ADMIRAL Project

Graham Klyne

Zoology Department

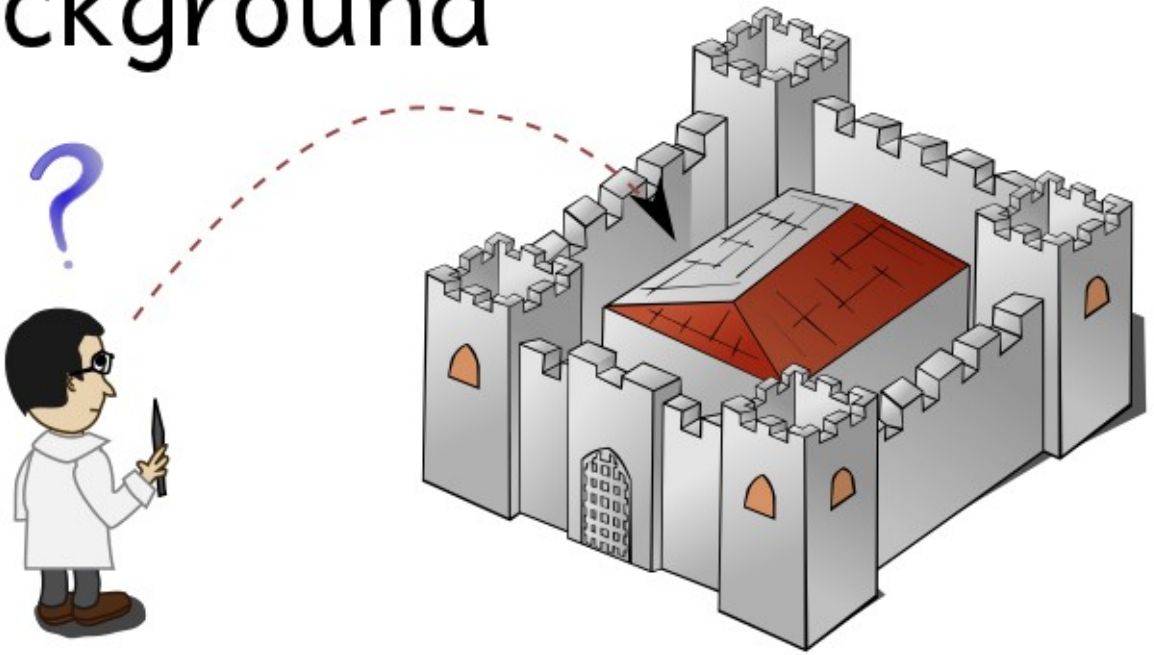
Oxford University

- Background (1 slide)
- Goals (1 slide)
- Technical gloss (1 slide)
- Achievements (1 slide)
- Missed achievements (1 slide)
- Lessons (the rest)

All clipart used is CC0, obtained from http://www.openclipart.org

Background

Empty repositories: we want to make it easy for researchers to lodge their data in repositories



Principles:

"Sheer curation" - dataset and metadata acquisition fitting into existing research practice

"Curation by addition" - take whatever we can get, and allow datasets to be incrementally improved until ready for publication

Goals

- Data management tools used by researchers
  - Easy for researchers to use
  - Serves as a bridge between current practice and a university data repository
- Data as first class output: citable by DOIs
- Supporting “long tail” small research groups
- Services for improving data annotations

ADMIRAL: Dramatis Machinae

Researcher's desktop - no special software

Admiral file sharing and data staging area

Oxford Bodleian Library Databank repository

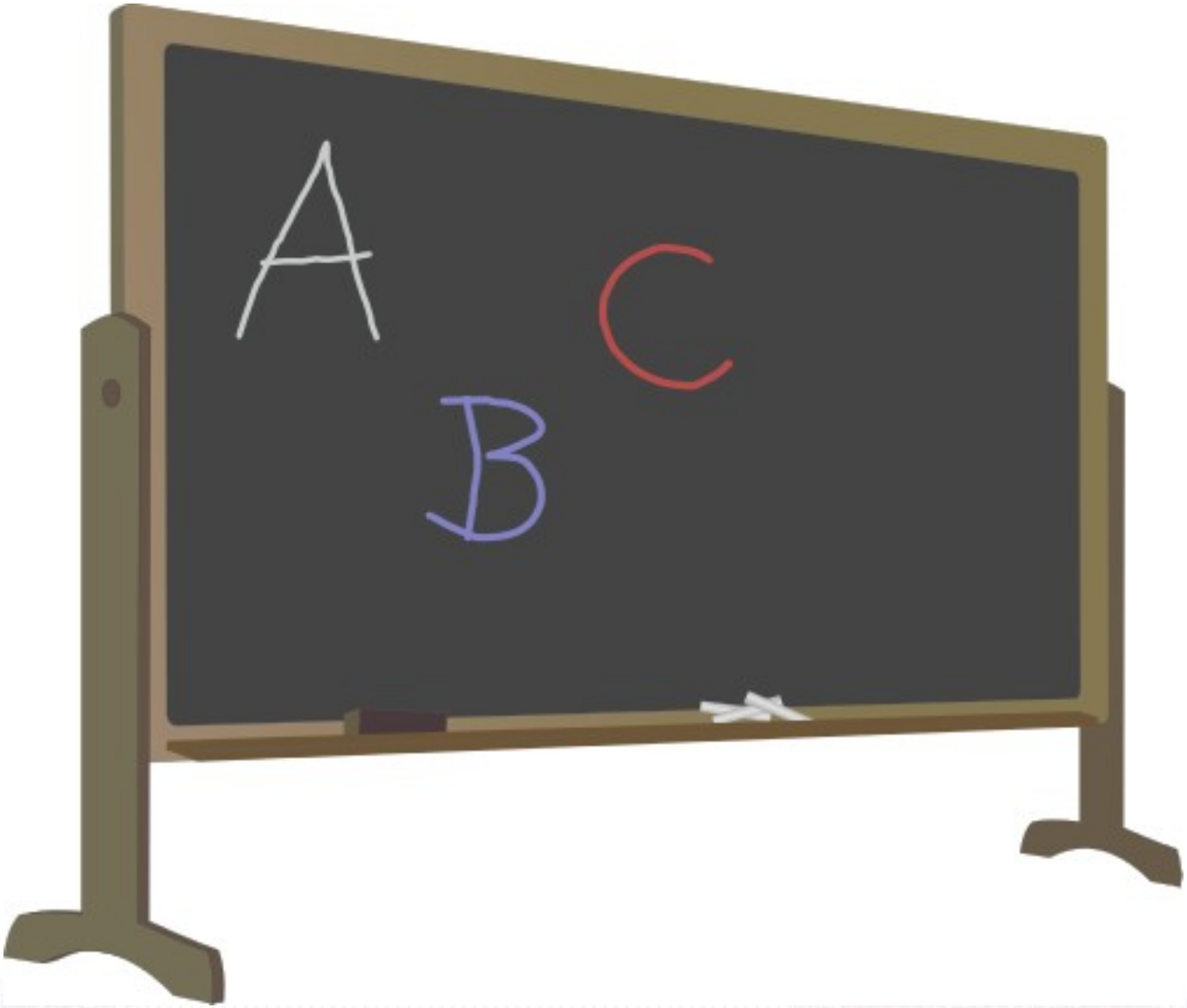
Achievements

- Simple file sharing platform with daily backups
- Used by researchers with no extra software
- Submission to data repository via web interface
- Staged model for data acquisition accepted
  - fits existing data collection and triage practices
- Engagement with departmental IT support
- Framework for assigning British Library Datacite DOIs is being implemented
  - late in the project, no published data DOIs yet

Missed achievements

- Inflexible dual access control (file and web)
- No University SSO integration
  - failure of existing desktop tools
- No meaningful progress on annotation tools
  - delays with core system
  - storage capacity was higher user priority
- Not yet achieved significant user engagement with data publication processes
  - delays to integration with Databank repository

Lessons



Less is more (work)

- Doing something lightweight and seamless with existing software components can be harder work than writing new software.
- But hopefully more sustainable.
- From the project proposal: *This project will depend heavily upon, and will bridge between, existing systems, reducing the development effort* – the last bit didn't quite work out that way!

Third party dependency risks

- Much effort was spent on getting third partyu software to behave (more or less) the way we (i.e. our researchers) wanted
- Access to deep expertise about the components used would probably cut development costs and improve outcomes
- In the project risk analysis, dependency on even very well established third party software did not fully recognize these difficulties

Working with users

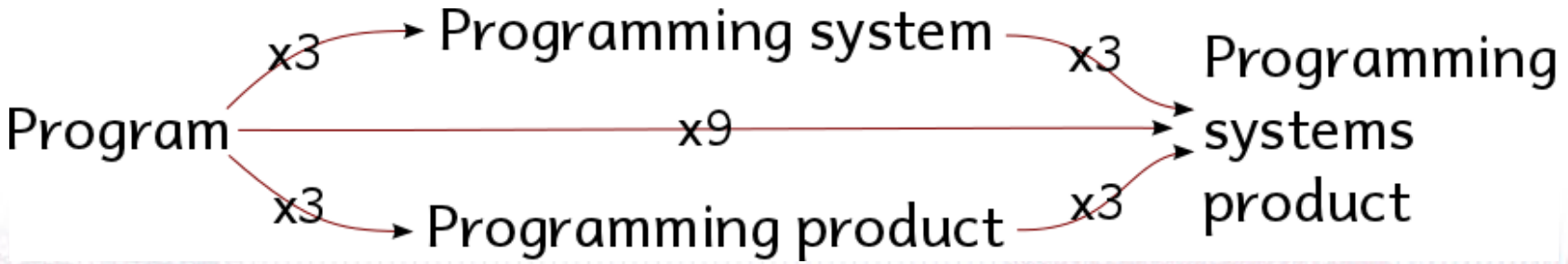
- Working with users takes time and effort. It can be hard to timeshare between user engagement and development work
- It's probably better to have different people focused on development and user-facing aspects of the work (e.g. Scrum PO?)
- It's hard to arrange frequent meetings with busy researchers, some of whom are often not in the department or even the country
- Feedback doesn't happen instantly, or quickly

Choice of test users

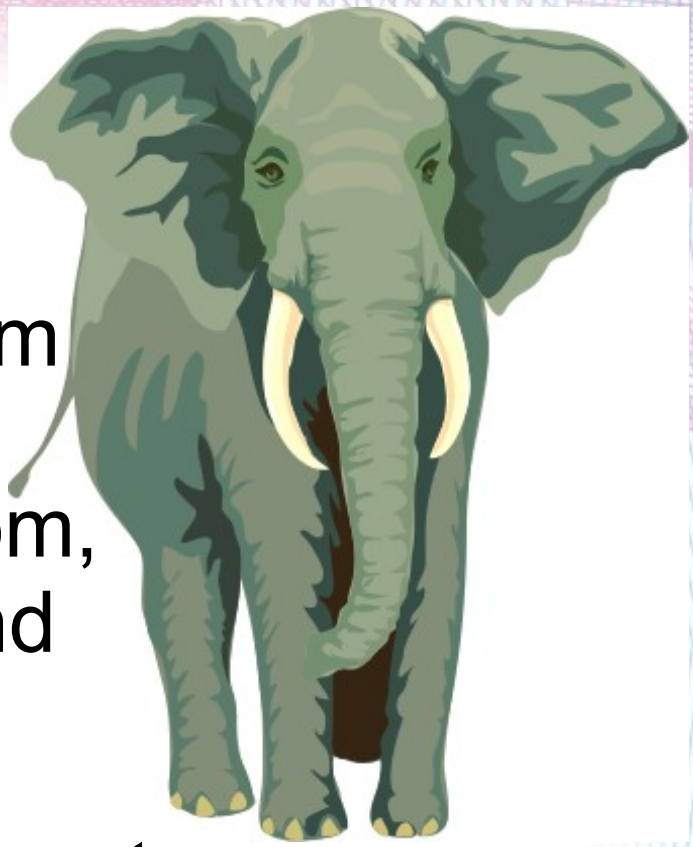
- Our earliest, very engaged test user has been most helpful, and more forgiving of interim deployments and subsequent disruptions
- Choosing engaged, sympathetic users can add enormous value to a project
- Having several test groups signed up has helped us to ensure we have useful level of such engagement. Offering the lions share of resources (e.g. storage) to a fully engaged user might be a good strategic priority

Quality cost

- Because we needed to create a system that researchers could depend upon, much effort was put into creating testing code
- Doing this at the system integration level can be complex and time consuming
- We believe the effort expended did pay for itself over the project life time
- Recall Frederick Brooks' *Mythical Man-Month:*



Storage costs



- The cost of medium-to-long term storage (Tbytes for 5-10 years) remains the elephant in the room, especially for HD video data and multidimensional microscopy
- Researchers seem to accept about £80/Tb/year, which is roughly the hardware cost recovery for SAN RAID array storage
- Full service managed storage appears to cost closer to £1000/Tb/year

Virtualization

- Operational factors and running multiple virtual environments on a single host can be like having all eggs in one basket
- The ability and capacity for live migration of systems to different hardware would provide resilience

Staff retention

- (Broken record time again!)
- Difficulty of attracting/retaining staff, particularly in an academic (non-service) department where staffing is subject to project funding
- Both Databank and ADMIRAL work packages underwent staffing upheavals
- Recognizing this is due to factors beyond JISC control, are there grounds to examine the nature of the overall innovation ecosystem?

Co-development risk

- There is an added risk associated with co-development activities
  - i.e. when independent groups develop inter-dependent components, such that resource conflicts in one group impact the other
  - project rhythms are not so easily synchronized
- This might be mitigated if the substantive new developments are controlled by a common chain of management, especially with respect to priority setting and resource commitment