Success Story

I had TeamCity up and running & doing automated builds within 15 minutes after installing it. I was SHOCKED by how quickly I was able to get it up and running. The next day, I located a spare machine in the office and set up automated CI builds. We’ve been running TeamCity ever since.

Kelly Leahy, Architect, MG-ALFA system

TeamCity’s Benefits for Milliman

Awareness and increased quality
Everyone knows almost immediately when someone broke the build, and so does the ‘guilty party.’ MG-ALFA has achieved better check-in quality now that people are tired of getting those ‘compilation failed’ emails.

MG-ALFA also benefits from TeamCity’s ability to watch their source control server for updates, as well as its ability to run any arbitrary command line, NAnt, or MSBuild (or many more) task in response to those updates.

Results availability
There is always a source to go to for a current build – no need to get the sources and build on your own machine, or go ask a build master to get you a build.
MG-ALFA takes advantage of the fact that TeamCity can host build artifacts. This allows testers and other team members to obtain binaries whenever they like, and avoid any complicated deployment of files to an FTP site or some other hosting location.

“TeamCity gives us a peace of mind,” explains Leahy. “We know that the build is working and developers won’t be delayed (or stuck) as a result of getting the latest versions of the code from the source control system.”

**Load distribution**

The ability to set up different build agents is a nice bonus too. They are used to segment the calculation engine builds from the user interface builds, since the calculation engine builds routinely take 20-30 minutes for a build versus less than 5 minutes for UI builds. This way, the UI team gets very quick feedback without being stuck behind a CE build that takes very long to finish.