

The background features a dark gray upper half and an orange lower half. Scattered across the dark gray area are several white line-art icons of server racks, each with a circular indicator and horizontal lines representing slots. A vertical orange bar is positioned on the left side of the image.

HALCYON

UNIFIED HPC CENTER OPERATIONS



Defining The Problem

Cluster Management For Faculty

Under a shared campus cluster model, with many research groups investing distinct amounts and annual new hardware acquisitions, management of users and resources can become very complex.

At Purdue, the Research Computing division set out in 2011 to design a cluster management solution to empower faculty to manage access to their own purchased resources.



Immediate Results

Time & Effort Savings

Staff time

6.7 hrs / wk

New storage space

8 days → 1 hour



In its initial incarnation, portal provided faculty, students, and center staff an immediate reduction in time and effort to request, provision, and manage allocations. User allocation alone was estimated as saving 6.7 hours of center staff time per week over the first 18 months. New storage spaces were able to reduce time-to-completion from 8 days to 1 hour within the first year of adoption of storage management.

Portal Evolution

More Features, More Problems

In 2011 a cluster management solution to empower faculty to manage access to their own purchased resources is designed and built.

Time-savings and preference for “self-serve” utilities were immediately apparent.

As operations at Purdue expanded, the internal portal evolved and took on many aspects of the operation of an HPC center beyond resource allocation and management. This included:

- HPC & storage resource management
- User management
- Customer relationship
- Communications
- Documentation
- Purchasing

While the original Purdue Research Computing portal served its purpose well over the years, making changes and adding features could be a cumbersome task. Dependencies were becoming increasingly complex and tangled.

The decision was made to re-architect the underlying code.

Re-Architecture

Planning For The Future

01 Abstraction of assumptions & settings

Standard local language for many aspects and various hard-coded site settings were pulled into the internal database, and interfaces were created for editing these. Other environment site settings were incorporated through new configuration files.

02 Restructure for modularity & extendability

The portal code was separated into modules—logical groupings based on the data being handled, tasks, and interfaces. 3rd-party services are integrated as plug-ins through the an event dispatcher system, allowing communication with sources such as LDAP or REST APIs.

Re-Architecture

External Interactions

REST API



Interaction points for external services and resources, modules implement REST APIs, documented according to the OpenAPI v3 spec.

Message Queue



External resources such as compute clusters or storage can make use of a message queue to make changes such as new directories or alter permission.

Command Line Utility



A robust command line utility is provided with numerous commands for inspecting and interacting with data.

API Documentation 0.0.1

Resources

GET

/api/resources

Display a listing of the resource.

Parameters

Name	In	Type	Default	Description	Accepted values
limit	QUERY	INTEGER	25	Number of result to return.	
page	QUERY	STRING	null	Number of where to start returning results.	
search	QUERY	STRING	null	A word or phrase to search for.	
order	QUERY	STRING	null	Field to order results by.	
order_dir	QUERY	STRING	asc	Direction to order results by.	asc, desc

POST

/api/resources

Create a resource

GET

/api/resources/{id}

Read a resource

PUT

/api/resources/{id}

Update a resource

DELETE

/api/resources/{id}

Delete a resource

GET

/api/resources/{id}

Read a resource

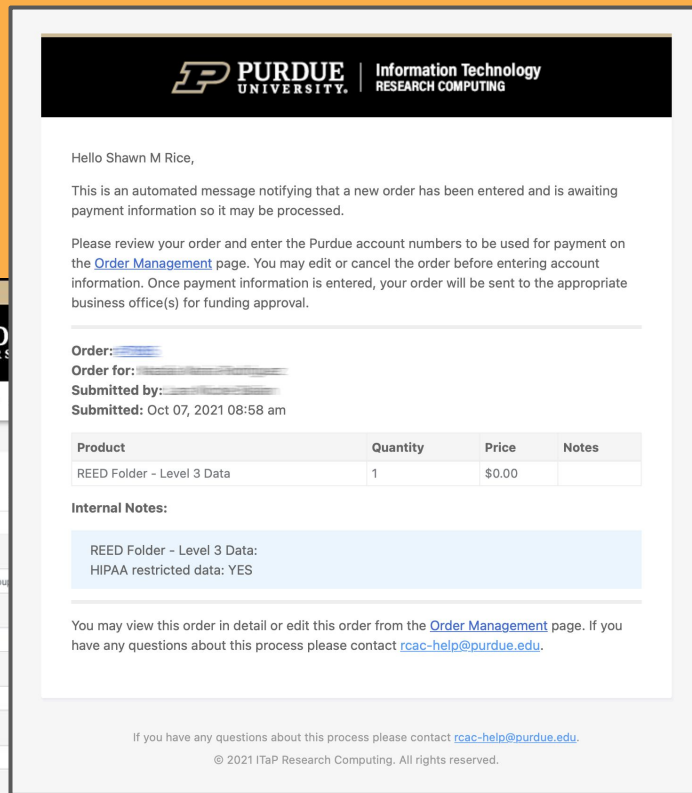
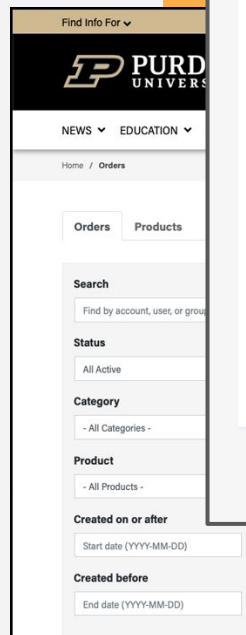
Sub-resources

GET	/api/resources/subresources/	Display a listing of the resource.	
POST	/api/resources/subresources/	Create a resource	
GET	/api/resources/subresources/{id}	Read a resource	

Self-Serve Interfaces

Make Purchasing Easy

Re-architecture did not stop with the codebase and included careful scrutiny and re-thinking of user interfaces. Allocations can be ordered through a familiar online shopping experience and users are notified of status changes via automated emails. Payment accounts can be added to an order with payment split across various accounts and the relevant business offices assigned as approvers. Much of this is self-serve, minimizing direct staff involvement.



Order ID	Order Date	Order Status	Order Amount
7203	2021-07-14	Pending fulfillment	\$ 0.00
7200	2021-07-13	Pending business office approval	\$ 300.00
7201	2021-07-13	Pending business office approval	\$ 511.61
7199	2021-07-13	Pending collection	\$ 210.00

Self-Serve Interfaces

Giving Control To Users

Resource allocations are centralized under research groups and options for control over membership, unix groups, storage directories and access are continually expanded, allowing for an ever increasing self-serve experience. This frees up time for staff and hastens a number of processes as the user no longer must rely on a middle-man.

The screenshot shows the Purdue University Information Technology Research Computing self-serve interface. The top navigation bar includes links for Find Info For, Apply, News, President, Shop, Visit, Give, Emergency, and a search icon. The Purdue University logo and "Information Technology RESEARCH COMPUTING" are prominently displayed. Below this is a secondary navigation bar with links for NEWS, EDUCATION, ACCOUNT, COMPUTE, STORAGE, ENVISION, PURCHASE, SERVICES, ABOUT, and EXSEDE. The main content area is titled "Stat Courses Group" and features a "group manager" badge. On the left, a sidebar for "MARK DANIEL WARD" lists various account and resource management options. The main panel shows tabs for Overview, Members, History, Orders, Queues, Storage, and Notices. A "Managers" section includes a "Filter users..." input and a table of users and their group memberships. The table has columns for Name, Username, Options, and four Unix Groups: statclass, statclass-apps, statclass-artist, and statclass-cistar. Two users are listed: Doug Crabill and Mark Daniel Ward, both with checkmarks in the statclass and statclass-cistar columns. A "Members" section is partially visible at the bottom.

Find Info For ▼ Apply News President Shop Visit Give Emergency Q

PURDUE UNIVERSITY Information Technology RESEARCH COMPUTING Log Out Get Help

NEWS ▼ EDUCATION ▼ ACCOUNT ▼ COMPUTE ▼ STORAGE ▼ ENVISION PURCHASE ▼ SERVICES ▼ ABOUT ▼ EXSEDE ▼

Home / Mark Daniel Ward / Groups / Stat Courses Group

MARK DANIEL WARD Stat Courses Group group manager

Accounts
Software Access Requests
Groups 6
Quotas
Orders 0
Accounts for Classes 1

Overview Members History Orders Queues Storage Notices

Export to CSV Add Member

Managers ?

Filter users...

User Info Unix Groups


Name	Username	Options	statclass	statclass-apps	statclass-artist	statclass-cistar	st
Doug Crabill	dgc	---	✓	✓	✓	✓	
Mark Daniel Ward	mdw		✓	✓	✓	✓	1

Members ?

Filter users...

Find Info For

ApplyNewsPresidentShopVisitGiveEmergency

PURDUE
UNIVERSITY

Information Technology
RESEARCH COMPUTING

account train02Log OutGet Help

NEWSEDUCATIONACCOUNTCOMPUTESTORAGEANVILENVISIONPURCHASESERVICESABOUT

Home / account train02

ACCOUNT TRAIN02

Accounts

Request Access

Accounts

Software Access Requests

Groups3

Quotas

Orders0

Accounts for Classes0

▲

Note:

The Data Depot storage service is transitioning to new hardware. You are a member of one or more groups with Data Depot space. Check below for the migration status of the Depot space. For further details, check the [FAQ](#).

Path	Group	Status
/depot/nihomics	NIH Big Data Omics Workshop	Migrated
/depot/bigcare	BigCare	Not migrated

Profile

Username

train02

Title

(unknown)

Phone

(unknown)

Department

(unknown)

Campus

(unknown)

Building

(unknown)

Self-Serve Interfaces

Centralized Information

Information relevant to a user can be found from their account page. This reduces time and effort on both the customer and staff in looking up details on interactions, allocations, and more.

Each menu and its related page is injected via an event dispatcher. Modules or plugins can subscribe to events and add information or entire pages as needed, allowing for integration of new features but avoiding hard-coding of dependencies.

All The Things

Unified HPC Center Operations

Resources

Manage resources and allocations.

Purchasing

Products, carts, and orders. Allow users to purchase resources with a familiar online shopping experience.

Documentation

How to use the resources provided by the HPC center.

News

News stories, events, maintenance and outage notifications.

Contact Reports

Customer relationship. Document when and what was talked about.

Online Presence

Informational pages, menus, contact forms, and other utilities needed for a HPC center's online presence.

Extensibility

More Features, No Problems

Extending functionality is considerably easier and entire modules can even be replaced with ease if the current functionality doesn't meet current needs.

The image displays three overlapping screenshots of a web application interface, illustrating its extensibility.

Top Left Screenshot: Shows a website header for Purdue University with navigation links: NEWS, EDUCATION, ACCOUNT, COMPUTE, STORAGE, ANVIL, ENVISION. The main content area features a sidebar with 'Manage News', 'Search News', 'RSS Feeds', 'Announcements', 'Events', 'Outages and Maintenance', and 'Science Highlights'. The main content area displays 'Outages and Maintenance' with a section titled 'Weber Cluster Maintenance' and another titled 'Unscheduled Brown and Hammer outages'.

Top Right Screenshot: Shows a 'Page Manager' dashboard. It includes a search bar, a '+ New' button, and an 'Options' button. Below these are filters for 'Published' and 'Access level'. The main table lists pages with columns: ID, TITLE, PATH, STATE, ACCESS, and LAST MODIFIED. The table contains 20 rows of data, including pages like 'Home', 'About Us', 'How to Acknowledge Use', 'Collaboration', 'Contact Us', 'Research Computing Impact', 'Presentations', 'PEARC 2018', 'PEARC 2019', 'PEARC 2020', 'PEARC 2021', 'XSEDE 2015', 'XSEDE 2016', 'Publications & Videos', 'Coates Installation Day', 'Steele Installation Day', 'Preview of Steele Installation Day', 'Faculty Talk about the Steele Community Cluster', and 'Saul Rosen'.

Bottom Screenshot: Shows a 'Manage News' section. It includes a sidebar with 'Manage News', 'Search News', 'RSS Feeds', 'Announcements', 'Events', 'Outages and Maintenance', and 'Science Highlights'. The main content area displays 'Outages and Maintenance' with a section titled 'Weber Cluster Maintenance' and another titled 'Unscheduled Brown and Hammer outages'.

Collaboration



Implementing Halcyon at another organization is a current goal, which will further highlight where flexibility and configuration can be introduced to better accommodate different infrastructure and HPC center policies.

Refining



While there has been significant re-architecture, there still remain a few values and interactions built into code which are specific to the environments and implementations of Research Computing at Purdue. Work on improving UI/UX continues.

Future Work

Things To Come

With Thanks

References



Shawn Rice
Research Computing
zooley@purdue.edu



Kevin Colby
Research Computing
colbykd@purdue.edu

Icons provided by *Komkrit Noenpoempisut*
<https://thenounproject.com/itim2101>

- 1** Kevin D. Colby, Daniel T. Dietz, Preston M. Smith, and Donna D. Cumberland, "Self-service Queue and User Management in Shared Clusters," in *Proceedings of the 1st International Workshop on HPC User Support Tools, HUST14, Nov. 2014, New Orleans, LA, USA*, IEEE Press, 2014.
- 2** Mohammed Zia, et al., "ColdFront: An Open Source HPC Resource Allocation System," Presented at the 5th International Workshop on HPC User Support Tools, HUST18, Nov. 2018, Dallas, TX, USA. 2018.
- 3** Laravel LLC, "The PHP Framework for Web Artisans", laravel.com. [Online]. Available: <https://laravel.com/>. [Accessed: Jul. 16, 2021].
- 4** iMatrix Corporation, "Distributed Messaging", *zeromq.org*. [Online]. Available: <https://zeromq.org/get-started/>. [Accessed: Jul. 16, 2021].
- 5** Donna D. Cumberland, "Six Sigma: Research Account Queue Management Process Improvement", West Lafayette: Purdue University Press, 2011.
- 6** Kevin D. Colby, Daniel T. Dietz, and Lev A. Gorenstein, "Shared Research Group Storage Solution with Integrated Access Management," in *Proceedings of PEARC17, 2017*, New York: ACM, 2017.