

Welcome!

Gen3 Community Forum, Oct 2022

Gen3 Community Forum – Day 3 Agenda



1. [Introduction](#) to Day 3
2. [Introduction](#) to the Four Breakout Sessions
3. [Reports](#) from Breakout Sessions
4. [Discussion](#) of Future Community Development

Slack

1. Join the designated Slack channel to post questions:
 - <https://gen3friends.slack.com/archives/C044Y5LUNQK>
2. If you are not able to join the Slack channel, please still post the question in Zoom chat, and a moderator will pick up the question and inform the chair.

Zoom

1. Please keep your microphone muted at all times unless speaking.
2. Questions will be asked by a moderator monitoring the chatrooms to properly conduct the flow of the meeting.
3. We encourage you to turn on video when speaking and elaborating on discussion topics.

Gen3 Community Forum - Day 3 Structure



1. Introduction to Breakout Sessions		9:00 p.m. - 9:15 p.m. GMT
2. Breakout Sessions		9:15 p.m. - 10:15 p.m. GMT
○ #1: Gen3 and third party standards		—
○ #2: Standing up a Gen3 Instance		—
○ #3: Communications and processes around bug and issue tracking		—
○ #4: Community prioritization of new Gen3 features		—
3. Reports from Breakout Sessions		—
4. Future Community Development Structure		10:15 p.m. - 10:20 p.m. GMT
5. Open Discussion		10:20 p.m. - 11:15 p.m. GMT
6. Day 3 Recap and End Summit		11:15 p.m. - 11:30 p.m. GMT
		11:30 p.m. - 11:55 p.m. GMT
		11:55 p.m. - 12:00 a.m. GMT

Breakout Session Introduction

Breakout Sessions

1. Gen3 and standards - GA4GH, Data GUIDs, and more
2. Standing up a Gen3 instance
3. Improving communications and processes around bug and issue tracking
4. Prioritization of new Gen3 features and joint roadmap planning (6-18 month roadmap)
5. Open topic - suggested by the community
6. Miscellaneous topics

Breakout 1: Gen3 and Standards



- Overview of GA4GH and Gen3's involvement
- DRS
- DRS prefixes and compact identifiers
- RAS + Passports
- Identifying relevant RDA standards and recommendations

Breakout 2: Standing up Gen3 instances



- Understand the basics of spinning up a Gen3 environment
- Get an overview of “Cloud Automation”
 - Where to start looking?
- Get an overview of deployment to AWS using cloud-automation
- Deploy gen3 on your laptop
- Upcoming improvements

Breakout 3: Communications & Processes, Bug & Issue Tracking



Discuss the current processes and brainstorm ideas to improve them.

- How to communicate that there is a bug?
- How to communicate that you have a fix?
- How to stay aware of the status of bugs?

Breakout 4: Gen3 Feature Development & Roadmap



How can community members collaborate with CTDS and each other to develop new features?

- Process to request or publicize needs and development plans
- Feature registry
- Feature documents
- Community prioritization
- Open discussion of ideas and opportunities to improve the process

Breakout Sessions are designed to engage and listen to community feedback.

To join a Breakout Room in Zoom,

1. Click Breakout Rooms in your meeting controls.
 - a. This will display the list of open breakout rooms created by the host.
 - b. Note: Click Expand All to expand all available rooms and see which participants are in that particular room.
2. Hover your pointer over the number to the right of breakout room you wish to join, click Join, then confirm by clicking Join.
3. Repeat as necessary to join other breakout rooms, or click Leave Room to return to the main session.

If you need assistance, please mention in chat the session you would like to join, and an administrator will place you into the room.

Gen3 Standards and Integration

Michael Lukowski

- GA4GH
 - DRS
- Passports
 - RAS

- Global Alliance for Genomics and Health
- Made up of work streams i.e. large scale genomics, cloud, discovery
- NCI GDC is a driver project
- UChicago / Gen3 is a large contributor in the cloud work stream
- We are also contributors to the cross work stream FASP (Federated Analysis Systems Project)

- Driver project on DRS
- I'm the API co-lead for DRS on the cloud work stream
- We maintain 17 Gen3 DRS servers (stats.gen3.org)
- More than 15 PB of data accessible through Gen3 DRS servers.

- In our DRS servers we use prefixes to make our identifiers globally unique.
- This is useful to create a permanent link between a file and a DRS ID
- An example would be `dg.4503/c97507dd-bb79-45ce-9186-3d14ad823f81`
- "dg.4503" represents the prefix that is associated with Bio Data Catalyst.
- We are able to resolve these through identifiers.org, n2t.net or dataguids.org

- Passports is a GA4GH standard for authorization between platforms and services.
- RAS is an NIH specific instance of a passport
- A RAS passport is GA4GH compliant, but a GA4GH Passport is not RAS compliant

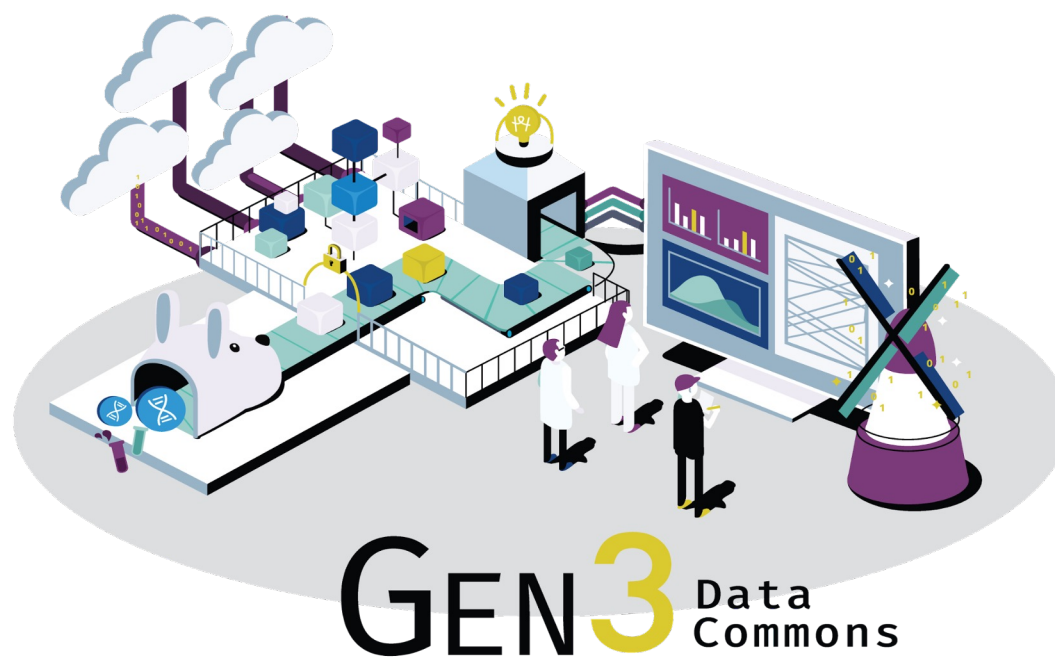
We accept RAS passports in the following environments

- NCI DCF (Data Commons Framework)
- NHGRI AnVIL (Analysis, Visualization, and Informatics Lab-Space)
- NHLBI Bio Data Catalyst

Thoughts on relevant RDA standards and recommendations.

Questions?

GEN3



Breakout 2: Standing up Gen3 instances

Standing up a Gen3 instance (getting started, documentation, how-tos, help, etc.)

- Understand the basics of spinning up a Gen3 environment
- Get an overview of “Cloud Automation”
 - Where to start looking?
- Get an overview of deployment to AWS
- Deploy gen3 on your computer
- Upcoming features

“Cloud-Automation”

<https://github.com/uc-cdis/cloud-automation>

- Custom Platform developed by CTDS to deploy Gen3
 - Manage cloud infrastructure via Terraform
 - Manage deployments in Kubernetes clusters
 - Operations against AWS and google cloud api's
 -amongst other things

cloud-automation/

tf_files/ # terraform infrastructure

kube/ # Kubernetes resources

Docker/ # Docker images

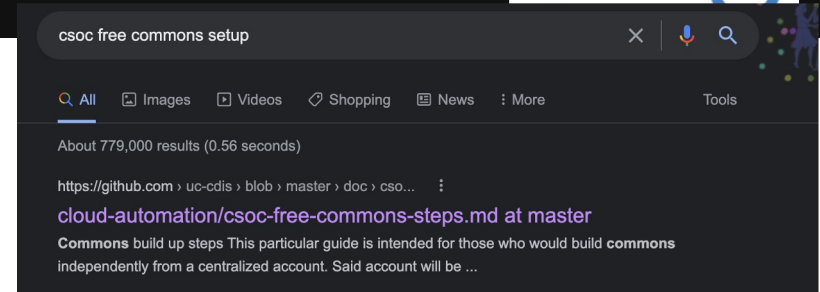
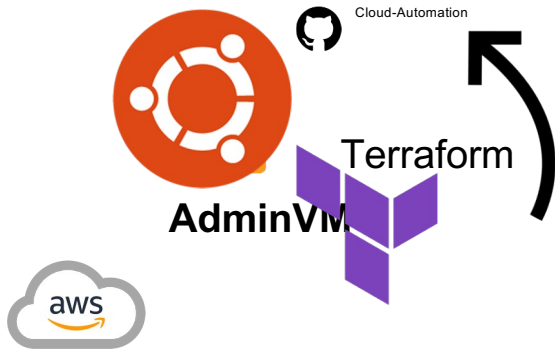
gen3/ # Tools

doc/ # gen3 help docs

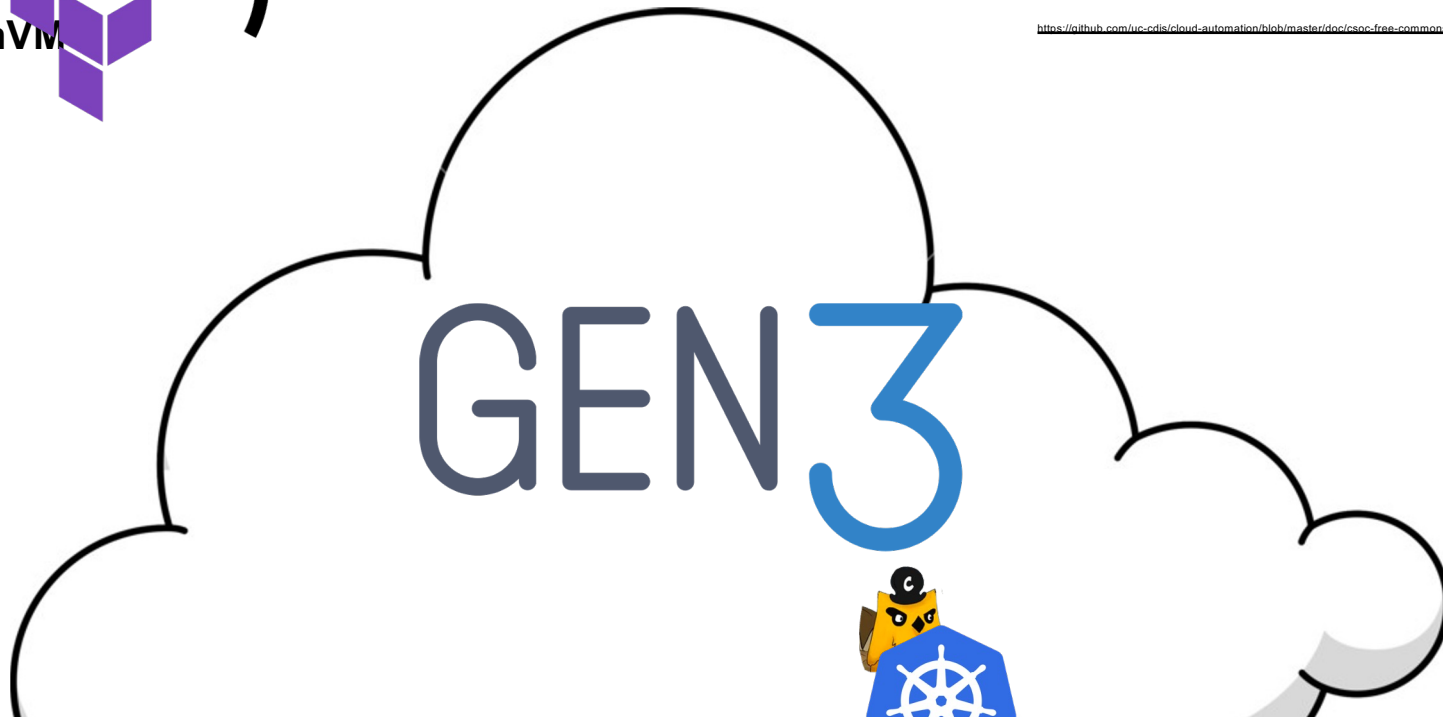
Quick Overview of our AWS deployment

GEN3

Ubuntu 18.04



<https://github.com/uc-cdis/cloud-automation/blob/master/doc/csoc-free-commons-steps.md>



“Cloud-Automation”



- Powerful
- Easy to accrue tech debt!
- Steep learning curve for new developers!
- Hard to just do deployments

- Easier to develop against gen3!
- Helm
 - Helm is a tool that streamlines installing and managing Kubernetes applications
 - Apt/Yum/Homebrew for K8S
 - Easier to deploy to any Kubernetes cluster
 - Deployment on laptop
 - Deployments on more clouds providers
 - Easy to deploy gen3
 - Easy to develop gen3

What is helm?



- Helm is a tool that streamlines installing and managing Kubernetes applications
- Apt/Yum/Homebrew for K8S
- Package into charts
 - Versioned artifact (like docker image)
- Publish to helm repository!
 - URL like helm.gen3.org
- Deploy to any Kubernetes cluster!
- Including your laptops!

- <https://github.com/uc-cdis/gen3-helm>

Q&A



GEN3 Data Commons

Breakout 3: Improving communications and processes around bug and issue tracking

Breakout session goals



We want to address these questions:

- How to communicate that there is a bug?
- How to communicate that you have a fix?
- How to stay aware of the status of bugs?
- Other problems you are encountering

How to communicate that there is a bug?



Current options to report a bug:

- Email support (for production issues)
- Slack #gen3_community channel
- Gen3 forum at <https://forums.gen3.org>
- GitHub repositories issues are currently disabled

How to write a good bug report?

- Environment (browser, software version, website URL...)
- Steps to reproduce
- Expected result
- Actual result (if there is an error, copy the whole error message)
- Logs, if any
- No credentials or PII!

How to communicate that there is a bug?



Our current process:

- Assess whether there really is a bug. Ask for details. Try to reproduce the bug.
- Create an internal ticket
- Prioritize the work to resolve the bug
- Implement and publish the resolution

What can we improve and how?

How to communicate that you have a fix?



Current options:

- Submit a pull request. Share the link in Slack or the forum.

Our current process:

- Create a ticket to review the pull request
- Prioritization
- Review the code changes
- Test the code changes (manual and/or automated tests)
- Make sure the pull request includes updates to the tests and documentation

This can take a while. But we love getting pull requests! We are adjusting our process so we can make more time for this. What can we improve and how?

How to stay aware of the status of bugs?



1. How to track the status of the bug I reported?

Our current process:

- We communicate to the submitter that a ticket has been filed
- We communicate to the submitter once the issue is resolved
 - Directly in the Slack or forum thread
 - or via the weekly Gen3 release notes

What can we improve and how?

How to stay aware of the status of bugs?



2. How to keep track of known issues?

Our current process:

- Bugs are tracked internally
 - How should we notify the community if there's a known defect?
- Bug fixes are in the release notes
 - Weekly Gen3 release notes posted in Slack
 - Monthly release notes <https://github.com/uc-cdis/cdis-manifest/tree/master/releases>

Questions? Recommendations?



Breakout 4: Community Prioritization of New Gen3 Features & Joint Roadmapping

Breakout Session Goals



- Listen to community expectations about contributing to the Gen3 open source codebase
- Find out about any ongoing efforts by groups developing Gen3 code
- Discuss how best to integrate Gen3 code developed outside of UChicago
- Provide insight into Gen3 internal development processes
- Start the process of developing a framework for collaboration among a community of Gen3 developers

- Gen3 is open-source under the Apache license
- The core development is done by the Center for Translational Data Science at the University of Chicago
- The software developers at the University of Chicago are funded by various government contracts and grants and are required to follow the objectives and deliverables associated with each contract
- Gen3 has no general development funding, so we have to align new features with some current contract or grant deliverable or look for new funding to drive new features

Peek behind the curtains



- Internal processes are in a transition period
- New document templates for features
- Moving to API-first architecture
 - Services are implementation details
- Incoming “big” changes to Gen3
 - Data lake
 - Improved workflows and automation for data ingestion
 - Gen3 Front End Framework
 - More clear distinctions between categories of data
 - Workflow execution service, with support for execution of third party containerized pipelines
 - Data mesh

Here's one idea:

- Community members can register a feature proposal
 - It should include some context and focus on desired outcomes and capabilities rather than design
- CTDS will maintain the registry, publicly
 - Helps community members understand what others are working on
 - Enables collaboration and shared requirements
- CTDS will periodically survey the community regarding prioritization of features and publish the results
- CTDS and the community can provide useful standards for feature development

The Community Forum is Resuming in...

10 minutes

10:30 p.m. GMT (5:30 p.m. CT / 9:30 a.m. ADST)

Report from Breakout 1: Standards

- What standards would the community would like to see?
 - Modeling
 - For example: Beacon
 - Access/ Authorization
 - Gen3 user.yaml vs GA4GH Passports
 - Workflows
 - Some commons currently running argo workflows
 - For example: TES API vs WES API
 - Consent Management
 - For example: AAI REMS
- Other discussion topics:
 - Is there a way to generate passports ourselves?

Report from Breakout 2: Standing up a Gen3 Instance

Breakout Room 2: Standing up a Gen3 Instance



Summary/Takeaways:

- Cloud Automation uses Terraform to manage day to day and housekeeping tasks for running a Gen3 instance.
- Cloud automation requires constant updates or else accrue tech debt.
- AdminVM dependency.
- The easier to deploy Gen3, the easier it is to develop Gen3.
- Helm streamlines installation and management of Kubernetes/ Trying to get Helm repo publicly available.
- Adding guides for setting up Telepresence.

Report from Breakout 3: Improving communications and processes around bug and issue tracking

Summary/Takeaways:

- There is a strong need for a ticketing system that can be used by the community.
- Ticketing system should reveal the criticality of the bug - to some extent. The community wants to be able to see, among the population of bugs in the collection of reports, which are simple typos in documentation, vs which are serious errors that cause trouble but do not impact integrity or security, vs crucial defects that do impact integrity or security.
 - We would not want to advertise bugs that are critical defects that we do not yet have a fix for, so we would hide those until a fix was available
- Dashboard for bugs/fixes would be best, but regular communication to community would be acceptable if dashboard is too hard

Report from Breakout 4: Gen3 Features & Future Community Development

Some key takeaways from the discussion



- Some activities - like standing up a new data model - take too long.
- Feature proposal registry was well received.
- Key concern: making it easier for the community to set up and run integration tests and contribute to the integration test suite.
- It would be useful to explore commercial engagement.
- Work needed to figure out how to share contributions that CTDS isn't ready to support. Ideas included community vs. core repos, plug-in architecture, API-first focus.

Gen3 Community Forum 4x aims

1. For the Gen3 core team to share knowledge about Gen3, its architecture, the Gen3 roadmaps and priorities, and the functions of various Gen3 teams (engineering, security, ops, support, and so on).
2. To strengthen the connection between the Gen3 core team and those developing, operating and using Gen3 platforms in order to build awareness of activities, pain points and future plans.
3. To design a set of ongoing community engagement activities - e.g. monthly developer and user meetups - strengthening the connection with community needs.
4. To discuss and agree on key shared development priorities between Gen3 and users - and plan the way forward.

Q&A



GEN3 Data Commons