



Extensible Platforms for eResearch

A case for building and using platforms in Research Software Engineer working groups

Samuel Bradley, John Hille, Dr. Pavel Golodoniuc, Dr. Andrew Rodger

Our software engineering groups quest for increased efficiency in research software and prototype delivery.

Challenges Faced by Research Software Groups

Research software engineers (RSEs) face several challenges that differ from those faced by counterparts developing commercial or in-house software.

Small Team Sizes

Projects often involve only 1 or 2 RSEs and rarely include support specialists such as a Business Analyst, Tester or DevOps Engineer. Low overall resources per solution.

Researcher Code

Much of an RSEs work involves modifying or integrating code developed by non-engineers or legacy code developed in older or unfamiliar languages.

Throwaway Code or The Next Big Thing?

Often the ongoing value of what an RSE team develops is not known until after either a considerable time has already been spent, or a solution is hacked together with no mind to future scalability or reuse.

High Variability of Work

Great for mental stimulation, bad for efficiency, collaboration and reuse.

A Rethink of How We Work

The first instinct when a loosely grouped team is faced with these challenges might be "every person or project for themselves" (see Fig. 1, Level 1), however the potential benefits of increasing the commonality of tooling, codebases and development practises can be huge.

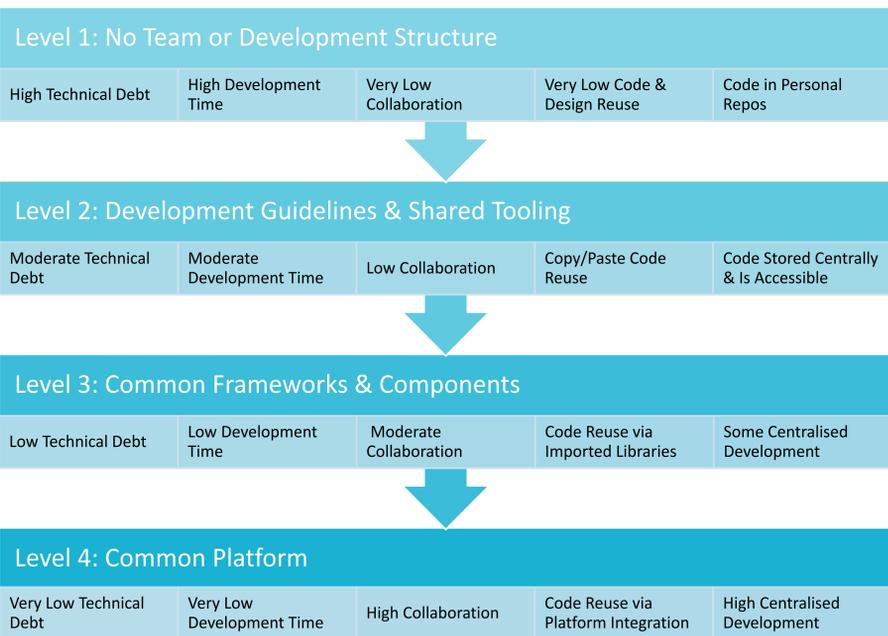


Figure 1: Levels and effects of structure in eResearch software development teams, based on internal experience.

Our Progress

The Discovery RSE team has successful long-term projects that have been operating at level 3 for many years. We have recently been applying these principles outside of individual projects to the broader team and seeking ways to go further...



As Australia's national science agency and innovation catalyst, CSIRO is solving the greatest challenges through innovative science and technology.

CSIRO. Unlocking a better future for everyone.

The Case for a Software Platform

For this purpose, a platform is a piece of software that offers core functionality needed by the RSE Team developed software. A platform is extended via separately developed modules.

The platform team shown below could be developing a new platform in-house or supporting an existing platform.

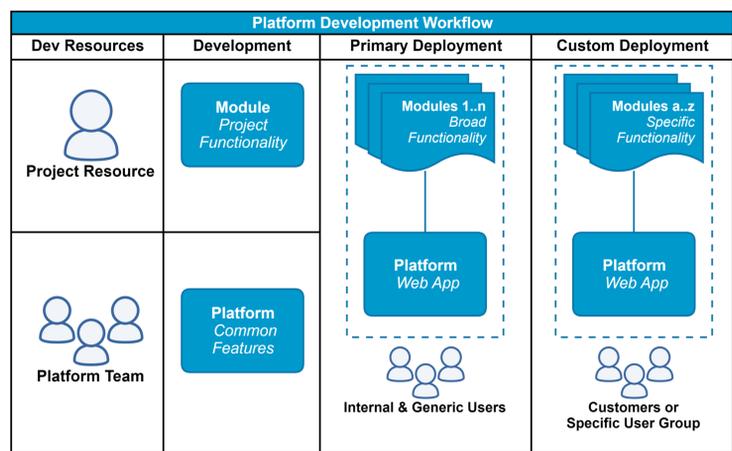
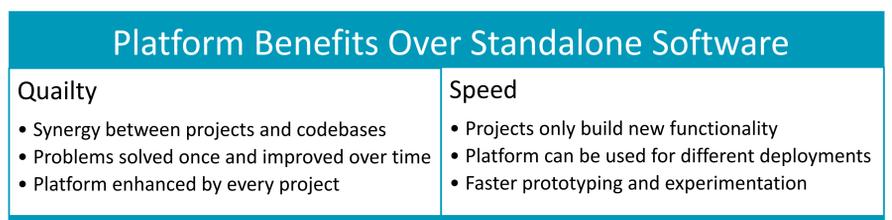


Figure 3: Platform Development Workflow, centralised development of common features, project functionality developed as modules. Modules added to primary deployment and/or bundled into a custom deployment to create a "product".



Scidra: Discovery's Future Software Platform

Scidra is in the early stages of development with these key design goals in mind:

Wrap researcher code don't rewrite it - Modules are lightweight wrappers over existing code, including legacy code. Minimises modifications to existing programs and code.

Minimise Custom Code - Don't repeat yourself, use existing components and generate code where possible.

Keep it simple and portable - Platforms of this nature can get incredibly complex, we intend to keep the core platform simple but flexible. Should run in any K8 environment.

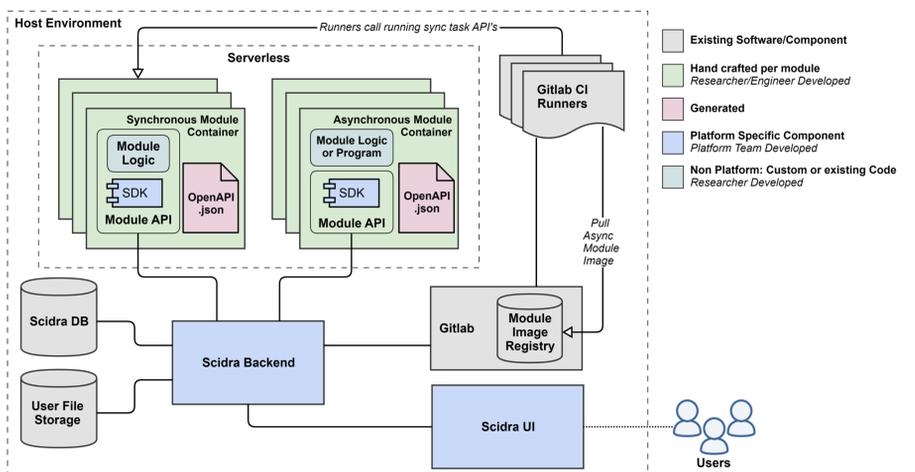


Figure 3: Scidra Architecture Diagram (Simplified)

FOR FURTHER INFORMATION

Samuel Bradley
Mineral Resources (Discovery)
sam.bradley@csiro.au
csiro.au/en/Research/MRF