CASE STUDY

Etsy—Self-Service Developer Deployment: An Example of Continuous Deployment (2014)

Unlike at Facebook, where deployments are managed by release engineers, at Etsy deployments are performed by anyone who wants to perform a deployment, such as Development, Operations, or Infosec. The deployment process at Etsy became so safe and routine that new engineers could perform a production deployment on their first day at work. Etsy board members and even dogs have deployed to production![20]

As Noah Sussman, a test architect at Etsy, wrote, "By the time 8am rolls around on a normal business day, 15 or so people and dogs are starting to queue up, all of them expecting to collectively deploy up to 25 changesets before the day is done."[21]

Engineers who want to deploy their code first go to a chat room, where engineers add themselves to the deploy queue, see the deployment activity in progress, see who else is in the queue, broadcast their activities, and get help from other engineers when they need it. When it's an engineer's turn to deploy, they are notified in the chat room.[22]

The goal at Etsy has been to make it easy and safe to deploy into production with the fewest number of steps and the least amount of ceremony. Likely before the developer even checks in code, they will run on their workstation all 4,500 unit tests, which takes less than one minute. All calls to external systems, such as databases, have been stubbed out.[23]

After they check their changes into trunk in version control, over seven thousand automated trunk tests are instantly run on their continuous integration (CI) servers. Sussman writes, "Through trial-and-error, we've settled on about 11 min-

utes as the longest that the automated tests can run during a push. That leaves time to re-run the tests once during a deployment [if someone breaks something and needs to fix it], without going too far past the 20 minute time limit."[24]

If all the tests were run sequentially, Sussman states that "the 7,000 trunk tests would take about half an hour to execute. So we split these tests up into subsets, and distribute those onto the 10 machines in our Jenkins [CI] cluster. . . . Splitting up our test suite and running many tests in parallel, gives us the desired 11 minute runtime."[25]

The next tests to run are the *smoke tests*, which are system-level tests that run cURL to execute PHPUnit test cases. Following these tests, the functional tests are run, which execute end-to-end GUI-driven tests on a live server—this server is either their QA environment or staging environment (nicknamed "Princess"), which is actually a production server that has been taken out of rotation, ensuring that it exactly matches the production environment.

Once it is an engineer's turn to deploy, Erik Kastner writes, "you go to Deployinator [an internally developed tool, see Figure 12.4] and push the button to get it on QA. From there it visits Princess. . . . Then, when it's ready to go live, you hit the 'Prod' button and soon your code is live, and everyone in IRC [chat channel] knows who pushed what code, complete with a link to the diff. For anyone not on IRC, there's the email that everyone gets with the same information."[26]

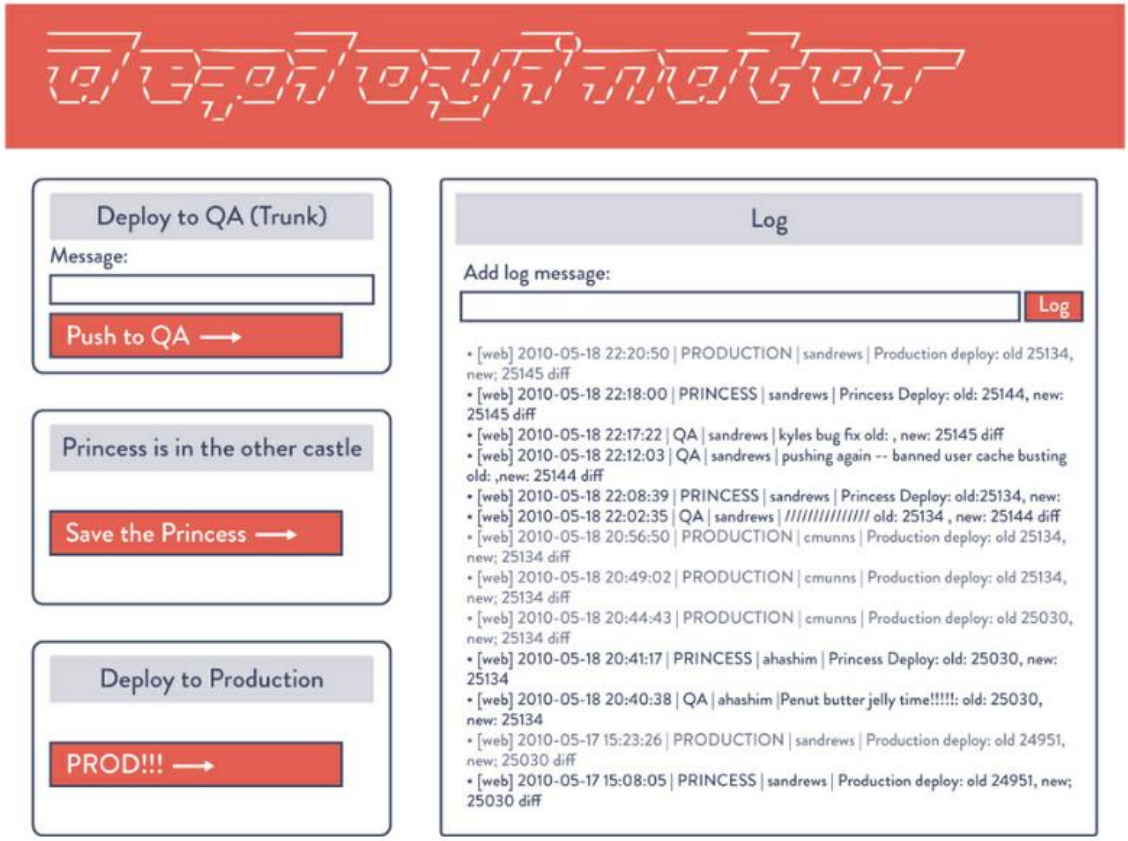

Figure 12.4: The Deployinator Console at Etsy

Source: Erik Kastner, "Quantum of Deployment," CodeasCraft.com, May 20, 2010, https://codeascraft.com/2010/05/20/quantum-of-deployment/.

In 2009, the deployment process at Etsy was a cause of stress and fear. By 2011, it had become a routine operation, happening twenty-five to fifty times per day, helping engineers get their code quickly into production, delivering value to their customers.

Automating the majority of the deployment process and including extensive automated testing created an easy