

## *What your Agile Friends Won't Tell You ...*

Walt Washburn, PMP

September 17, 2009

Theory attests Requirements Management (RM) methodologies significantly reduce cost and risk. However, contemporary interest in agile development is causing RM systems, methodology, and experience to wane. There is a place for agile techniques, but there is also great opportunity and return on your invested effort to deploy RM.

### **Agile Challenges**

Challenging the headlong embrace of agile techniques are:

- outsourcing;
- external service providers;
- distributed teams; and
- partnerships between companies or corporate divisions where a broad spectrum of stakeholder needs must be satisfied.

Companies that partner with other development shops, whether in country or off-shore, must deal with a significant layer of complexity not easily addressed by agile techniques. If you find that your key stakeholders are in differing or geographically separate business units, then your challenges to manage expectations, assure accurate and timely communications, and maintain consensus are daunting - especially if these stakeholders have incompatible interests, demands, or priorities. These scenarios are not hypothetical.

A development team can not pursue requirements while constructing software when they must build applications concurrently with several other departments and then release to production together. It is not unusual for web applications on UNIX mid-range platforms to inter-operate with one or more mainframe based back-ends, and use messaging layers and shared business objects. Some or all of these platforms will release components of a strategic application together. Success is dependent upon tight control and management of requirements across all platforms.

Agile methods work well when applications are developed for specific, homogeneous stakeholders - situations where the same group of people conceive, specify, and use the application. Good candidates for agile methods will have very loose coupling, or lack coupling to other systems. Applications simple enough to document architectural, structural, and algorithmic essentials as comments in the code are candidates. Applications unlikely to experience growth in scope, complexity or functionality might be ones to consider. In such circumstances, build iterations of prototypes and deliver incremental functionality until everyone is happy. It's the Agile Way.

For the rest of us - those who are maintaining complex web applications with content management, multiple layers of commerce components, portal delivery, with middle-tier shared components, and adjacent platforms... For those that must satisfy diverse business interests in a resource-constrained environment - building without a requirements management methodology and process ... well, it's like skiing naked -- fast, exhilarating, and lots of fun ... until the inevitable happens.

Protect your development team with a web-delivered, database-driven requirements management (RM) system.

## **RM Benefits**

The benefits from using a requirements management database application are many. Consider the following as the low-hanging fruit -- immediately achievable to your organization:

- The RM system supports collaboration across communities of interest, geography, and time zones.
- It can provide a secure communications platform with outsource partners and service providers.
- Web-based systems operating on industry-standard ports and http protocol are usually not stymied by corporate firewalls and security policies, or are easily proxied for those environments that are seriously locked-down.
- The web-centric RM system allows varied stakeholders such as internal Business Analysts (BA) and Subject Matter Experts (SME) to specify and define systems and components that others with less domain knowledge and more geographic separation (think: off-shore) can help build.
- Web-based RM systems add structure to requirements documentation and can enforce consistency of data collected across diverse stakeholder communities or projects.
- RM systems capture and provide history of requirements development and change.
- RM systems "in the cloud" can offer the ability to store documents and screen shots with requirement records.
- They provide continuity in time and across time zones. Web-based systems help compensate when members of the team come and go, get reassigned, etc.
- RM systems enable bi-directional traceability - this allows the team to assess requirements coverage, of course, but also provide the ability to assess what functional elements are touched by any change in the body of requirements.
- RM systems often integrate other key supporting functions, like issue / risk / defect tracking or quality assurance testing with all the collaboration, geographic, and timezone leveling benefits already mentioned.
- Web-based systems allow automated monitoring and communication through email notifications as the distributed team works.

## **Return on Investment (Effort)**

Look into the following areas and activities for returns on your investment of effort:

- Stakeholder management and satisfaction through access to documentation, records keeping of approvals

and change history.

- Complexity Management by maintaining requirement-unique ids, tracking requirement dependencies and linkages, documenting baselines, and controlling change.
- Budget and Scope Management through change control, baselining, and release scheduling.
- Budget, Scope, and Complexity Management by specifically citing and controlling requirement priorities.
- Resource Management is enhanced by the release baselines developed against requirements sets. The documentation of such baselines and approvals across the various stakeholders involved makes public and accountable all such decisions.
- Version Control and Release Management is especially important when several development organizations deliver software in scheduled integrated software releases. Key in this more complex scenario is the ability to see how changes in requirements that occur in one platform drive technical complications or cost in another.
- Change Management: Web-based RM systems automatically capture and record change history and records of decisions and approvals as a natural process of using the application, obviating the impossible task of assuring individual adherence to stop and record such non-functional attributes as they change.
- Other non-functional attributes of requirements can be recorded and exploited (hierarchical relationships, dependencies, changes, links to other projects).

### **It Doesn't Have To Be Complex or Expensive**

Deploying such a tool does not have to be expensive or held hostage by higher priority demands on the infrastructure teams. Use commodity servers and open source applications!

One such package that meets all of the objectives mentioned in this article is rth, a PHP application that runs on the ubiquitous LAMP application stack (see <http://sourceforge.net/projects/rth/> ). The Open SuSE GNU-Linux distribution can be installed with full LAMP functionality right "out of the box" (see [http://en.opensuse.org/Welcome\\_to\\_openSUSE.org](http://en.opensuse.org/Welcome_to_openSUSE.org) )

You no longer have to be a Linux mechanic to deploy applications such as this. It is entirely feasible to experiment with such a system within a work group or department using a discarded server of even very modest capability. You can download, install, and configure rth in 20 minutes if you are experienced, but it should not take much more than an hour if you are not.

The rth project page on [SourceForge.net](http://SourceForge.net) points to a team blog which in turn points to a demonstration rth suite that allows you to try it out without any fuss of installing it for yourself.

rth allows teams to capture and manage requirements across application components and across application

releases. It provides version control, and files storage associated with requirements records. It supports assigning priorities to requirements, and tracks changes as requirements evolve. It supports requirement hierarchies, where requirements can be associated to other requirements, or to test sets providing the basis for measuring and reporting on requirements coverage in the testing phase. The application provides for change management at the requirements level, but also provides records keeping for requirements approvals and change locking.

This application integrates requirements capture and management with quality assurance testing and defect / issues tracking. It helps manage the test executions and test results at test time. Each of the functions can be configured to automatically email for status or edit changes in the records.

Of course there are a number of other open source and inexpensive, cloud-delivered solutions available as well. So what have you to lose but difficult requirements management challenges? You can launch a proof of concept skunk-works project that will cost no more than your initiative and a couple of hours to find, build, and configure a work group server.