

SAFE ARCHER champions

Stephen Booth, USL

EPSRC

NERC SCIENCE OF THE ENVIRONMENT



CRAY
THE SUPERCOMPUTER COMPANY

epcc



Introduction

- Aim of this session is to interactive
 - Intend to use the slides as required rather than presenting all of them.
- Who here uses SAFE
 - Questions?
 - Comments?
 - Requests?
 - Discussion?



I have slides on

- [SAFE overview](#)
- [Registration and account requests](#)
- [Reporting and accounting](#)
- [Project management](#)
- [Integration with the HPC systems](#)
- [A Pathfinder project for national AAI](#)



The SAFE

- The SAFE is the web-based portal that provides an interface between us and our users. It also acts as a tool for project managers to manage their project resources.
- Performs the following functions:
 - Registration and user database.
 - Accounting and report generation
 - Self-service operations
 - Devolved project management
 - Helpdesk



History

- Been under continual development since 2002
- Used for various HPC services
 - HPCx
 - HECToR
 - ARCHER
 - DIRAC
 - EPCC local services
 - HARTREE
- Recently finished a major rework of the navigation model



New V4 interface

↑ Your details Service information Projects managed Login accounts IFMS Help and Support

 **SAFE for ARCHER** 
Service Administration from EPCC

Welcome to the ARCHER SAFE

This is the ARCHER SAFE. It is a web-site used to administer the ARCHER HPC service.

All of the functions of the SAFE are available from the menus at the top of the page.

Use the **Your details** menu to view and update the information we hold about you or to change the settings of your SAFE account.

Use the **Service Information** menu to view information about the service or to generate reports from our database.

Use the **Login Accounts** menu to apply for an account on ARCHER. All accounts need to be part of a funded project so this request will need to be approved by a manager of the project you select before the account can be created. If you already have an account you can also use this menu to view the status and make changes to your accounts.

The **Help and Support** menu gives access to the help-desk and contains links to SAFE guides and documentation.

If you are a project manager you will also see a menu for managing your projects. If this menu is coloured orange then this indicates that there are some pending requests for you to process in one of your projects.

[ARCHER SAFE guide](#)

SAFE is an [EPCC](#) product



Documentation

- There is a SAFE guide on the ARCHER web-site
- <http://www.archer.ac.uk/documentation/safe-guide/>
- Covers both user and project manager operations
- *Click on the video links for you-tube videos demonstrations!*



Registration



User details and registration

- We don't know our users in advance.
- Users self-register with the SAFE (providing appropriate details)
 - SAFE verifies the email address as part of registration by emailing completion URL. User sets initial password.
 - Then apply to join a project, requesting an ARCHER account.
 - Project manager approves/declines the request.
 - Account is created and user retrieves initial ARCHER password via safe.
- Users can bind their Institutional login identity (Shibboleth/UKAMF) to their SAFE account.





ARChER SAFE Signup

This is the ARChER SAFE.

Registration form

Fields marked in **bold** are mandatory.

Login systems may have different levels of support for any ssh keys you register here. See the individual system documentation for more details.

All information supplied is held and processed in accordance with the ARChER Personal Data and Privacy Policy. You can find full details [here](#).

Email Address	<input type="text" value="name@example.com"/>
Nationality	<input type="text" value="United Kingdom"/>
Title (Mr,Mrs,Dr etc.)	<input type="text"/>
First Name	<input type="text"/>
Initials	<input type="text"/>
Last Name	<input type="text"/>
Institution for reporting	<input type="text" value="Anglia Ruskin University"/>
Institution	<input type="text"/>
Department	<input type="text"/>
Phone number (include Int. dialing code e.g. +44 for UK)	<input type="text"/> + followed by numbers and spaces
Opt out of user Emails	<input type="checkbox"/>
Address Line 1	<input type="text"/>
Address Line 2	<input type="text"/>
Address Line 3	<input type="text"/>
Address Line 4	<input type="text"/>
Town/City	<input type="text"/>
Postcode	<input type="text"/>
Country	<input type="text" value="United Kingdom"/>
Gender	<input type="text" value="Not Selected"/>
SSH Public key	<input type="text" value="Browse..."/> No file selected.
Career stage	<input type="text" value="Not Selected"/>



Login Accounts

- Users request login accounts on ARCHER via the SAFE
 - ARCHER policy to not allow accounts in multiple projects though SAFE supports this.
 - Standard operations also supported via SAFE.
 - Password resets
 - Package group requests
 - Etc.
 - Users can upload SSH public-keys to SAFE
 - SAFE verifies minimum key-length
 - Some DIRAC hosts don't support passwords and require a public-key. SAFE provides capture and dissemination of these.



Account requests

- Users get to choose their account-name
 - Plug-in policies can restrict possible names or auto-generate Username.
 - DIRAC use a global auto-generated dirac-id.
 - Same id on all systems
 - Reserved space of usernames for shared-use machines.
 - System team can override this when creating account.
- SAFE also suggests UID for account
 - Chosen from a reserved range.
 - SAFE reserves UIDs per person in case FS mounts added later.
- Once created user can view account details and request standard actions.





SAFE for ARCHER

Service Administration from EPCC



Login account details

Username	User16112				
Machines	Name	Model	Login		
	rdf		login.rdf.ac.uk		
Status	Active				
Creation Date	02-Oct-2013				
Projects	z02				
User disk info	Volume			Files	
	rdf (general)			4 Files	
	work (fs4)			1 Files	
z02 resources	Volume	Usage	Quota	Files	File Quota
	rdf (general)	10 GiB	21 GiB	1,799 Files	
	home (home1)	360 GiB	456 GiB		
	work (fs4)	16 GiB	400 GiB	1,428 Files	1,000,000 Files

[New Login Account Password](#)
[View Login Account Password](#)
[Add Machine](#)
[Add Certificate](#)
[Join Project](#)
[New Package Group Request](#)



Reporting and Accounting



Service reporting and accounting

- All usage and allocation data is held in the SAFE.
 - Batch job information
 - Historical disk-usage
- Powerful reporting engine can generate reports based on all information the SAFE holds
 - Project and user meta-data is remarkably useful.
 - % of use by institution, nationality gender career-stage etc.



Cross system reporting

- Reporting engine can combine data from different sources.
- Examples:
 - On ARCHER reservation data and job-data can be merged.
 - Allocation data and usage data can be viewed at the same time.
 - On DIRAC data from different systems can be merged.
- Cross system reporting only makes sense where common units are used CPUs may not be equivalent on different systems.
 - Normally use AU (allocation units)



Properties

- Reporting system sees data-records as collections of named properties
 - Database fields
 - *Derived properties*, Expressions of other properties
 - E.g. $\text{SlowDown} = (\text{EndTime} - \text{StartTime}) / (\text{EndTime} - \text{SubmitTime})$
 - Includes constants
 - Includes de-reference to other tables $\text{UserID}[\text{PersonID}][\text{InstitutionID}]$
 - Usually store properties corresponding to original data and derive abstract properties for reporting.



Reports

- Reports are written in a report generation language
 - Report definitions are XML documents
 - Syntax for
 - Charts/Tables
 - Access-Control
 - Record-Filtering
 - Etc.
- Accounting sub-system is independent of budget sub-system
 - Can just use SAFE reporting engine e.g. with fair-share or externally managed budgets.
 - General enough to handle other use cases like data-transfer logs.



Accounting Data uploads

- Usage data needs to be uploaded to SAFE to record use.
- SAFE does not mandate a particular data-format, aim is to upload the data in native format.
 - Also aim to keep DB format as close to original data as possible
- Plug-in parsers for most common batch systems.
- Policy plug-ins generate charge values and other side-effects
 - Generating aggregated records
 - Decrementing allocations
 - Associating jobs with known software packages
 - Etc.



Policies

- Policies are important for local variations
 - Charging policies differ
 - Charge by node or by CPU-time
 - Priority queues
 - Discount policies.
 - Cross linking data from different logfile.
 - Constructing aggregate records to speed report generation
 - The same tools are used differently by different services use so have to customise the interpretation of data.



Reference properties

- Properties are strongly typed:
 - Number, String, DateTime, Reference
- Reference properties are pointers to records in other tables.
- Property expressions can access properties on the remote record.
 - Charge = Residency * Machine[ChargeRate]
 - When mapped to SQL this implies a JOIN to the remote table.



09/09/16

Service DATA from the SAFE



21



SQL mapping

- Property expressions need to be mapped to SQL fragments.
- Internally expressions stored as abstract-syntax-tree objects.
 - Visitor pattern used to implement operations.
 - Evaluate expressions
 - Create SQLValue
 - Create SQLExpression
- SQL fragments represented as SQLValue objects. Implement methods to:
 - Add fragment to SQL statement
 - Create value from java ResultSet
 - Optionally provide SQLFilter to modify FROM and WHERE SQL clauses.

SQLExpressions

- Sub-type of SQLValue
 - SQL fragment is a single expression equivalent to value produced.
 - Can be combined at SQL level.
- Required for SQL reduction operations
 - MIN, MAX, SUM etc.
- In other contexts SQLValue is sufficient

Non SQL operations

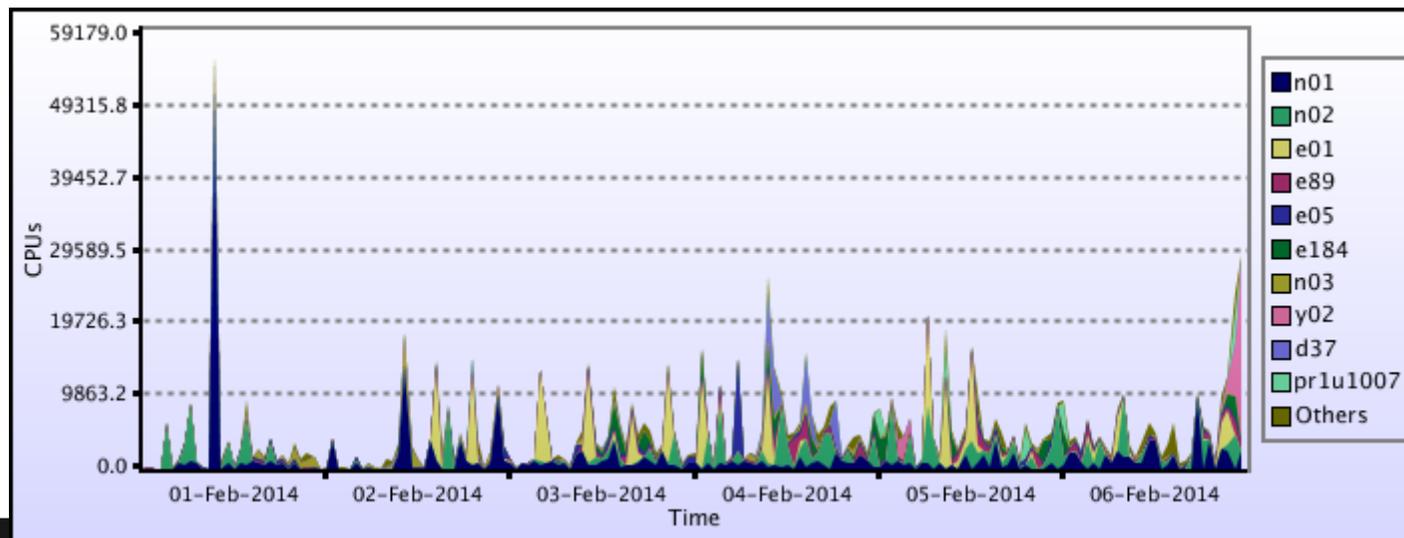
- Also possible to implement properties using Java fragments with no SQL equivalent.
- Reporting code falls-back to iterating over objects.
- Only needed in special circumstances.
 - Elapsed working hours for helpdesk reporting.

Time mapping

- SAFE provides special support for records that overlap ends of the time periods.
 - For long time periods it is sufficient to select records based on a single time property (e.g. completion time of job).
 - For short time periods (comparable or shorter than the job length) this gives a distorted view of the data.
 - This is a particular problem for graphs showing evolution against time as each point on the graph corresponds to a short time period.

Naïve Time graphs

- If we plotting records that finished in a time period.
 - Graph depends strongly on choice of plot periods.
 - Graph is also difficult to interpret
 - Example shows average size of job completing in each period.



Weighting functions

- Records that overlap ends of time period need to be scaled.
- Two distinct types of scaling are needed depending on nature of property being.
 - Accumulating properties
 - Values that accumulate during record
 - CPU-Time,
 - Wall-clock
 - Charge
 - Instantaneous properties
 - Values that can be measured at a particular instant in time
 - CPUs
 - Memory used

Accumulating properties

- Need to be weighted by fraction of record that overlaps with the period

$$V \downarrow period = V \downarrow record \downarrow overlap / \downarrow record$$

- Values may be summed to give value accumulated in period.
- Divide by period length to give an average rate.

Instantaneous properties

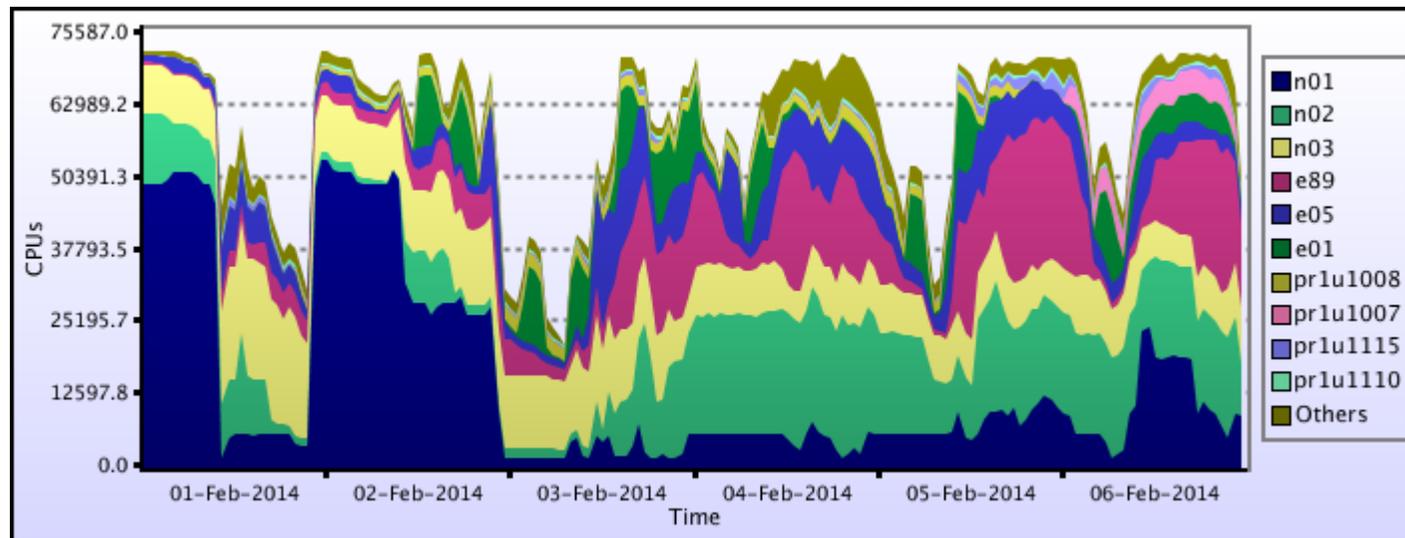
- Value at any given instant is the sum of values from records that cross that point in time.
- Representative value for the period is the time average
- Weight by fraction of *period* that is overlapped and sum over records.

$$V_{\downarrow period} = V_{\downarrow record} \ell_{\downarrow overlap} / \ell_{\downarrow period}$$

- Can turn instantaneous property into accumulating by multiplying by record length
 - Time average is the same as the rate calculated from this property.

Time graphs

- Time averaged CPU plot
 - At this resolution many jobs cross multiple time periods.
 - Different division into plot periods would give broadly similar plot.



09/09/16

Service DATA from the SAFE



30



Time bounds

- Need to choose two time valued properties to denote start and end of record.
- This may be different depending on quantity.
- CPUs used would use *job-start* and *job-end*
- Jobs waiting would use *job-submit* and *job-start*



09/09/16

Service DATA from the SAFE



31



Advanced use cases

- Recently we have been exploring importing mpi-job-launcher data (ALPS logs on Cray systems)
- Essentially syslog event data
 - Process start-end events come from 2 different places and result in 2 different log-lines
- SAFE uses an “incremental” parser to merge these into a single database record.
 - First line results in partial record
 - Data from second line merged to produce final data.
- Policy links these back to the PBS database record for the job.
 - Some users run many mpi jobs within a single batch job.



Executable data

- Alps logs contain the **aprun** command line
- 2nd Policy performs nested parse of this to generate executable names.
- 3rd Policy links this to known applications using regular-expressions.
- End result is data on;
 - Node utilisation within batch jobs
 - Application usage data
 - Thread binding information



Memory and power

- The ALPS system sits between PBS and the execution nodes
 - On Cray systems PBS does not know memory and energy usage of back-end nodes.
 - However alps does have its own monitoring plug-ins
- Another log-file to merge and link in the database.



Project Management



Project management

- We allocate resources to projects. Membership and resource management within a project is devolved to project managers via the SAFE.
 - Approve/reject membership requests
 - Create new groups/budgets
 - View and Set group and user disk quotas.
 - Allocate CPU-time allocations to budgets.
 - Time is allocated to projects in time-windows
 - Managers can move this allocation between budgets
 - Time un-used at the end of the window is lost.





SAFE for ARCHER

Service Administration from EPCC



ARCHER Project Notifications for - y07: Packages

This page shows project notifications for y07. Operations that require approval by a project manager appear on this page.

New project management requests

Project: Packages (y07)

Full Name	User Name	Machine	Type	Request date	
Dr Nobody Jones	User22836	ARCHER	New Account Request	Fri Mar 21 08:39:11 GMT 2014	Process
Mr Nobody A Jones	User23092	ARCHER	New Account Request	Mon Mar 31 15:37:01 BST 2014	Process
Miss Nobody Jones	User26459	ARCHER	New Account Request	Fri Jan 16 11:31:58 GMT 2015	Process
Dr Nobody R Jones	User27664	ARCHER	New Account Request	Thu Apr 02 13:31:39 BST 2015	Process

These are outstanding requests by people who have applied to join your project - you should click on **Process** to accept or reject them.





SAFE for ARCHER

Service Administration from EPCC



Update ProjectManager

Principal Investigator	Dr Stephen P Booth
Name	USL
Description	<input type="text" value="test"/>
Project Code	z01
New Account Signup Notification List (Comma separated list of email addresses)	<input type="text" value="uadmin@epcc.ed.ac.uk"/>
Research Grants	<input type="text"/>
Subject area	support: Support project
Frequency of automatic reports	MONTHLY
Recipients for automatic reports	<input type="text" value="s.booth@ed.ac.uk,uadmin@mlist.is"/>
Frequency of alerts	DAILY
Recipients for alerts	<input type="text" value="s.booth@ed.ac.uk"/>



Groups

- Internal project management is expressed in terms of groups.
 - These can be Unix groups (allocating disk space)
 - These can be Budgets (allocating cpu/node time)
 - Or they can be both.
- Project managers control the creation and membership of groups through the SAFE.
 - For per-user allocations make a group for each user.
- Some management operations can be delegated to group-managers.





SAFE for ARCHER
Service Administration from EPCC



Project: z01 (USL)

Details:

Code	z01
Start Date	18-Dec-2006
End Date	31-Mar-2017
Research Grants	
Principal Investigator	Dr Stephen P Booth
Signup Notify List	uadmin@epcc.ed.ac.uk
Description	test
Machines	<ul style="list-style-type: none"> • services • ldsx4 • hector • phase2b • ims • phase3 • ARCHER • site • rdf

Management

Dr Nobody Jones
Dr Nobody D Jones
Dr Nobody Jones
Ms Nobody M Jones
Mrs Nobody Jones

Dashboard status

This is a status dashboard that generates alerts and reminders.

Severity	Name	Target	Notes
WARNING	allocation.project	z01 - USL	No project allocations

Project mailings

Project Groups

Project z01 (USL) has the following groups and resources:

	Volume	Usage	Quota	Files	File Quota
z01	rdf (general)	7,589 GiB	10,000 GiB	5,121,440 Files	
	home (home1)	1,373 GiB	5,000 GiB		
	work (fs3)	7,217 GiB	19,986 GiB	3,128,790 Files	4,000,000 Files
z01-cse					
z01-csreserve					
z01-csetds					
z01-hook					
z01-hook1					
z01-prace					
z01-reserve					
z01-test	Volume			Files	
	work (fs3)			5 Files	
z01-test2					
z01-test3					

[ARCHER SAFE guide](#)

SAFE is an EPCC product





SAFE for ARCHER

Service Administration from EPCC



[<< Back to Project Admin Page](#)

Time Budgets

Projects have a limit to the amount of service-machine runtime they can use. This 'time budget' is measured in Kilo-Allocation-Units (KAU).

Time budgets are grouped into resource pools. This allows time on different machines to be managed separately. Though in practice many projects will only have access to time from a single resource pool.

When a project is created, all of the AUs allocated to it are initially stored in it's 'reserve budget'. By managing your budgets, you can move AUs from your reserve budget into your project's main budget, or into the budget of any of your subgroups. Only members of a subgroup may run jobs that use AUs of that subgroup.

To manage your budgets, click on the following button.

Disk Quotas

Disk quotas can be managed on a per user basis, or by group.

Click on one of the following buttons to manage a user or group quota.

User Quotas

Description	Manage
Archer work file system	<input type="button" value="work (fs3)"/>
Other funding bodies filesystem on the RDF	<input type="button" value="rdf (general)"/>
Archer home file system	<input type="button" value="home (home1)"/>

Group Quotas

Description	Manage
Archer work file system	<input type="button" value="work (fs3)"/>
Other funding bodies filesystem on the RDF	<input type="button" value="rdf (general)"/>
Archer home file system	<input type="button" value="home (home1)"/>

User quota report

[<< Back to Project Admin Page](#)





SAFE for ARCHER

Service Administration from EPCC

[<< Back to Project Resources Page](#)

Time Budget Management

Project	z01
Resource Pool	XC
Total allocation	40,000.0 kAUs
Allocation Period	Feb - Nov 2016

Budget Allocation

Projects have a limit to the amount of service-machine runtime they can use. This 'time budget' is measured in Kilo-Allocation-Units (KAU).

How to Manage Your Budgets

- Select both a 'to' and 'from' radio-button.
- Enter the amount of time to be moved from the 'from' budget to the 'to' budget.
- Click on the move button. This will only change the 'After Change' column and will not actually change your budgets.
- Repeat the above steps until the 'After Change' column shows your desired budget allocations.
- Click on the 'Submit Budget Allocation Changes' button to submit your changes. **No changes will be made to your budgets unless the 'Submit Budget Allocation Changes' button is pressed!**
- If at any time you wish to reset all of the 'After Change' column to the values in the 'Current' column, press the 'Reset' button.

Group Time Allocations

Group	Allocation	Remaining	After Change	Move From	Move To
z01	0.0	0.0	0.0	<input type="radio"/>	<input checked="" type="radio"/>
z01-cse	10,000.0	5,283.8	5,283.8	<input checked="" type="radio"/>	<input type="radio"/>
z01-csereserve	0.0	0.0	0.0	<input type="radio"/>	<input type="radio"/>
z01-csetds	500.0	482.3	482.3	<input type="radio"/>	<input type="radio"/>
z01-hook	0.0	0.0	0.0	<input type="radio"/>	<input type="radio"/>
z01-hook1	0.0	0.0	0.0	<input type="radio"/>	<input type="radio"/>
z01-prace	0.0	0.0	0.0	<input type="radio"/>	<input type="radio"/>
z01-reserve	29,500.0	29,500.0	29,500.0	<input type="radio"/>	<input type="radio"/>
z01-test	0.0	0.0	0.0	<input type="radio"/>	<input type="radio"/>
z01-test2	0.0	0.0	0.0	<input type="radio"/>	<input type="radio"/>
z01-test3	0.0	0.0	0.0	<input type="radio"/>	<input type="radio"/>
Total	40,000.0	35,266.1	0.0		

How much to move: kAU

[<< Back to Project Resources Page](#)

Disk space management

- Safe can manage disk allocations
- Project managers can re-allocate disk quota between groups (up to overall project limit)
- Can also manage user quotas though these are unconstrained
- Changes requested via ticket interface
- Current usage and quota values uploaded at regular intervals
 - Values available to users
 - SAFE keeps history tables so disk usage history can be shown in reports.



Guest Budgets

- On ARCHER we don't allow users to belong to more than one project.
 - This is to keep disk resources clearly associated with a single project.
- Sometimes a user needs access to time resources from multiple projects.
- Users from outside a project can be given access to guest budgets.
 - Only a Time-budget never a Unix-group.
- User applies for access. Manager of the owning project approves/rejects access.



Package Groups

- Package Groups are Unix groups controlling access to software packages.
 - Only a Unix-group never a Time-budget.
- Requested in a similar way to guest-budgets
 - Approvals handled by the helpdesk.



Systems interface



Helpdesk

- ARCHER helpdesk is also part of the SAFE.
 - All users automatically have access to helpdesk web-interface
 - Single unified view of contact details, projects etc.
 - Automated self-service operations can be integrated with the helpdesk.
- Most traffic via Email so optimised for this use case.
- Planning a major overhaul of this code soon.



Ticket interface

- The SAFE requests machine changes via “ChangeTickets”
 - These are independent of the helpdesk code but if helpdesk is enabled automatic tracking helpdesk tickets are created.
 - Implement a set of standard operations
 - CreateAccount, NewPassword, AddToGroup, CreateDirectory etc.
 - Can be viewed via a web-interface
 - Also downloadable in XML or JSON for automated scripting.
 - Tickets handled by an action plug-in.
 - Default action to email system administrator.
 - Action performed manually then completed via web-form
 - Can fully-automate implementation via scripting action (script consumes XML/JSON description of ticket)
 - We usually keep humans in the loop but script the implementation avoiding data transcription problems.



Completing tickets

- Ticket completion allows information to be passed back to the SAFE
 - Can refuse tickets or override SAFE selected values for Username UID GID etc.
- Tickets can be routed to different support groups depending on service machine.



Resource limits

- SAFE can manage CPU-time allocations
- Every time allocation state changes SAFE writes a new version of the “budget-files”
 - Accounting data is uploaded
 - Time moved between budgets
 - Account created or added to budget
- Simple format files giving current state of budget and budget ACL.
 - Need to add hook to batch system to enforce these rules.
 - Can also pull current budget values via http



Enforcement

- Batch system has to enforce SAFE access rules when using devolved budget management.
 - Usually this means the user has to provide a budget-code for each job
 - Or force users to only have access to one budget.
- Normally process accounting logs once a day.
 - Time window for users to exceed allocation.
 - SAFE records this as an overdraft. Project needs to pay-off excess to unlock budget.
- Reporting engine can still be used without enforcement.



Machines at multiple sites

- For DIRAC we used a single SAFE instance to manage multiple geographically distributed sites.
 - Many of these were shared machines with other non-dirac users as well.
- Initially kept things simple:
 - One account per user per site.
 - One active project at any time.
 - No disk space reporting/management.
 - No project sub-groups.
- As service evolved more advanced features were adopted by some sites.



Implementation

- SAFE is a J2EE servlet application written in java (version 1.7+)
 - Usually run under tomcat (version 7+)
 - Usually use apache httpd as a front-end mod_jk or mod_proxy_ajp
 - Mysql database back-end
 - Needs SMTP mail-server to send email
- Can be configured to integrate with external authentication mechanisms (pretty much anything supported by apache)
- SAFE designed to work without requiring browser side scripting. E.g. using html5 input types



Installing SAFE

- Basic install is straightforward
 - SAFE will bootstrap into empty database.
- However no two HPC services are run exactly the same
 - Even services run by a single organisation can be different as circumstances and technologies change
 - SAFE contains a large number of options and switches for the variations we have had over 14+ years
 - Still plenty of (perfectly reasonable) options we have not explored.
 - Always expect some tailoring to be required.



A pathfinder project for a National AAI

- Recently Grant started 1 Sept
- Looking at developing an Infrastructure for:
 - Authentication
 - Authorization
 - Accounting
- SAFE is a part of this project.



The user problem

- It is quite common for users to use many different online facilities.
 - Most institutions provide good AAA integration for services within that institution
 - Single-sign-on common identities service integration etc.
 - This often extends to 3rd party services bought directly by the institution
 - Big shared facilities have to set up their own AAA solutions
 - Very large distributed research collaborations may introduce their own Infrastructures
- Many “normal” users find themselves having to deal with multiple systems.



The project problem

- Projects can have a similar problem when it comes to management interfaces
 - Unless they are small enough to use a single facility or large enough to mandate the complete software stack.



Authentication (AuthN)

- The user proves their identity to a service.
 - May be a partial identifier (information about the user not which user)
- To support Single-Sign-On (SSO) this is best provided by a trusted third party the IDentity Provider (IDP).
 - Trusted by the user and any service they want to use.
 - Each user wants to use a single IDP though all users do not have to use the same one.
- Probably the best choice the users home institution
 - Best position to confirm identity, manage credential resets, most likely to know when a user changes job.
 - Requires a federation of institutional IDPs



Authorization (AuthZ)

- The project confirms to the service that the requested resource access should be allowed.
 - Though this normally takes place as a user requesting access to a resource; the resources were typically bought or granted to an organisation (the project) and the interaction is ultimately between the project and the service.
- In most classic AAI models authorization rules are expressed in terms of *user-attributes* provided by the IDP during authentication.
 - This is a problem if the project and the IDP are not the same.



Accounting

- The service provides resource usage information to the project.
- Accounting information may be used for:
 - Auditing – ensuring the resource was used appropriately.
 - Billing – generating a charge for variable-cost services.



Moonshot

- JISC developed technology
- Adapts the RADIUS protocol used by eduroam to other use-cases (web-access and ssh)
 - Universities already run radius servers 👍
 - Needs extra software client-side 👎
 - Not many moonshot IDPs up and running yet 👎



SAFE + Moonshot

- We will be looking at how we can integrate SAFE and Moonshot
 - Allow users to log-in to the SAFE using moonshot
 - Use the SAFE as a moonshot IDP for users without a local IDP
 - Having the SAFE add additional user-attributes while still authenticating against the users preferred IDP.
- Not looking to mandate a solution
 - Looking for an architecture where multiple independent implementations can co-exist.

