



Doing Agile Right

A White Paper by: Michael Sayko

January, 2010

Doing Agile Right

Copyright © 2010 Michael Sayko

All rights reserved. No part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, without the prior permission of the copyright owners.

Doing Agile Right

Agile development is no longer an emerging technology. From small software vendors to large IT organizations, Agile is now the preferred software development methodology. As development teams embrace Agile, they deal with new concepts, like user stories and the backlog. While mastering these concepts, developers work closely with end customers to translate evolving requirements into functional software. Over time, development teams recognize that doing Agile right involves more than applying new concepts. Doing Agile right means having the right mindset.

Advocates of Agile may assert the need to value individuals and interactions over processes and tools. However, adopters of Agile recognize this does not mean forgoing tools that enable productivity. Instead, effective Agile teams embrace straightforward tools that help them deliver working software. These clear-cut tools not only support the delivery of functional software at the end of short iterations, but they also manage the user stories and tasks needed to produce that software.

The Need Filled by AgileCycle

Unfortunately, tools that support different aspects of Agile development tend to be used as standalone products, or with limited integration. When used in isolation, no tool supports the entire Agile development lifecycle. AgileCycle tackles this problem by providing an end-to-end view of user stories from their creation, to their implementation in code, to their realization as working software. Not only does AgileCycle facilitate the planning and execution of Agile projects, but it manages the code developed for each iteration, the builds created from that code, and the deployments of builds to test and production environments.

AgileCycle is able to manage the complete Agile development lifecycle because it integrates best of breed products that collectively support all aspects of the lifecycle. In particular, AgileCycle integrates:

- Rally for Agile lifecycle management,
- AccuRev for software configuration management, and
- AnthillPro for build and deployment process automation.

Doing Agile Right

The integration provided by AgileCycle is administered from a dashboard that ties together Rally, AccuRev, and AnthillPro.

Rally Manages the Agile Lifecycle

By all measures, Rally is the industry leading Agile lifecycle management tool. 2,000 customers, 54,000 projects, and 86,000 users manage their Agile projects with Rally. Rally was designed from the start to support Agile practices. User stories, tasks, iterations, and backlogs are concepts that are essential to Agile projects. Yet, traditional project management tools are not designed to support these concepts.



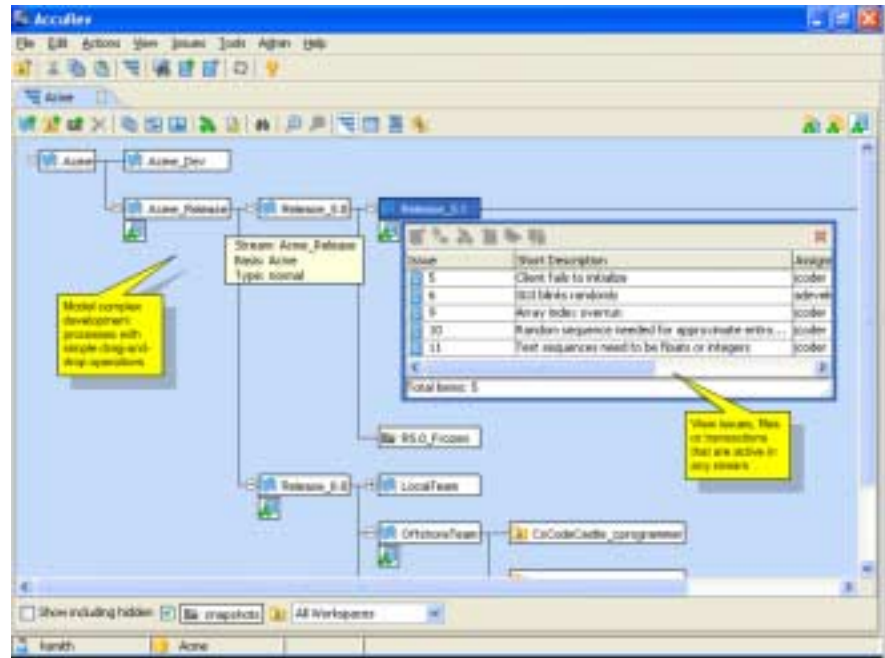
Rather than mapping Agile concepts into the traditional project management paradigm, Rally models Agile concepts the way they were intended to be used. With a customizable dashboard, Rally provides project management, time tracking, requirements management, test management, and defect management for Agile projects.

AccuRev Enables Agile Development

AccuRev is the only modern software configuration management tool designed to address the flexibility and dynamic nature of process change that is part of being Agile. With a stream-based architecture, version controlled private workspaces, and change packages integrated

Doing Agile Right

with issue tracking, AccuRev provides a natural model for Agile development.

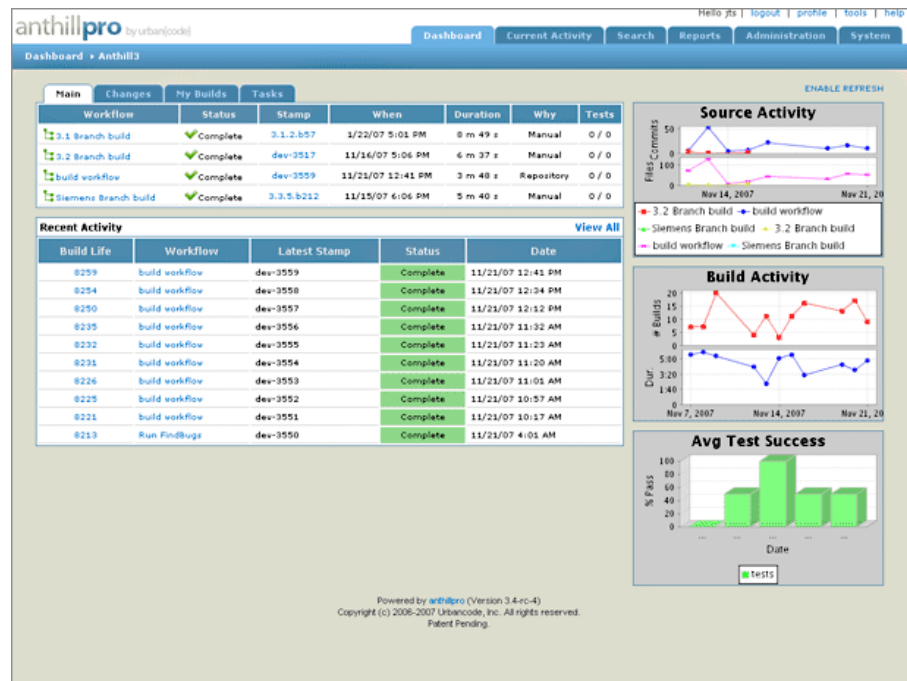


Additionally, atomic transactions and an append only database ensure a consistent and immutable audit trail for all operations.

AnthillPro Automates Build & Deployments for Agile Projects

Automation is essential for Agile projects, especially the steps to build and deploy the software application. Now a third generation product, AnthillPro established the standard for build and deployment process automation. AnthillPro models build and deployment steps in jobs. Jobs are embedded in a workflow, which is AnthillPro's unit of automation. The workflow to create a build can be started by the click of a button on the AnthillPro dashboard, or from a trigger that fires based on a schedule. Build artifacts created from the workflow are stored in an artifact repository so that they can be used to deploy to test and production environments.

Doing Agile Right



Central to AnthillPro is the concept of a build life. AnthillPro creates one build that can be deployed to different environments from test through production.

AgileCycle Supports the Iteration

Rally, AccuRev, and AnthillPro are the foundation of AgileCycle. The integration that is AgileCycle pulls together Rally, AccuRev, and AnthillPro to support the concepts that comprise the Agile lifecycle. Perhaps the most important Agile concept supported by AgileCycle is the iteration. More than any other concept, the short iteration as a full lifecycle is what that distinguishes Agile from other methodologies.

AgileCycle is built around the iteration. Starting with Rally, you capture requirements as user stories. You plan iterations by prioritizing stories and refining them into tasks. You track the progress of each story through the lifecycle of the iteration.

While Rally manages the work planned for each iteration, AccuRev isolates the code developed for the iteration. AgileCycle maps each iteration in Rally to a stream in AccuRev. Your developers use version controlled private workspaces in AccuRev to work tasks allocated to the iteration. These workspaces back to the stream created specifically for the iteration. As the developers make progress on tasks, they promote code from their workspaces to the backing stream. AccuRev's

integrated issue tracking system and change packages track the code changes at the granularity of a task. Each change package identifies the versions of files created to work the task.

As the developers promote change packages to the AccuRev stream that corresponds to the iteration, AnthillPro monitors the stream and triggers builds. AnthillPro creates both continuous integration (CI) and production candidate builds. The CI builds verify that the application compiles after a change package is promoted to the backing stream for the iteration. Production candidate builds can be scheduled to run daily to integrate changes. Each production candidate build is identified by a snapshot in AccuRev, and is stored in the AnthillPro artifact repository. The build is automatically deployed to a test environment where an automated test suite can be executed. If the build passes tests in the integration test environment it can be promoted to higher environments, including the production environment.

AgileCycle Provides an End-to-End View of User Stories

The purpose of an iteration in the Agile lifecycle is to translate customer requirements into working software. The user story captures a requirement. The user story and associated tasks can be used to track progress to meet the requirement. AgileCycle extends the traceability of the user story beyond the code to the iteration and the specific build that implements the story. Additionally, AgileCycle identifies the environments in which the build is running so that the user story can be validated by testing the build.

AgileCycle provides the end-to-end view of user stories by connecting:

- the story in Rally to
- the iteration in Rally to
- the stream in AccuRev to
- the change package in AccuRev to
- the build in AnthillPro to
- the environment in AnthillPro where the working software is deployed.

Additionally, your development process can be customized through the stream hierarchy in AccuRev. When different streams are used to model stages in your development process you can determine the stage of each user story. You identify the stream in AccuRev with the code that traces back to the user story in Rally. You can also tie the stream

in AccuRev to the environment in AnthillPro. Running in the AnthillPro environment is the build that implements the user story.

AgileCycle Supports Change

Agile is about embracing change that is a natural part of software development. Unfortunately, traditional project management and software configuration management tools are not designed to support this type of change. For instance, Agile projects must be able to move code across iterations when work is retargeted. The project management tool needs to track the change in plans while the code is moved in the software configuration management tool.

AgileCycle supports the change that results from retargeting work. Stories and task in Rally can be moved from one iteration to another. Likewise, the code in progress for the story can be moved from one AccuRev stream to another. Both Rally and AccuRev support the retargeting of work with minimal effort. Your team can focus on developing software rather than fighting the tools that manage your Agile software development process.

AgileCycle Combines the Capabilities of 3 Products into 1

AgileCycle combines the capabilities of Rally, AccuRev, AnthillPro, and their integrations into a single product. The product is sold by AccuRev for a one price. The cost of AgileCycle includes a perpetual license for AccuRev and AnthillPro along with the first year of maintenance. Included in the price of AgileCycle is the first year subscription to Rally. AccuRev provides installation, training, and support for AgileCycle.

About the Author

Michael Sayko is a software configuration management consultant based in Austin, Texas. He is experienced with parallel development, continuous integration (CI), and build and deployment automation of Java EE applications. Michael uses AccuRev and AnthillPro regularly in his work as a build and release engineer. You can reach him by email at mss@acm.org.