The PowerWorld Development Process

- Jamie Weber, Ph.D.
  - Director of Operations, PowerWorld Corp.
    - weber@ powerworld.com
    - (217) 384-6330, ext 13

1

Topics

- Software Development Process that PowerWorld Corporation Uses
  - Extreme Programming
- PowerWorld Development Team
- Tools Used by PowerWorld
  - Borland Delphi
  - Code Management System
- Patch Versions
- How PowerWorld decides what new features to develop

2

Software Development Process: Waterfall Process

- What you learned in school was the "Waterfall Process"
  - http://en.wikipedia.org/wiki/Waterfall_model
  - Customer States Requirements
  - Design for Requirements
  - Implementation (coding)
  - Verification (testing)
  - Maintenance

3

What's wrong with the Waterfall Process

- Requirements are usually constantly changing
  - You end up writing software that solves the problem you *thought* you had instead of what you end up really having.
- Customer is only involved up front – doesn't allow them to guide changes in requirements
- Part of the design may be too difficult to implement requiring a reconsideration of the design
  - Or design may just be too costly
- If you wait until the end to test, then you might spend a huge amount of resources on software that has no hope of working
  - Incremental development builds confidence that the software design will actually work
  - Incremental development allows you to abandon a bad design early on

4

What PowerWorld Uses: Extreme Programming (XP)

- This is a real development process with years of history and corporate research behind it
  - http://www.extremeprogramming.org
  - http://en.wikipedia.org/wiki/Extreme_Programming
  - Process is possible because of object-oriented programming
    - (software can be written and updated in pieces)
- Extreme Programming Values
  - Communication
  - Simplicity
  - Feedback
  - Courage

5

Extreme Programming: Communication

- Communicate with the customer in all parts of the process
  - Developers must understand what the customer is trying to do, not just what the software does
- Developers also must communicate extensively with one another
  - More than one developer is familiar with each part of the software
    - More than one person can help with bug fixes and support
    - Employee turnover and retirements won't kill the software
  - Developers have a shared vision

6

Extreme Programming: Simplicity

- Implement software in the simplest manner initially.
  - Might not be the optimal approach initially, but software can be improved later
  - Prevents you from wasting time optimizes parts of the software that are not important
  - Customer will communicate with feedback what needs to work better
  - Makes is so all developers can understand most of the software

7

Extreme Programming: Feedback

- Send software updates frequently to customer and get feedback on functionality as well as design
- Give feedback to the customer about estimates of how much time component tasks are going to take
- Feedback from the system by testing the software

8

Extreme Programming: Courage

- Have courage to *refactor* the code
  - Completely rewrite a section of code so that it can be better modified in the future
  - This goes hand-in-hand with Simplicity because you may need to rewrite some old code to improve performance
- Know when to throw-away your code
  - Just because you spent time on some code doesn't mean it's worth keeping

9

PowerWorld Software Development Team – 8 permanent people

- Tom Overbye, Ph.D. (1994 and before)
  - Company founder and a professor at University of Illinois
- Jamie Weber, Ph.D. (January 1997)
  - Director of Operations
- Kollin Patten, M.S.E.E. (May 1997)
  - Director of Engineering
- Ray Klump, Ph.D. (October 1997)
  - Half-Time Employee (also professor at Lewis University)
- Santiago Grijalva, Ph.D. (May 2001)
- Oscar Munoz, M.S.E.E. (August 2002)
- Kyle Johnson, B.S. C.S. (November 2003)
- Caroline Marzinzik, M.S.E.E. (October 2004)

10

PowerWorld Corporation Business and Consulting Team

- Mark Laufenberg, Ph.D. (July 1996)
  - President
- Kelley Shimmin (May 1999)
- Scott Dahman, M.B.A, M.S.E.E (August 2003)
  - Director of Business Development
- Interns that have participated in software development
  - Craig Martini (99), Yan Sun(02-03), Matt Davis (03-05), Zeb Tate (04-06), Dave Savageau(04-05), Steven Judd (06)

11

PowerWorld Software Team Support

- All developers are involved in all parts of the software process
  - Give Training
  - Write Software
  - Provide email and phone support
  - Write Help Documentation
  - Communicate with customers regarding potential new features

12

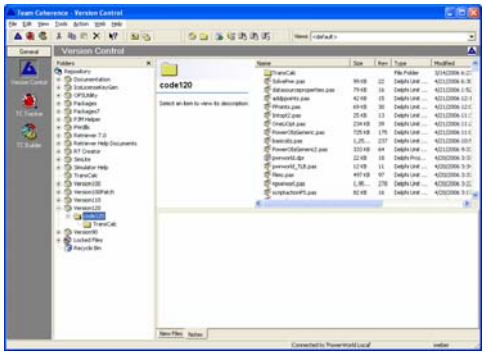
Tools used by PowerWorld: Development Language

- Development Language
  - Borland Delphi 7.0 presently
    - Plan to switch to Delphi 2005 this year
    - Visual, Object-Oriented, windows development language
  - Most of code is written in-house
  - Make use of third-party, royalty-free software when appropriate
    - Toolbars and Menus
    - Movie Maker
    - HTTP, HTTPS, FTP, FTPS Connections
    - Lock and Key Software
    - UCalc32 DLL (parses expressions)
    - Zip/Unzip DLLs

13

Tools used by PowerWorld: Code Management System

- Team Coherence Version Manager
  - Made by Quality Software Components (QSC)
  - http://www.qsc.co.uk



14

Tools used by PowerWorld: Code Management System

- Allows all developers to keep their code synchronized
- In order to edit a file a developer must check it out
  - This locks in down so other developers can not edit it
- When finished editing, code is checked back in
  - Code is always in a state that can be compiled
- Maintains the revision history of all source files
- Allows remote users to work on the same files

15

PowerWorld Patch Maintenance

- When a new version of the software is released we create a new “project” in the Team Coherence Software
  - This means that we have two sets of source code that must be maintained
  - Future development is performed on the new project
  - Bug fixes are made to both projects
  - Simple new features may be added to both projects
    - Allows us to get new features out quickly
    - Features which effect the file format can not be added.

16

Putting out a new patch version

- When a new patch version is necessary, a label is defined in the Team Coherence tool and the present state of the patch source code is marked with the date of the patch
- New patch versions are compiled
- Patches are placed on our website
- Website is updated to reflect the changes that have been made to the patch
- This occurs on the order of once a week

17

How PowerWorld decides what new features to develop

- Bug fixes come first
  - New development must wait for bug fixes to be complete
  - Bug fixes are sent out via the patch website
- Extreme Programming Rule – Customer Feedback
  - When a few customers request a simple change it will be made immediately
- For Larger software modifications
  - Market Need
    - Is there a big enough market to support the development costs of the software tool?
  - customer cost-sharing and testing arrangement is typical
    - Historically, PowerWorld has between 2 and 4 cost-sharing arrangements going on at all times

18

What the Customer gets from a Cost-Sharing Arrangement

- Work with a company with a proven track-record of providing software that meets their customers needs and supports the software after completion
- Get to drive the development of an important software tool using outside software developers
  - More cost-effective than writing custom software
  - Customizations integrated in Simulator are maintained and supported by PowerWorld going forward
  - Software is tested by the PowerWorld User-Community as a whole
- Customers' staff is able to become intimately aware of the functionality of PowerWorld
- Minor requests from cost-sharing clients are typically added at no cost

19

What PowerWorld get from a Cost-Sharing Arrangement

- Extreme Programming Principles
  - Customer is intimately involved in designing the software needs
  - Customer contributes their own staff time to providing feedback and testing
  - PowerWorld staff learns tremendously from the customer interaction
- Defrays the development cost
  - Not looking to recoup all our costs
- Proves to PowerWorld's management that the enhancement meets a market need

20

Brief History of Cost-Sharing Development

- Development of Visualization Software
  - NSF Small Business Innovative Research Grants (1997 – 2003)
- OPF and SCOPF Development
  - US DOE – EIA (1998 – 2002)
  - Tennessee Valley Authority (2001 – 2004)
  - Dynegy (2002)
  - NSF SBIR Grant for OPF enhancements
- PVQV, ATC, Contingency Analysis, Misc Development
  - Bonneville Power Administration (1999 – present)
  - NSF SBIR for PVQV tool
- SimAuto Development
  - Enron (2000)
- Retriever Development
  - Tennessee Valley Authority (2000 – present)
  - Commonwealth Edison (1999)
  - PJM Interconnection (2003 – 2005)
  - ISO – New England (2004 – present)
  - SPP (2005 – present)
  - NSF Small Business Innovative Research Grant (2003 – 2005)
- Geographic Information System Support, Stability Data, Line Impedance Calculator
  - Defense Intelligence Agency (2004 – present)

21

Present On-Going Software Development Contracts

- Bonneville Power Administration
  - GE File Support and Various Miscellaneous tasks
  - Technical Documentation
- Defense Intelligence Agency
  - GIS capabilities
  - Stability Data File Support (brings all their data into one graphical environment)
- Tennessee Valley Authority
  - Retriever modifications
- ISO – New England
  - Retriever modifications
  - Real-Time Models for Real-Time Analysis

22

PowerWorld Corporation Development Process Summary

- Flexible Development
- All developers are involved in every facet of the process
  - Ensures that knowledge is not locked in one employee
  - Clients can talk to any of us and get help
- Customers drive the development
  - Customer feedback is vitally important to us
  - Call us: (217) 384-6330
  - Email us: support@powerworld.com

23