CASE STUDY

NETWORKEDASSETS

ZEROTURNAROUND

NetworkedAssets

NetworkedAssets is a seven year old company founded in Berlin with 26 employees working in Germany, Austria, Switzerland and Poland, of which 22 are developers (at the time of writing). The company focus is on the integration of processes and data typically for telecoms companies who own large IP networks (by large we mean large... > 1 million users on a network). NetworkedAssets are also Expert Partner of Atlassian.

The software house develops on Java and the JVM, mostly, based on Unix and Linux platforms creating an application solution which includes network management and monitoring, provisioning network resources, distributing resources over access points, automation of network management tasks, on demand IPTV solutions and more.

The development team wanted to streamline everything they do to be a more efficient and productive team. They work as a distributed agile team using the scrum template with two week iterations. Continuous Integration is crucial to the team who use Hudson (historical decision) to build the application from scratch and run unit tests after each commit. They also have daily static tools which are run to monitor the code quality such as Sonar and Findbugs.

NETWORKEDASSETS

Let’s use JRebel

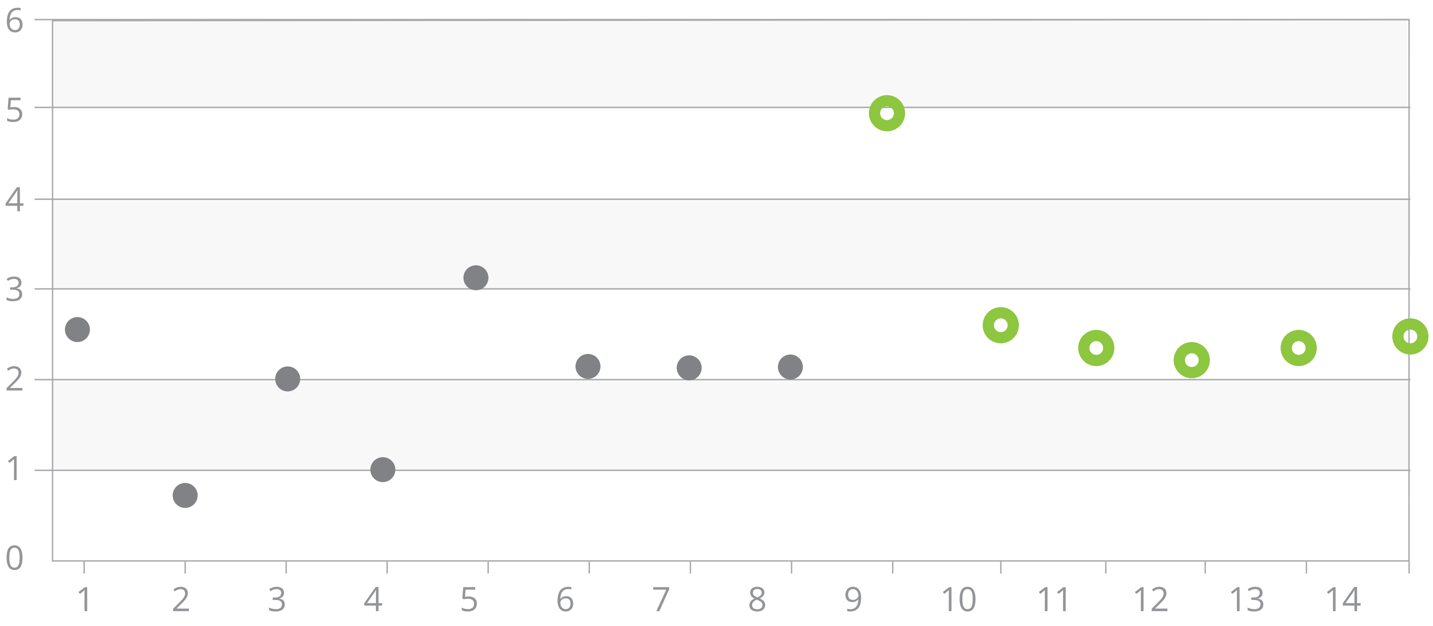
NetworkedAssets decided that JRebel was also a crucial part of their development process as their application redeploy time was not insignificant (typically around 4-5 minutes 2-3 times an hour), which adds up. The application contains Spring, Hibernate, Apache Camel, Guava and many more frameworks, which JRebel works with to enable the NetworkedAssets developers to change code or framework resources and instantly view and test their changes without the need for any delays.

The application developed in Eclipse and built with Maven (this is where JRebel integrates with the build directly) and also uses the Google Web Toolkit framework. The app is made up of four modules, each with a deployment unit plus many libraries, which sum up to around 100,000 lines of NetworkedAssets code and approx. a 100MB application including bundled libraries. The app is deployed and runs on Apache Tomcat, so as one of the more lightweight and faster application servers on the market, this isn’t a particularly slow environment.

Metrics & Measurements

NetworkedAssets decided to run some metrics on how much JRebel improved their development productivity so they compared the velocity across iterations where the devs were using JRebel and iterations where devs were not using JRebel. The team has performed eight Sprints without JRebel. After that the tool was installed and the six following Sprints took benefits from using it. The results even shocked the ZeroTurnaround team!

Team Velocity per sprint



	WITH JREBEL	WITHOUT JREBEL
Mean	2.77	1.97
Median	2.38	2.16
Std.Dev.	1.04	.74

The Math

Marian Jureczko, Senior Developer at NetworkedAssets did some science to prove the awesomeness coefficient.

In order to evaluate the significance of the difference a statistical hypothesis should be formulated:

H0 = **There is no greater Team Velocity in the 'With JRebel' period than in the 'Without JRebel' period.**

and an alternative hypothesis, that could be accepted when the statistical test gives reasons to reject the H0 hypothesis:

H1 = **There is greater Team Velocity in the 'With JRebel' period than in the 'Without JRebel' period.**

Before the test will be conducted we must chose the α (significance level). Typically values from .01 to .1 are selected. The smaller α value the greater possibility of not rejecting H0 hypothesis, even if the hypothesis is false. On the other hand, the greater α value the greater possibility of rejecting the H0 hypothesis, even if the hypothesis is true. In our case, we have social-related empirical data and small sample, hence it is reasonable to chose α = .1 (relatively big value).

The 1-tailed t-test was calculated using an on-line calculator (http://studentsttest.com/) and p = .05932 was obtained. The p value is smaller than the chosen α, hence we have reasons to reject the H0 and accept the alternative hypothesis - **There is greater Team Velocity in the 'With JRebel' period than in the 'Without JRebel' period.** In other words, JRebel significantly improves Team Velocity.

A **40% INCREASE** IN VELOCITY RESULTED WHEN JREBEL WAS USED WHICH LEADS US TO CONCLUDE AN **AWESOMENESS COEFFICIENT OF 1.**

WITH THANKS TO THE NETWORKEDASSETS TEAM WHO HELPED PUT THIS CASE STUDY TOGETHER.

Make sure you check out
the NetworkedAssets site -
www.networkedassets.com

Joerg Mueller-Kindt, CEO NetworkedAssets
Marian Jureczko, Senior Developer and Scrum Master NetworkedAssets
Andrzej Kozioł, Senior Developer, NetworkedAssets
Lo Voelsen, Marketing|Sales NetworkedAssets.