights, MI 48314
Voice: 586.254.2530
Email: raschummer@whitelightcomputing.com
Web site: http://whitelightcomputing.com
Blog: http://rickschummer.com/blog

This session will cover several product categories and tools developers should consider adding to their toolkit, some of the competitive advantages they provide, and maybe even a demonstration or two of products Rick uses to give the perception to his clients that he is a professional developer.

Introduction

This session outlines many of the tools we use to run our companies, support our customers, and craft software, but are not tools we use to create source code (like Visual FoxPro, .NET languages, Java, etc.). This session is very dependent on participation by the attendees. Getting ideas from one developer is fine as a starting point, but gaining the experience of many developers is where the true benefits of this session come into play.

The plan for this session is to answer the following questions:

- What are the tools you use to distinguish you from the competition?
- What are the tools which provide a competitive advantage?
- What are the various software categories you should consider?
- What are the competitive advantages the categories provide?
- What are the solid products developers are using in these categories?

Disclaimer

There are a few disclaimers you need to read before getting into this whitepaper:

- We are not going to discuss Visual FoxPro vs. .NET vs. Java vs. Dabo vs. TheNextBigThing.
- This is not an all inclusive list of tools you need to be a professional.
- Just because you are not using one of the tools does not mean you are not a professional.
- Just because you are using the tools does not make you a professional.
- Each developer or shop have different needs, skills and types of software solutions.
- This is not a complete review of the software packages discussed.
- I have estimated each topic/category could be an entire conference session so we might have to limit the discussion and move on based on time to cover more.

Changes and updates

This whitepaper was developed in the first third of 2006. Tools come and go and what developers feel are the best during one snapshot in time can change in the future. If you are using tools not included in this whitepaper and have comments you feel would be beneficial to future readers of this whitepaper, please send them to me (Rick Schummer - raschummer@whitelightcomputing.com). I plan on updating this on a regular basis and publishing the changes/updates on my Web site (http://whitelightcomputing.com).

All you really need (from a humorous perspective)

We often look at complicated processes, methodologies, and best practices when deciding on the tools needed to run a successful business. The fact is there are just a handful of items you need to get the job done:

- Crystal Ball to see into the future. This is particularly handy when the clients want to know how their business is going to change. It is also handy so you will know what technologies will still be around for the next five years so you do not waste time learning new techniques in technologies vendors decide to throw away because they need to sell you something completely different.
- Magic Wand to perform magic. After all, our craft produces more magic than David Copperfield ever dreamed he could perform during his career.
- Time Machine is handy when there is never enough time in the day. Also handy when you are billing by the hour. This eliminates the fundamental problem with hourly billing with a limited number of hours you can sell.
- Genie in a Bottle is extremely beneficial when your user needs something in a hurry. Just remember genies are only allowed to grant you three wishes, so ask for more wishes for your third wish. If your genie does not have the additional wish clause, find out about this up front so you can manage your wishes more carefully.
- Telepathic Communication is good for many situations. Knowing what your customers are thinking instead of saying will clear up many of the problems you encounter when collecting requirements and doing technical support. It is also beneficial in situations when you need to be proactive with future features.
- Staples Easy Button is handy in those stressful moments when you are closing on a deadline and have too many things to do. While you can use a time machine some of these cases, the Staples Easy Button is less intense on energy sources. You can pick up the Easy Button for approximately $5.00 at www.staples.com.

While I am using a lot of humor in this section, I think it should be apparent we all need to look for the fun in our jobs and careers. OK, now for the serious list of categories and some of the tools considered best of class in each category.

Database design and modeling

Database modeling tools allow developers to design the databases outside of the native tools, independent of the target database.

Competitive advantages

- Better than native tools (duplicate structures, move columns, key propagation, data domains)
- Same model, different target databases
- Able to print designs and hang as wallpaper in the office

- Useful for technical doc / specs

Some best of class tools

1. xCase: http://www.f1tech.com/xCase/ and http://xcase.com
2. ERWin: http://www3.ca.com/solutions/Product.aspx?ID=260
3. DBDesigner: http://fabforce.net/dbdesigner4/
4. MySQL Workbench: http://dev.mysql.com/downloads/workbench/1.0.html
5. Visio: http://office.microsoft.com/en-us/FX010857981033.aspx

Comments from developers

1. Some design upfront is a good thing, no matter what you use. Some CASE tools are usually better than the native tools. The native designer works, but when you are writing tools with other databases, it can fall short.
2. Modeling new versus maintaining existing: it seems more developers use native tools to maintain existing databases, and some prefer to use the CASE tools for initial data modeling. Developers have mentioned that the tools are good when you first design/build, but has a different perspective once you are in development. He wonders if it is something that isn't holistic enough to handle the 'future' enhancements.
3. Some developers use SQL Compare and believe it is an excellent tool, and doesn't use CASE tools during development (preferring scripting tools to write code to manage database creation). After the database is stabilized you need to put it into something better.
4. CASE tools have two goals, to help you implement and to help you design. The same tool doesn't have to help you with both problems.
5. Version control is important with CASE tools, how they each handle version control is important. It is easier to integrate source code because it is integrated into VSS, but some developers are having trouble placing the scripts into the documents. SQL Compare can help take care of these problems, and some developers have discussed the interesting idea of using XML tools to accomplish this goal. Some developer shops require all scripts to be placed in version control. No tool is going to protect you from people being sloppy.
6. If you use Enterprise Manager, you write the script then run the script and it will do lots of stuff behind the scenes, but it's worth the overhead and not as necessary to have a CASE tool anymore.
7. The Stonefield Database Toolkit doesn't do data modeling, but can run routines with new data dictionary and updates and preserves the data changes to be moved into production. Visio Enterprise can do round trip design.

Bug Tracking

Plain and simple errors happen, and part of our job is to track the anomalies our applications reveal from time to time. One true difference between a professional and a pretender is how the professional developer responds to the problems presented to them.

Competitive advantages

- Better managed release schedules
- Better support for customers
- Track the history of reported issues
- Assign resources to fix and address issues
- Track the history of solutions, FAQs

Some best of class tools

1. FogBugz: http://www.fogcreek.com/FogBugz/
2. Dragnet: http://www.sourcegear.com/dragnet/
3. DBF: Use the VFP IDE or your own custom application
4. Word: http://office.microsoft.com/en-us/FX010857991033.aspx
5. BugCentral: http://bugcentral.com
6. Mantis: http://www.mantisbt.org/
7. Bugzilla: http://www.bugzilla.org/
8. Gemini: http://www.countersoft.com/
9. OnTime: http://www.axosoft.com/Products/OnTime.aspx?cn=otm_overview
10. IssueView: http://www.issueview.com/default.asp

Comments from developers

1. The nice thing about tools is they can be as simple or complex as you want it to be.
2. Make sure the software doesn’t make you fill out fields you don’t need.
3. Finding a tool where you define a work flow has a big advantage. New reports get assigned to triage, in triage you assign it to who it needs to research the report and take the appropriate action.
4. Many products force developers to compromise. This means the way the software works or information tracked does not exactly meet the needs for the company adopting the software.
5. Determining what is a bug is a very subjective process - don't necessarily want users doing this.

6. Some developers like to tie bug reports to the billing process.

Wiki

Wikis are dynamic Web sites allowing users to easily develop and manage content, but more importantly, research answers to common questions, and collaborate and communicate knowledge between team members or communities.

Competitive advantages

- KnowledgeBase for company
- Provides mechanism for collaboration on specs, ideas, future products
- Communication
- Easily accessible
- Scalable

Some best of class tools

1. OpenWiki: http://openwiki.com/
2. FoxWiki: http://fox.wikis.com/
3. TWiki: http://twiki.org/
4. Tiddly Wiki: http://www.tiddlywiki.com/
5. SharePoint: http://www.microsoft.com/sharepoint/default.mspx
6. Intranet: You build Web pages via HTML tools and post them on internal Web site

Comments from developers

1. Even small companies can run their own wiki.
2. Even one-person shops can run a wiki.
3. Many developers use it to replace all the sticky notes and provides collaboration in a team environment
4. Very scalable since it is based on Web technology.
5. You can use it is your own knowledge base. Expose a wiki to the clients for requirements.
6. Or you can just use HTML pages.
7. Rather then exposing it to Web server, put it on their network. Your wiki can be hosted internally on the Intranet, but you don’t have to host it yourself.

8. Some developers use SharePoint and can open documents through the SharePoint portal. When you open and edit it, will automatically get stored back to SharePoint. Allows you to use other MS applications in it.

9. Another developer uses SharePoint because there is a nice interface, and includes Explorer add-on, and there are permissions, versioning and publishing all built into it. In publishing, if you make a change, it will need to be approved before it is published. (Meaning you can keep working on it for awhile.) Portal technology gives you flexibility to pull data from lots of different places. Built in list management. List can help you track lots of things. The SharePoint server is not a free-be, but it is well worth it. Has a nice search capability, so you can search through Word, Excel, and PowerPoint. There are some concerns with administration and maintenance.

10. Some clients can't handle using a wiki, so it might be better to have a more stable document management system.

Testing Environment

Testing your applications in different configurations (both hardware and software) is a well known best practice. The difficult part is defining the combinations of the two your application will officially be supported on and how to test your solution on these configurations. There are tools to help you set up different configurations and to help with the testing of your applications.

Competitive advantages

- Solid releases, solid reputation
- Machine for each platform you deploy to (Win98, WinNT, W2K, WinXP, W2K3, Vista)
- Different user configurations (software and hardware configurations)

Some best of class tools

1. VMWare: http://www.vmware.com/
2. VirtualPC: http://www.microsoft.com/windows/virtualpc/default.mspx
3. Ghost/DriveImage: http://www.symantec.com/home_homeoffice/products/backup_recovery/ghost10/index.html
4. FoxUnit: http://www.foxunit.org/

Comments from developers

1. Some developers with TabletPC using VMWare have issues because of the multiple orientations.
2. Using a virtual machine isn't a cure all, you can't be sure it is going to work on their machines in their applications.

3. Strong opinions with many noting VMWare is much better then Virtual PC. Developer builds an image and runs everything in it except productivity tools. This enables him to take snapshots and clone images and still keep a stable copy. This way they can take a copy to show, and not messing with labels and tags in his source control software. One developer worked on a project where an image was continually being made and put onto the developer's machines so everyone was always working with the same thing.
4. Compuware puts on a one day show every year, and one developer watched a guy from HP give a presentation that developed a system where everything was done on a virtual machine and servers and the team starts developing. 80% of the live servers were development servers and sitting there idle. So you can have many machines running in virtual machines instead. At any point in time they can show what their code does.
5. Several developers use a combination of different machines and virtual machines.
6. Several developers feel Virtual PC is slow unless you have lots of physical RAM.
7. Some developers prefer to do all development (especially Web development) in a virtual machine.
8. FoxUnit is a great way to unit test your business objects, but not so good to test the user interface elements.

Source Code Control

Source code control (or as it is alternatively called, version control) allows you to track what developers changed, who made the changes, what they changed on what day, and allows teams of developers to not step on each others changes. When implemented correctly and natural policies followed, source code control will help you make sure only one developer changes source at one time and only tested software makes it into released builds.

Competitive advantages

- Version control (understanding when behaviors are introduced, bugs reduced)
- Team development with reduced aggravation
- Controlling build cycles

Some best of class tools

1. VSS: http://msdn.microsoft.com/vstudio/previous/ssafe/productinfo/
2. Vault: http://www.sourcegear.com/vault/
3. SourceOffSite: http://sourcegear.com/sos/index.html
4. NGSource: http://www.ngsource.com/SourceControl.htm
5. CVS: http://www.nongnu.org/cvs/

6. Subversion: http://subversion.tigris.org/

Comments from developers

1. CVS stinks!
2. You can use tools like Beyond Compare with version control software.
3. Some developers say using SCC integration in Visual FoxPro is asking for trouble, others live by it.
4. One developer uses source control to keep laptop and desktop in sync.

Creating Help Files

Creating Help files is not a fun task for many developers and is usually better handled by a technical writer. Finding a tool to develop a Help file is important to reduce this pain for developers and to create a professional looking Help file that is useful for your end users.

Competitive advantages

- Can be the one thing which separates your commercial product from the competition
- Can save technical support and reduce training costs (especially in IT Department)

Some best of class tools

1. West Wind HTML Help Builder: http://www.west-wind.com/wwhelp/
2. Doc-To-Help: http://www.componentone.com/
3. Help & Manual: http://www.helpandmanual.com/hmpage.htm
4. HelpBreeze HTML Help Edition: http://www.solutionsoft.com/hlpbrz.htm

Comments from developers

1. Writing a Help file can double the scope of a project.

Screencasting

Screencasting is the process of recording your computer desktop, and publishing the video for the purpose of communicating a message to the viewer who watches the video. This could be for the purpose of showing new functionality in your application, marketing the app to potential buyers, training end users, and even having your end users demonstrate problems back to the development team.

Competitive advantages

- Demo future features to clients without need for build and install
- Training

- Marketing

Some best of class tools

1. Camtasia: http://www.techsmith.com/
2. Wink: http://www.wink.com/wink%20screencast
3. HyperCam: http://www.hyperionics.com/hc/index.asp
4. ScreenCam: http://www.smartguyz.com/

Comments from developers

1. It is easier than it looks.
2. It is harder than it seems.
3. Great way to demonstrate hard to use functionality.

Instant Messenger & VOIP

Communication, communication, communication. The tools in this section help you communicate and collaborate with customers, team members, contractors, subcontractors, vendors, etc., for at least voice, but in some cases video.

Competitive advantages

- Saves on telecommunication costs
- Rapid access (not always advantage)
- Team collaboration
- Open communication

Some best of class tools

1. Skype: http://www.skype.com/
2. MSN Instant Messenger: http://messenger.msn.com/Xp/Default.aspx
3. GTalk: http://www.google.com/talk/
4. Vonage: http://vonage.com/

Comments from developers

1. Skype has been a great tool when I need to contact developers on my project. We all live in different geographic areas, so Skype saves us money on phone bills. The service is crystal clear.
2. I have called people all around the world with Skype and it just plain works.

3. Instant messenger tools allow me to get answers to my questions fast.
4. I sell mentoring services to other developers and provide them access to me via instant messenger to help facilitate communications. It is a business opportunity.
5. Allowing instant access to you for your customers is a two edge sword. Be careful who you give your IM ID to, or you could find yourself distracted by nonsense more than you want.

Remote Desktop

Problems happen and sometimes the easiest way to see what is going wrong is to watch the user in action. Remote desktop software and tools are a way for you to connect to your customer's computer or even another developer's computer and help them out with a problem. It is also a great way to connect back to the office when you are on the road.

Competitive advantages

- Access customer's desktops
- Access your machines when away
- Demonstrations
- Training

Some best of class tools

1. Remote Desktop: Built into Windows 2000 and later, available from Microsoft for other platforms.
2. PcAnywhere: http://www.symantec.com/home_homeoffice/products/remote_pc_fax/pca115/index.html
3. VNC: http://www.realvnc.com/
4. Copilot: https://www.copilot.com/learnmore/
5. GotoMyPC: https://www.gotomypc.com/
6. LogMeIn: http://logmein.com

Comments from developers

1. None at this time

Online Collaboration

Developers who need to collaborate with team members and customers can get together in a meeting room and hash things out with a whiteboard, some dry erase pens and open discussion. What happens when the people involved cannot get together in one location? This is where online collaboration tools come in and people connect virtually.

Competitive advantages

- Work with teams (developers and clients) for collaboration
- Reduced travel costs
- Adhoc scheduling

Some best of class tools

1. GotoMeeting: https://www.gotomeeting.com/s/gg/aw/gotomeeting
2. LiveMeeting: http://www.microsoft.com/office/uc/livemeeting/default.mspx
3. WebEx: http://www.webex.com/
4. NetMeeting: http://www.microsoft.com/windows/netmeeting/

Comments from developers

1. None at this time

Desktop Searching

All computer users have a challenge of finding information they have stored somewhere on their computer, the network, and the Internet. The promise of information at your fingertips has long been promised, but with more and more digital information available users are finding it more difficult to locate the files they need, and the information contained in the files.

Competitive advantages

- True information at your finger tips
- Find information in any file no matter what the format
- Search files anywhere on the computer

Some best of class tools

1. dtSearch: http://www.dtsearch.com
2. Google Desktop: http://desktop.google.com/
3. Copernic Desktop Search: http://www.copernic.com/
4. Yahoo! Desktop Search: http://desktop.yahoo.com/

Comments from developers

1. Consume a lot of resources on the computer.

File Comparison

All developers have the need to compare the content of two files, the list of files in two different folders, and understand the common content and the content that is different. Many of us have done this manually for years without knowing the tools available to automate this process.

Competitive advantages

- Compare output during testing
- One great way to compare data is to output to XML and do text comparisons
- Synching source code during development

Some best of class tools

1. Beyond Compare: http://www.scootersoftware.com/
2. Compare It: http://www.grigsoft.com/wincmp3.htm
3. Ultra Compare: http://www.ultraedit.com/index.php?name=Content&pa=showpage&pid=109
4. WinMerge: http://winmerge.sourceforge.net/

Comments from developers

1. I use Beyond Compare all the time to synchronize source code, move files from one folder to another, and to integrate it with source code control to review history.
2. Each and every time this session has been presented; Beyond Compare is used by the majority of developers.

Blogging and Podcasting

Blogging and podcasting are natural ways for developers to share information with other developers and even your customers.

Competitive advantages

- Marketing
- Community building
- Sharing

Some best of class tools

1. Pick a host, pick a tool, just do it

Comments from developers

1. The more people blog, the more good information is available to the community.
2. Too much to read, too little time.
3. I have new business opportunities more frequently since I have started blogging. I know I have signed new business as a direct result of blogging.

Special thanks

This session was first presented to a group of Fox developers in Toledo as part of an open discussion on tools used in our businesses. The participation by this group was invaluable in assembling this session and the materials present. I want to thank Sharon Hodgson , Mary Pilon, and Rick Borup who took excellent notes during the session and were willing to share them with me so I could assemble this whitepaper.

Conclusion/Summary

This session is designed and focused on providing food for thought about the tools supporting and enhancing your ability to be fast to market, and best in class. I hope this session exposed and explored tools useful in supporting the business of developing software.

Updates and corrections to this document can be found on Hentzenwerke’s Web site, www.hentzenwerke.com. Click “Catalog” and navigate to the page for this book.