

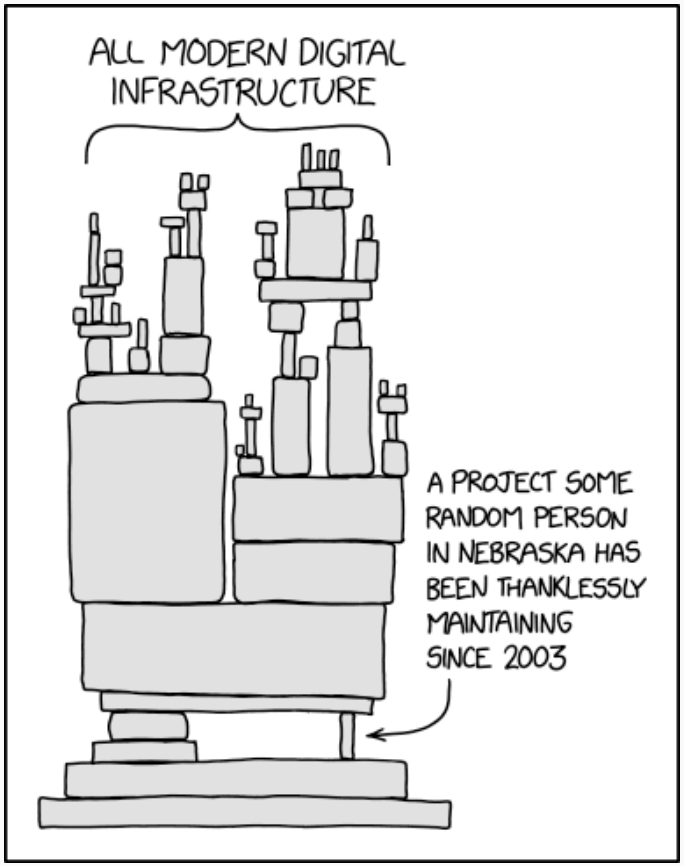

Adapting Telescope Operations to Science Platforms

How your organisation could benefit from a DevOps Roster

James Tocknell, Brent Miszalski and the full RDS team

Data Central, RDS, AAO, Macquarie University

Increasing support gap



XKCD 2347

- RDS/AAO increasing the number and complexity projects/systems supported, but hiring rules prevent directly creating more support/infrastructure specific roles
  - Initial mitigation was to use existing team members with the right skillset to plug gaps.
- Support burden falling on specific members of the team, reducing their effectiveness on other projects.
- Burnout a strong possibility

Solution: Create the DevOps Roster

- Heavily borrowed from the SALT (Southern African Large Telescope) Operations and User Support process
  - Focus on answering user helpdesk questions and daily tasks (e.g. data inspection, run pipelines, document issues, etc.)
  - Rotating roster to spread the load
- Started early this year (after various discussions about the support gap)

SALT (Wikipedia)

Objectives of the DevOps register

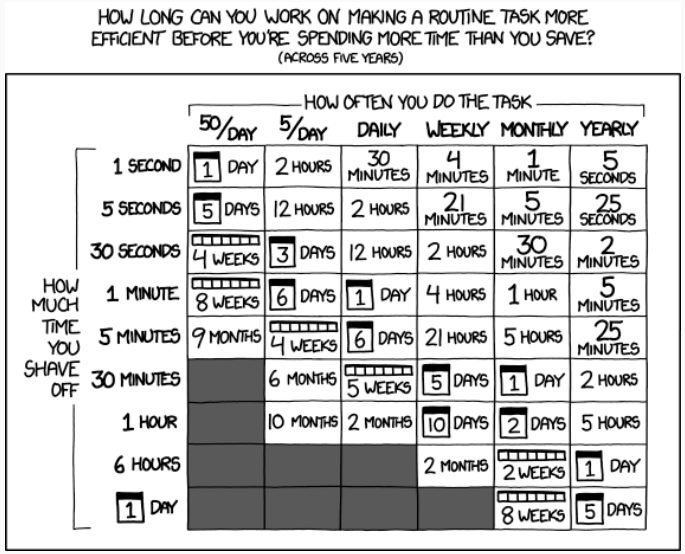
- Spread the support load widely to reduce burnout
- Document all existing systems
- Recognise and track system-wide support and maintenance (in the same way academics have service)

What is the DevOps Roster

- Weekly rotating roster of 2 people (chosen so there's a backup and to pass on advice/knowledge)
- Act as initial triage for user support, but also (as a lower priority role) complete tasks/tickets that have been put on the devops roster
- Pull in others to assist if needed
- Log what work was done, any issues seen, and any ongoing issues for next week's roster
- **Not an on-call roster**, not expecting those on it to work outside normal hours

Tasks associated with the DevOps roster

- These vary in both time and the skillset/knowledge required to complete the task
- Document how to handle common issues (e.g. forgotten usernames/passwords, switching emails)
- Minor changes to systems/develop a small web app to automate a task
- Maintenance of systems (e.g. software updates)
- Major changes/platform-wide automation



XKCD 1205

Tools and Processes

- Primary tools are Slack and Jira/Confluence, with a shared calendar for the roster
  - There are a number of Jira Helpdesks which are how users file tickets
  - A single Jira project for DevOps tasks (to ensure tasks are not forgotten)
  - A confluence space for documentation
  - Slack is used to coordinate, request assistance and ensure that the roster is running smoothly

Spreading knowledge

XKCD 1888

- All members of the DevOps roster are engineers, but have a variety of backgrounds and experiences
- Most members of the team were not familiar with all the services Data Central/RDS runs before starting the roster
  - Nearly all services now have multiple people who can fix issues
- Reduce risks around the loss of institutional knowledge
  - People can take leave and not worry about needing to be contactable
- Provides an additional path for mentoring junior members.

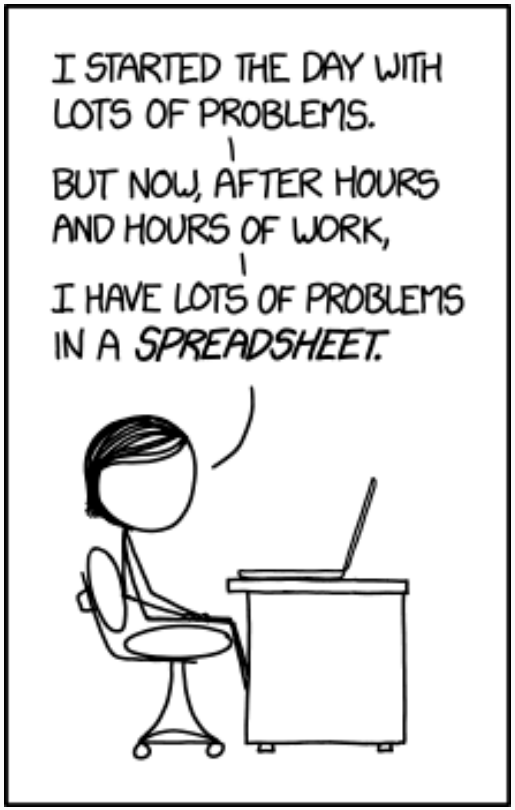
Applying the experience: moving campuses

- From July to November AAO moved from our old building (shared with AU Gov) to MQ Campus (3-5km).
- 2 major shutdown events (as servers were transported to MQ campus)
  - Those on the roster during those events handled user tickets around systems being down, allowing those moving servers/getting systems back online to focus on those activities.
- This wouldn't have been possible if those rostered on were unfamiliar with with the systems/services or how the user support system worked.
  - Able to identify if the system was planned to be down due to the move, or if it was unplanned.

Future of our DevOps roster

- This has been very successful for us
  - Response and resolutions times of tickets decreased
  - Internal tools developed to resolve common tasks (e.g. 2FA resets)
  - Continuous incremental progress on large tasks (e.g. only one system still needs manual certificate changes)
- No major changes planned for now
  - 2 people per week working
- Do a retrospective next year

How to create a DevOps Roster in your group



XKCD 1906

- Have buy-in from leadership
- Identify team members who should be on the roster:
  - Should be as wide as possible
  - Should include members of various skillsets and backgrounds
- Document how the roster works
  - Have a few people to create and update the roster
- Have a place for knowledge sharing

Questions and comments?

- Questions for the audience:
  - Do you or groups you know have a similar roster/support/maintenance system?
  - How do you balance the competing pressures of maintenance vs externally visible work?