Gen3.2 - How to build a Gen3 data portal using the new frontend framework

Gen3 Community Forum
1 May 2024
The Agenda

- Introduction
- Gen3.2 Features and Capabilities - Craig Barnes, CTDS
- Migrating from Windmill to Gen3.2 - Matthew Peterkort, Oregon Health & Science University
- Q&A
Gen3.2 Features and Capabilities

Craig Barnes
Center for Translational Data Science
University of Chicago
Gen3 Frontend Framework: Gen3.2

The Gen3 Frontend Framework provides:

- Integrated analysis tools
- Custom content
- Per commons codebase which enables a more flexible source code management and deployment
- Improved user, development, and administration experience
- Upgraded technology stack
- Extensibility and customization
Core Technologies

- Core:
  - React 18
  - Typescript
  - Redux-toolkit
- Application Framework: Next.js 14
- Styling: Tailwind CSS
- UI Components:
  - Mantine.dev
  - mantine react table
- Gen3 Components
- Content:
  - HTML and MDX based static pages
  - Next.js custom page
Gen3.2 Features

- Styling and theming
- Data renderer customization for tables, charts
- Global selection feature: My Data Library
- Analysis tools
- Commons specific pages
- Update designs and improved UX
Gen3.2 Applications

Existing:
- Explorer
- Discovery
- Workspaces
- Data Dictionary
- GraphQL query UI
- Profile
- Data Submission

New:
- Application Center
- My Data Library
- Administration UIs
- LLM Search
- Cohort Discovery
- 3rd party tools and applications
Styling and Theming

The portal’s style can be customized by setting:

- Color theme
- Fonts
- Icons

Configuration tools
Style overrides
Gen3.2’s Color Theme is based on USWDS theme color token: primary/secondary/accent following the proportional 60/30/10 relationship.
Navigation

Configured like data-portal
Customizable icons
3 layouts:

Classic:

Horizontal:
**Gen3 Data Platform**

The Gen3 Data Platform is a single web interface which allows visitors to **discover, access and analyze data**. Making data easily findable enables secondary, cross-study analyses, promotes dissemination of research and accelerates new discoveries.

[EXPLORE GEN3 DATA](#)

The Gen3 platform consists of open-source software services that support the emergence of healthy data ecosystems by enabling the interoperation and creation of cloud-based data resources, including data commons and analysis workspaces. Gen3 aims to accelerate and democratize the process of scientific discovery by making it easy to manage, analyze, harmonize, and share large and complex datasets in the cloud.

[LEARN MORE](#)
Development support

- Striving to make developing commons as simple as possible.
- Initial set of documentation:
  - https://github.com/uc-cdis/gen3-frontend-framework/tree/develop/docs
- .env files for configuration
- Fine grain connection to remote Gen3 services
- Credentials based login
- Run Gen3.2 development outside of helm charts
- Local revproxy instruction
- Supported in gen3-helm charts
- Continuously refining development support
Environment Files

NextJS supports `.env` files for various deployments:
- `.env.development`
- `.env.production`

You can add more:
- `.env.testing`
- `.env.staging`

```
GEN3_COMMONS_NAME = gen3
NEXT_PUBLIC_GEN3_API = https://localhost:3010
```

Pending:
- Set env variables/config in helm `values.yaml`
Environment Files

Override any Gen3 endpoint

NEXT_PUBLIC_GEN3_FENCE_API=https://localhost:3010
NEXT_PUBLIC_GEN3_DOMAIN=https://localhost:3010
NEXT_PUBLIC_GEN3_MDS_API=https://localhost:3010/mds
NEXT_PUBLIC_GEN3_AI_SEARCH_API=https://localhost:3010/ai
NEXT_PUBLIC_GEN3_GUPPY_API=https://localhost:3010/guppy
NEXT_PUBLIC_GEN3_FENCE_API=https://localhost:3010
NEXT_PUBLIC_GEN3_AUTHZ_API=https://localhost:3010/authz
NEXT_PUBLIC_GEN3_WORKSPACE_STATUS_API=https://localhost:3010/lw-workspace
NEXT_PUBLIC_GEN3_SUBMISSION_API=https://localhost:3010/api/v0/submission
Credentials based login

Mostly for development at CTDS:

- Credentials based login
- Login with no fence redirect
- Connect FE to remote commons
- Only in development environment
- Scope restrictions on token
Explorer/Cohort Builder

Extensible to custom data renderer for table cells and charts
Gen3.2 implementation of Guppy UI components

Pending work:
- Elimination of filter tabs
- Support for sharing selection with the same facet on other indexes
- Subtable in main table.
- Row details
- Authz based filtering
New Explorer UI
Explorer in action: imaging-hub
Discovery Page for metadata browsing and searching
Support for custom renderers for table cells and rows
Once registered the cell renders can be added to the configuration
Define a data hook for retrieving and processing metadata from the Gen3 MDS
Basic functionality is working for all components
Todo:
- Advanced search
- Details page
- Selection
- Actions: export to workspace
Gen3.2 support extensibility using data renderers
Enables data to have visual representation in Tables and Charts

Overview of the process:

1. Write a function that takes a value and returns a ReactElement
2. Register function:
   ```javascript
   DiscoveryCellRendererFactory.registerCellRendererCatalog({...});
   ```
3. Add the function to the configuration:
   ```json
   "field": "__manifest",
   "contentType": "manifest",
   "cellRenderFunction": "inline",
   ```
### Data Renderer Example

**Inline chart renderer** __manifest field Discovery Table

<table>
<thead>
<tr>
<th>Study ID</th>
<th>Study Name</th>
<th>Study Title</th>
<th>Source</th>
<th>Date Paid</th>
<th>Subjects</th>
</tr>
</thead>
<tbody>
<tr>
<td>1906_Genomes_Project</td>
<td>1010_Genomes_Project</td>
<td>1000_Genomes_Project</td>
<td>1000_Genomes_Project</td>
<td>0</td>
<td></td>
</tr>
</tbody>
</table>

The 1000 Genomes Project is a collaborative effort involving research groups in the UK, US and other countries to produce an extensive catalog of human genetic variation that will support future medical research studies. It will expand the data from the International HapMap Project, which provided a resource that has been used to identify more than 150 regions of the genome that are associated with common diseases such as coronary artery disease.

**ACCOUNT_Collapsed_Arm**

Discovery_Collapsed_Arm

Discovery_Collapsed_Arm

ACCOUNT_Collapsed_Arm

**CCLE (Cancer Cell Line Encyclopedia)**

<table>
<thead>
<tr>
<th>Cell Line</th>
<th>CCLE (Cancer Cell Line Encyclopedia)</th>
<th>CCLE (Cancer Cell Line Encyclopedia)</th>
<th>Broad Institute</th>
</tr>
</thead>
<tbody>
<tr>
<td>13292816</td>
<td>13292816</td>
<td>13292816</td>
<td>13292816</td>
</tr>
</tbody>
</table>

The CCLE (Cancer Cell Line Encyclopedia) project is a collaboration between the Broad Institute and the Novartis Institutes for Biomedical Research and its Genomics Institute of the Novartis Research Foundation to conduct a detailed genetic and pharmacologic characterization of a large panel of human cancer models, to develop integrated computational analyses that link distinct pharmacologic vulnerabilities to genomic patterns and to...

**UCLA Consortium for Neurobiophychiatric Phenomics**

<table>
<thead>
<tr>
<th>Study</th>
<th>UCLA Consortium for Neurobiophychiatric Phenomics</th>
<th>UCLA Consortium for Neurobiophychiatric Phenomics</th>
<th>OpenNeuro</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456789</td>
<td>123456789</td>
<td>123456789</td>
<td>123456789</td>
</tr>
</tbody>
</table>

The Consortium for Neurobiophychiatric Phenomics (CNP) is a large study funded by the NIMH Bootstrap Initiative that aims to facilitate discovery of the genetic and environmental bases of variation in psychological and behavioral traits. It includes neuroimaging analyses to elucidate the mechanisms that link the human genome to complex psychological conditions, and to foster breakthroughs in the development of novel treatments for neuropsychiatric disorders. 18...

**Gene Networks Specific for innate immunity Dilemmas**

<table>
<thead>
<tr>
<th>Network</th>
<th>Gene Networks Specific for innate immunity Dilemmas</th>
<th>Gene Networks Specific for innate immunity Dilemmas</th>
<th>IntraSchool of Medicine at Mount Sinai</th>
</tr>
</thead>
<tbody>
<tr>
<td>123456789</td>
<td>123456789</td>
<td>123456789</td>
<td>123456789</td>
</tr>
</tbody>
</table>

The molecular factors involved in the development of Post-Traumatic Stress Disorder (PTSD) remain poorly understood. Previous transcriptomic studies investigating the mechanisms of PTSD apply targeted approaches to identify individual genes under a cross-sectional framework lacking an holistic view of the behavioral and psychological properties of these genes at the system level. Here we sought to apply an unsupervised gene node-based approach to...
Data Renderer Example

Popup chart renderer

manifest field Discovery Table
The application framework of Gen3.2 is NextJS (vers 14.1)
Adding new pages is a matter of adding a tsk file to `src/pages`:
There is a template in `pages/SamplePage.tsx`

```javascript
const SamplePage = ({ headerProps, footerProps }: NavPageLayoutProps) => {
  return (
    <NavPageLayout {...{ headerProps, footerProps }}>{
      <div className="w-full m-10">
        <Center>
          <Paper shadow="md" p="xl" withBorder>
            <Text>This is an example custom page in Gen3</Text>
            <Text>You can add your own content here, and add a link to this page in
                 the navigation bar by editing the config file in
                 navigation.json
            </Text>
          </Paper>
        </Center>
      </div>
    </NavPageLayout>
  );
};
```
COVID-19 X-ray images classification

Author: [Name] Date: [Date]

Introduction:

- The novel coronavirus (COVID-19), which first appeared in Wuhan, China in December 2019, spread rapidly around the world and became a global pandemic. This resulted in a devastating effect on both daily lives, public health, and the global economy. It is critical to detect the positive cases as early as possible to prevent the further spread of this epidemic and to quickly treat infected patients. The need for auxiliary diagnostic tools has increased as there are no accurate automated tools available. Recent findings obtained using radiology imaging techniques suggest that such images contain salient information about the COVID-19 virus. Application of advanced AI techniques coupled with radiological imaging can be helpful for the accurate detection of this disease, and can also be assistive to overcome the problem of a lack of specialized physicians in remote villages.

- This notebook is a replication of the DarkCovidNet model and transfer learning of DooGNet model to classify radiological images as indicative of either "COVID-19" ("Fever"") and "Pneumonia" to assist in the diagnosis of COVID-19. This is an extension of the notebooks published along with the paper in [arXiv](https://arxiv.org/abs/2003.06475) by [Yi et al., 2020: Automated detection of COVID-19 cases using deep neural networks with X-ray].


Set up notebook

```
# Install necessary packages
!pip install fastai==2.0.3 --user --no-cache-dir
# Install prerequisites
!pip install matplotlib
!pip install pillow
!pip install seaborn
!pip install numpy

from fastai.vision import *
from fastai import
```

The above code installs the necessary packages and sets up the environment for data visualization and analysis.

Download X-ray images from Kaggle API

```
```

```
from fastai.vision import *
```

```
!unzip xray_images.zip
```

```
!ls xray_images
```

```
# Convert images to proper format
import os
os.chdir('xray_images')
for dir in os.listdir('.'): os.rename(dir, dir.replace('-','_'))
```

```
from fastai.vision import *
```

```
!ls xray_images
```

```
# Perform transfer learning
from fastai.vision.models import
```

```
!ls xray_images
```

```
# Classify images using the model
from fastai.vision import *
```

```
!ls xray_images
```

```
# View classified images
```

```
!ls xray_images
```

```
# Analyze results
```

```
!ls xray_images
```

```
# Evaluate model performance
```

```
!ls xray_images
```

```
# Save results
```

```
!ls xray_images
```

```
# Export results to CSV
```

```
!ls xray_images
```

```
# Visualize results
```

```
!ls xray_images
```

```
# Further analysis...
```

```
!ls xray_images
```

```
# Final results...
```

```
!ls xray_images
```

This notebook provides a comprehensive guide to using radiological images for COVID-19 diagnosis, utilizing advanced AI techniques and transfer learning from existing models.
Running with data-portal

Possible to run data-portal simultaneously with Gen3.2

In helm charts (currently on a gen3-helm branch)

- Add configuration for frontend-framework and data-portal in value.yaml
- Add `frontendRoot: gen3ff`

Gen3.2 will be on / and data-portal will be on /portal

`frontendRoot: portal`

data-portal will be on / Gen3.2 will be on /ff
Roadmap

May 2024:
- Workspaces
- Update explorer UI
- Discovery page improvements

June 2024:
- Analysis Center
- My Data Library
- Data Dictionary

July 2024:
- New application
- Homepage layouts
- Data Submission

Remaining:
- Eliminate issues and flaws
- Testing framework
- Accessibility compliance
- Documentation
- Improved deployment configuration
- Modals
- Nextjs app router
- SSR components
- Simplify configuration:
  - admin UI
  - config validation
- …
Migrating from Windmill to Gen3.2

Matthew Peterkort
Oregon Health & Science University
Local Development

https://github.com/uc-cdis/gen3-frontend-framework/blob/develop/docs/Local%20Development/QuickStart.md

Gen3.2 Frontend Quickstart

This guide will get you up to speed on configuring a Gen3 Frontend Applications. There are other tutorial on running it locally (for example: (Local Development)[]).

🔗 Getting Started

Gen3 Data Commons using the Gen3 Frontend Framework is a matter of the following:

- create a clone of Gen3 Data Commons Application
- Configure the commons by editing the configuration files in the config directory.
- Add your pages and content
- Deploy via helm charts or Docker.

Changes to the Gen3 Data Commons Application can be pulled from the Common Frontend Repository. You need configure git to pull from the Common Frontend Repository.

```bash
  git remote add upstream https://github.com/uc-cdis/commons-frontend-app.git
```
Local Development Setup

- **NVM** is a painless way of managing Node versions
- Gen3 helm kubernetes instance required
- It is important to understand that the local development setup passes portal requests to your local helm nginx setup
- If strictly following the docs using example revproxy config make sure you change your values.yaml hostname to be localhost
- If you wish to customize and add to gen3-frontend-framework fork from: https://github.com/uc-cdis/commons-frontend-app
- If you wish to develop and make contributions to frontend framework fork from: https://github.com/uc-cdis/gen3-frontend-framework/tree/develop
- See gen3-frontend/framework/docs/Local Development/Using Helm Charts/ Local Development with Helm Charts on the develop branch for setup docs.
Deployment Setup

- Gen3 Helm supports frontend framework deployments with minimal changes to existing helm gen3 deployments: [https://github.com/uc-cdis/gen3-helm/pull/160](https://github.com/uc-cdis/gen3-helm/pull/160)
- Make sure that “npm run build” and “npm run start” work as expected before building a custom image.
- Separate production and development environment variables are defined in the sample Commons. Make sure that the production hostname matches environment defined in helm and in the frontend-framework.
Sample Commons Directory Layout

- **src** - Top level source code pages imported from @gen3/frontend
- **public** - The actual custom content files
- **Config** - Highly customizable settings files that control the look and feel of the website
Config Directory – Global Configurations

```json
packages > sampleCommons > config > {} session.json > ...
{
    "sessionConfig": {
        "updateSessionTime": 5,
        "inactiveTimeLimit": 20,
        "logoutInactiveUsers": false
    }
}
```

```json
packages > sampleCommons > config > {} siteConfig.json > ...
{
    "commons": "gen3"
}
```

```json
packages > sampleCommons > config > {} themeFonts.json > ...
{
    "heading": ["FranklinGothic", "Montserrat", "sans-serif"],
    "content": ["FranklinGothic", "Noto Sans", "sans-serif"],
    "fontFamily": "FranklinGothic"
}
```
Page Level Configurations

```json
packages > sampleCommons > config > gen3 > {} login.json > [ ] topContent

1
2
3
4
5
6
7
8
9
10
11
12
13
14
15
16
17
18
19
20
21
22
23
24
25

"topContent": [],

{ "text": "ACED",
  "className": "text-center text-3xl font-bold"
},

{ "text": "DISCOVER, ANALYZE, AND SHARE DATA",
  "className": "text-center text-xl font-medium"
},

{ "text": "We are uniting world leading researchers to tackle the most critical clinical need. Scientists in the Alliance are working together at the forefront of technological innovation to translate research into realistic ways to improve cancer diagnosis, which can be implemented into health systems and meaningfully benefit people with cancer.

If you have any questions about access or the registration process, please visit aced.dphotech.com",
  "className": "text-center text-xl font-medium"
},

"bottomContent": [],

{ "text": "If you have any questions about access or the registration process, please visit aced.dphotech.com",
  "className": "text-center text-sm"
},

"image": "images/gene.svg",
"showCredentialsLogin": true
```
Page Level Configurations

ACED

We are uniting world leading researchers to tackle the biggest challenges in early detection, an important area of unmet clinical need. Scientists in the Alliance are working together at the forefront of technological innovation to translate research into realistic ways to improve cancer diagnosis, which can be implemented into health systems and meaningfully benefit people with cancer.

EXPLORE GEN3 DATA


LEARN MORE
Gitops Migration

- Backwards compatible config files
- Migration from gitops to FF is as simple as a couple of copy and pastes
- Expanded configurations for landing page, colors, profile and others.
- Many configs map straight across to the expected json file in sampleCommons/config/gen3
Adding a New Font

- Add font file to public/fonts
- Add path to font file to globals.css
- Add font name to config/themeFonts.json for corresponding text types
**Before / After**

**Gen3 Data Platform**

The Gen3 Data Platform is a single web interface which allows visitors to discover, access and analyze data. Making data easily findable enables secondary, cross-study analyses, promotes dissemination of research and accelerates new discoveries.

[Explore Gen3 data](#)

The Gen3 platform consists of open-source software services that support the emergence of healthy data ecosystems by enabling the interpretation and creation of cloud-based data resources, including data commons and analysis workspaces. Gen3 aims to accelerate and democratize the process of scientific discovery by making it easy to manage, analyze, harmonize, and share large and complex datasets in the cloud.

[Learn more](#)

---

**ACED**

An international alliance of leading researchers to tackle the biggest challenges in early detection, the importance of early detection for cancer. Scientists in the alliance work together across the breadth of technological innovation to translate research into realizable, practical improvements in cancer diagnosis, which can be implemented into health systems and extensively benefit people with cancer.

[Explore Gen3 data](#)


[Learn more](#)
Adding a New Page

- Add a new .tsx file to `gen3-frontend-framework/packages/sampleCommons/src/pages`.
- `SamplePage.tsx` is given as a template for adding new pages.
- `<NavPageLayout>` component and props serve as boilerplate, ex:
Linking to Navigation Bar

- Add a link to SamplePage.tsx to sampleCommons/config/gen3/navigation.json, or anywhere else that allows linking pages
- To do this add a dict entry to “items” list as shown in right-hand image
- Supports full customization of icon, Name, and tooltip
- SamplePage is attached to the “DEMO” button in right-hand example
Guppy Api Fetch

- Guppy API fetching function part of the @gen3/core library
- More advanced fetching techniques are used in the cohort builder.
- Api endpoints fetching techniques are also supported in the core library

```javascript
/**
 * Similar to graphQLAPI except this
 * guppyAPIFetch function is a branch of guppy core API defined below and
 * graphQLAPI is a branch of gen3Services API.
 */

export const guppyAPIFetch = async <T>(
  query: guppyApiSliceRequest,
): Promise<guppyApiResponse<T>> => {
  const res = await fetch(`${GEN3_GUPPY_API}/graphql`, {
    headers: {
      Accept: 'application/json',
      'Content-Type': 'application/json',
      'Access-Control-Allow-Origin': '*',
    },
    method: 'POST',
    body: JSON.stringify(query),
  });
  if (res.ok) return res.json();
  throw await buildGuppyFetchError(res, query);
};
```
Uses a basic `useEffect` hook to fetch data and use data in visualization

- Guppy expects query format

```javascript
export interface guppyApiSliceRequest {
    readonly query: string;
    readonly variables?: Record<string, unknown>;
}

export interface guppyApiResponse<H = JSONObject> {
    readonly data: H;
    readonly errors: Record<string, string>;
}
```
Guppy Output Processing

- Data Component used for processing and mapping data into a grid of “reference” entries.
- Essentially a grid of guppy results where each result is a “reference” to a patient
Putting it all together

- Code demonstrates using basic JS hooks, fetching functions, and Mantine components to fetch data from Guppy
- Guppy data is displayed as a grid of entries
- Header and footer props are maintained to keep look and feel consistent with the rest of the site
Putting it all together
Acknowledgements

● Speakers
  ○ Craig Barnes - Center for Translational Data Science, University of Chicago
  ○ Matthew Peterkort - Oregon Health & Science University

● Gen3 Forum Steering Committee
  ○ Robert Grossman - Center for Translational Data Science, University of Chicago
  ○ Steven Manos - Australian BioCommons
  ○ Claire Rye - New Zealand eScience Infrastructure
  ○ Plamen Martinov - Open Commons Consortium
  ○ Michael Fitzsimons - Center for Translational Data Science, University of Chicago