(Lack of) DevOps Case Study

Dev the developer, Tess the tester and Optimus from Operations were all involved in a project.

Dev wasn't sure where the latest code for the system was. Another developer emailed Dev a zip of source files and told him to use that. Still Dev couldn't get the system to work on his machine - every time he ran it there was a new dependency he hadn't known about that he also had to find, install and configure.

Other developers working on different parts of the site had finished their changes were keen to get them put live. "You can't release your stuff until mine's finished too," was a common phrase. So those developers started work on other features, until the other developers were done - but now their features were only part-done, and the whole cycle started again.

Tess performed her tests on the system in the staging area, but found many things were broken. "But it works on my machine, " said the developers, "and if I can't reproduce it, I can't fix it."

When developers did fix the bugs Tess found, she had to wait weeks before she could test them again. "Every time you want a new version for testing, we have to build and deploy it manually, and that takes a day out of our development schedule," complained the developers.

Finally the date set by Marketing for releasing the new system came round.

To deploy the new version of the application, Optimus had to shut down the live system. "No-one will be using it at this time," he thought as he killed the servers and overwrote the old files with the new ones. He saw some configuration files with different values in from before, but just copied them over anyway. Finally, after half an hour of frantic copying, he brought the site back up again.

Dev browsed the live web application, and clicked the link launching the feature he'd worked on. His excitement soon turned to horror when he realised he was looking at his old, buggy version of the page - where was the code he'd worked so hard to perfect?

Other people in the company were equally unimpressed. Many of the previously working parts of the website were now broken. Optimus was asked to put the old version back again. "Sorry, can't do that - I don't have a copy any more of the old version or its configuration settings." Until the developers could produce a new version, the site was going to stay like that.

Optimus received many complaints from the business that the performance of the live site was much worse than expected. Tess said that in testing, it had seemed OK. Optimus pointed out that the live environment had lots of extra firewalls and virus scanning, which the test environment didn't have, and that might explain the reduction in performance.

Due to a related news story, the site got mentioned on a national news report, sparking a huge number of unexpected visitors. But "Error 500 - server busy" was all that greeted them. "We've increased the server's memory and stuck another network card in, but the application was only designed to run on a single server, so there's nothing more we can do," said Optimus.

# Think about and discuss...

* What was going wrong on this project?
* What could people have done differently to improve things?
* Do you have experience / examples of other horror stories similar to this?