

# Software Development Services from PowerWorld

---



Jamie Weber  
weber@powerworld.com



**PowerWorld**  
Corporation

2001 South First Street  
Champaign, Illinois 61820  
+1 (217) 384.6330

support@powerworld.com  
<http://www.powerworld.com>

## Vital First Question:



How does a *successful* software development project happen?

# The PowerWorld Philosophy

---



- An overriding company goal has been to work closely with our customers to provide the best possible power system visualization and analysis software, and top-notch customer service.
- As a small company when somebody calls or emails PowerWorld they quickly get to interact with the people doing the actual software design and development.
- Having great employees definitely helps!

# PowerWorld Software Development Team

## – 9 permanent people

---



- Tom Overbye, Ph.D. (1994 and before)
  - Company founder and a professor at University of Illinois
- Jamie Weber, Ph.D. (1997)
  - Director of Operations/Software Development
- Kollin Patten, M.S.E.E. (1997)
  - Director of Engineering
- Kyle Johnson, B.S. C.S. (2003)
- Caroline Marzinzik, M.S.E.E. (2004)
- Matt Davis, Ph.D. (2009, part-time 2003)
- Angel Aquino-Lugo, Ph.D. (2010, part-time 2007)
- Thomas Nicol, M.S.E.E. / B.S. C.S. (2011)
- Kate Davis, Ph.D. (2011 (part-time), part-time since 2007)
- Saurav Mohapatra, Ph.D. (August 2015, part-time 2012)

# PowerWorld Corporation Business and Consulting Team

---



- Mark Laufenberg, Ph.D. (1996)
  - President
- Kelley Wegeng (1999)
- Scott Dahman, M.B.A, M.S.E.E (2003)
  - Director of Business Development
- Nina Chanlin (2014, part-time 2012)
- Santiago Grijalva, Ph.D. (2001 - 2009)
  - Left in August 2009 to be a tenured professor at Georgia Tech

# PowerWorld Software Team Support

---



- All developers are involved in all parts of the software process
  - Write software
  - Provide email and phone support
  - Give training
  - Write help documentation
  - Communicate with customers regarding potential new features

# 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

# Traditional Software Development

---



- What is traditionally done is the “Waterfall Process”
  - [http://en.wikipedia.org/wiki/Waterfall\\_model](http://en.wikipedia.org/wiki/Waterfall_model)
  - Customer States Requirements
  - Design for Requirements
  - Implementation (coding)
  - Verification (testing)
  - Maintenance



# 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

# What PowerWorld Uses: Agile Programming

---



- This is a real development process with years of history and corporate research behind it
  - <http://agilemanifesto.org/>
  - [http://en.wikipedia.org/wiki/Agile\\_software\\_development](http://en.wikipedia.org/wiki/Agile_software_development)
  - Process is possible because of object-oriented programming
    - software can be written and updated in pieces
- Agile Programming Values
  - Communication
  - Simplicity
  - Feedback
  - Courage

# Agile 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

# Agile 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

# Agile 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

# Agile 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

# 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
- Agile 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
    - Range of these projects is \$1,000 - \$400,000
    - Most projects are \$20,000 - \$40,000 (a couple man-months)

# 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



# What PowerWorld Gets From a Cost-Sharing Arrangement

---



- Agile 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 that the enhancement meets a market need

# We're Available

---



- Consider us your external “in-house” software development group