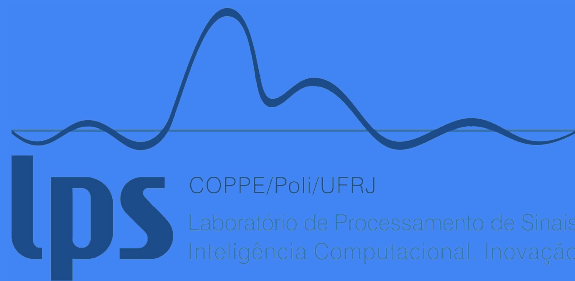# The migration to a standardized architecture for developing systems on the Glance project

Norfolk, US - May 2023 | CHEP 2023

Carlos Henrique Ferreira Brito Filho
Gabriel José Souza e Silva
Joel Closier
Gloria Corti

UNIVERSIDADE FEDERAL
DO RIO DE JANEIRO

lps
COPPE/Poli/UFRJ
Laboratório de Processamento de Sinais
Inteligência Computacional. Inovação

LHCb

The Glance Project logo

The Glance Project
- Active since 2003
- Used by 3 CERN experiments
- +20 web systems

FENCE

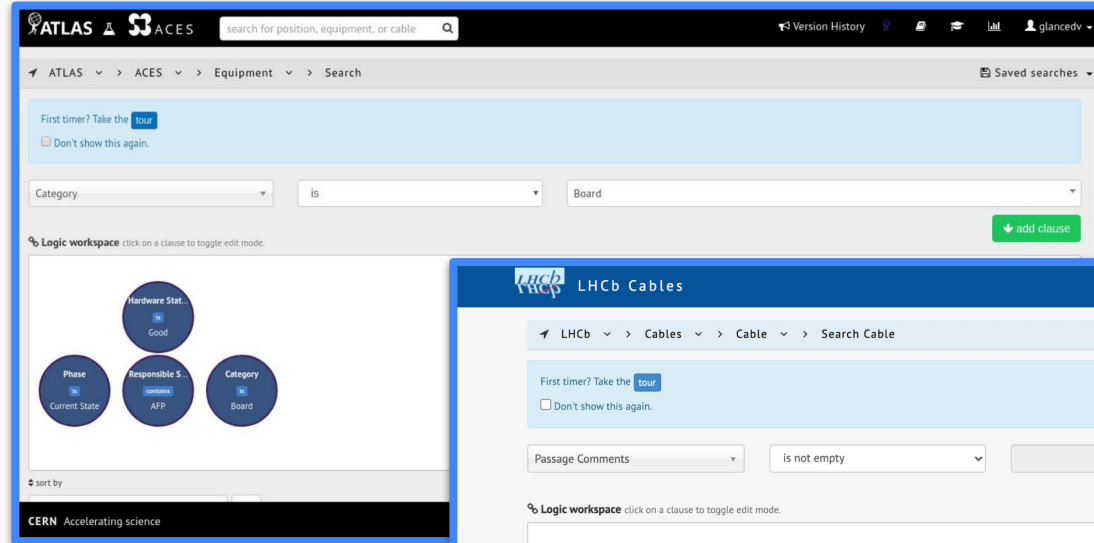The FENCE (**F**ront-end **EN**gine for gla**NCE**) framework
- Easily configurable with the use of .json files
- First attempt at a standard on the Glance project

```
$fence = new \Fence\Fence();

$dbmanager = new DBManager();

$fence->add_content(
    new CableSearchInterface(
        "cable_super_search_model.json",
        "cable_super_search_view.json",
        "source/cable.json",
        $dbmanager
    )
);

$fence->render();
```
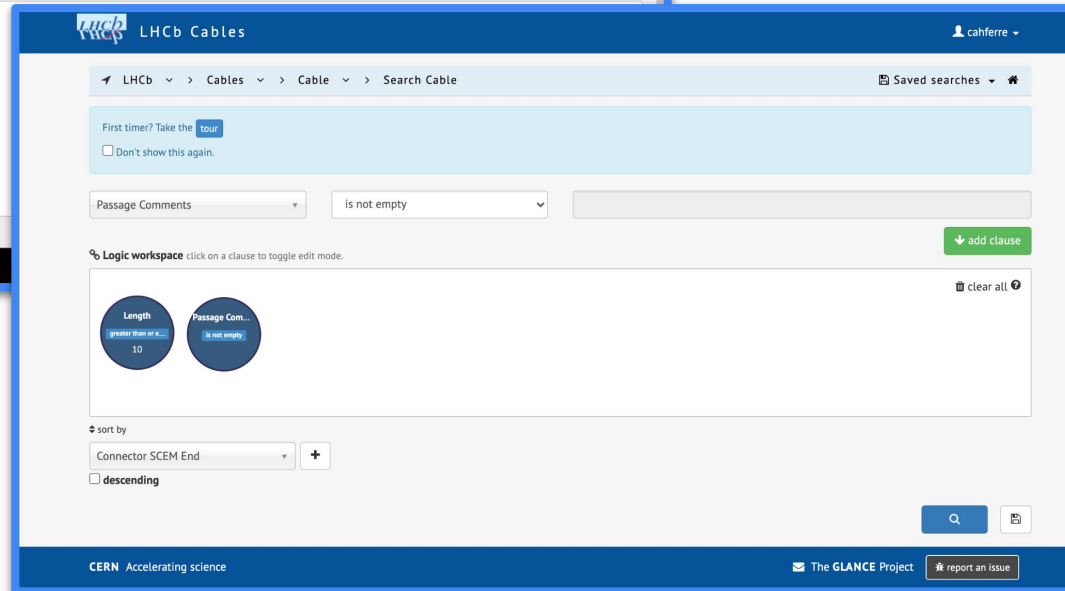
Fence framework

FENCE - Issues

⚠️ Issues

- Limited customization

FENCE - Issues

25 (30-39) FENCE systems in production

40,754 FENCE lines of code

65 core FENCE classes Total of 6,404

~100 FENCE Jira closed tickets per month

~15k FENCE (ATLAS e ALICE) sessions per month

FENCE statistics in 2018
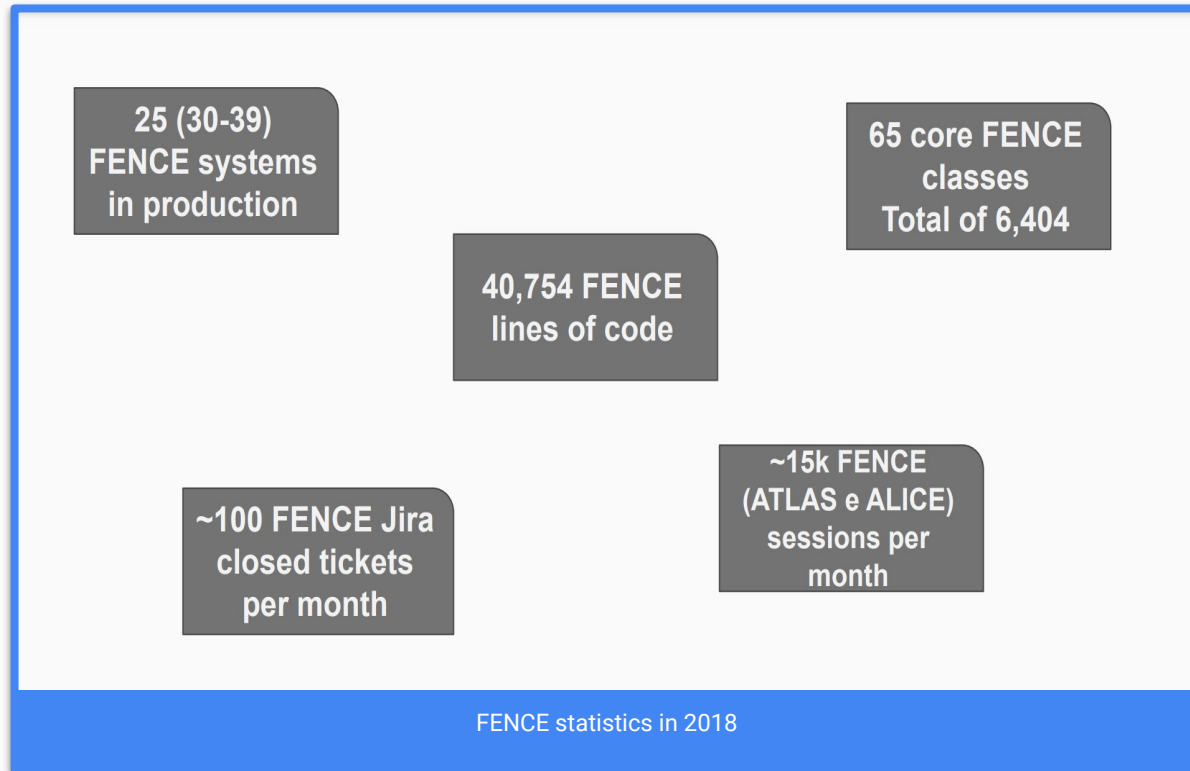
⚠️ Issues

- Lack of documentation

FENCE - Issues

Builders/Factories

Configuration File

REUSE

FENCE framework
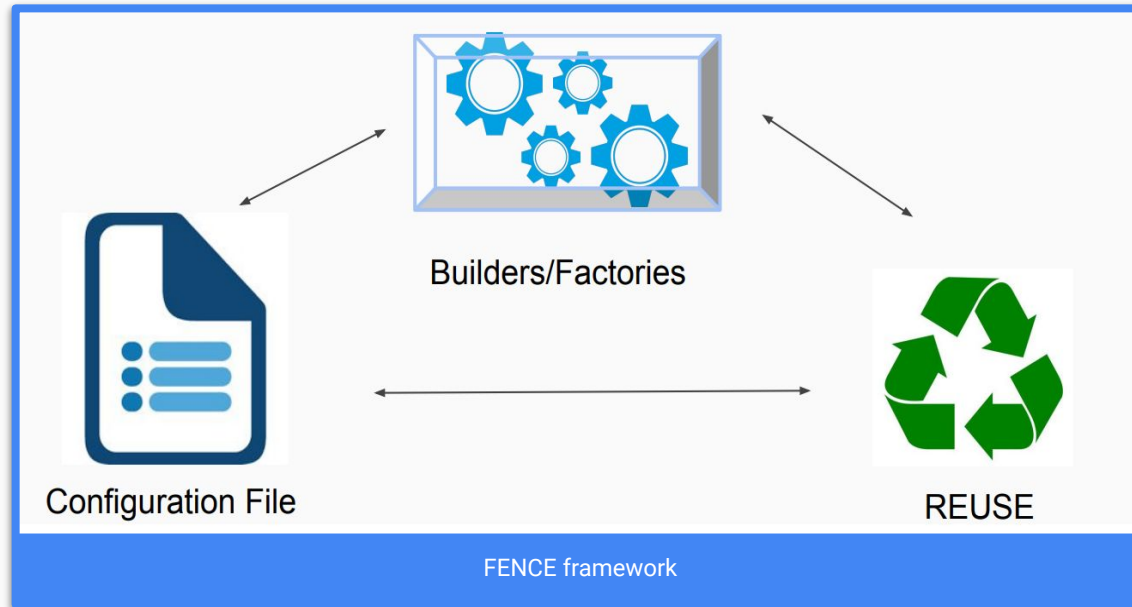
⚠️ Issues

- High coupling
  - not using Glance as an API
  - no separation of concerns
  - in-house built tool used for fetching data

```
$dbmanager = new DBManager();

$members = FacTree::envBuild('member')
    ->order([
        'firstName',
        'lastName',
        'initials',
        'membershipAccessId'
    ])
    ->build($dbmanager, $id);

return $members;
```
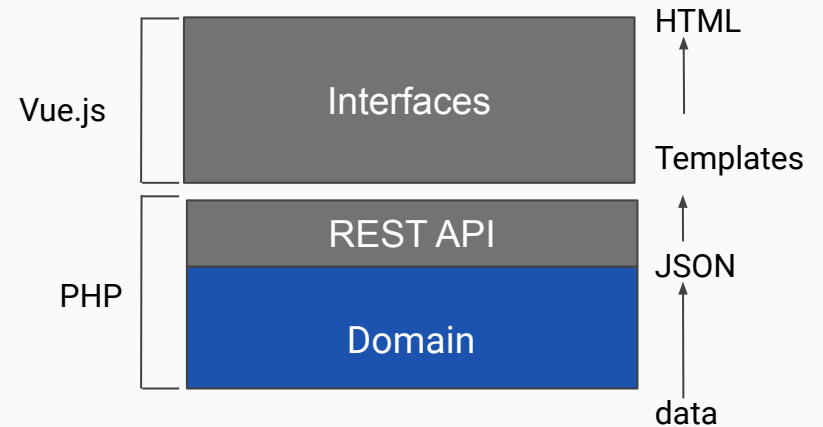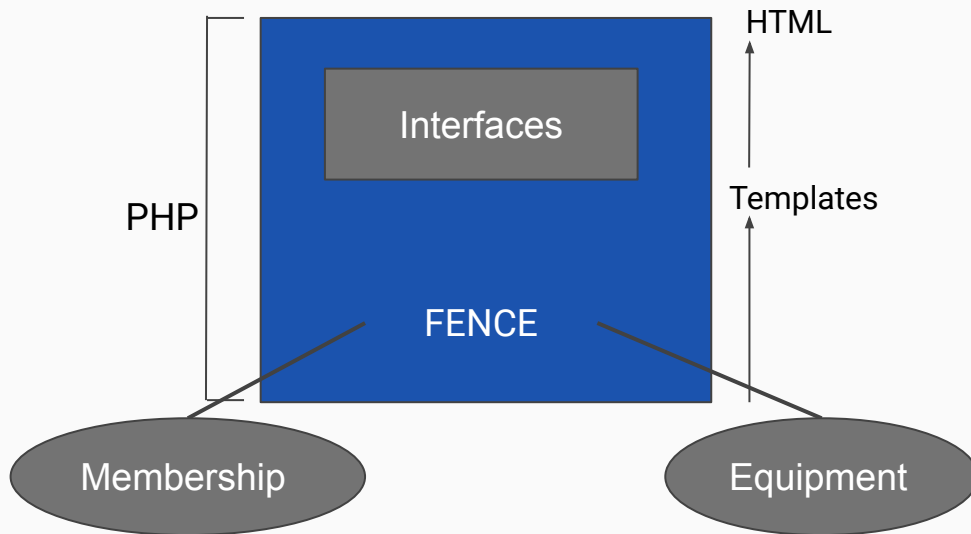
```json
{
    "table": "ME_MEMBER_VIEW",
    "primary_key": "id",
    "model": "\\LHCb\\Membership\\Models\\Member",
    "components": {
        "firstName": {
            "column": "FIRST_NAME",
            "carrier": "setFirstName"
        },
        "lastName": {
            "column": "LAST_NAME",
            "carrier": "setLastName"
        },
        "orcid": {
            "column": "ORCID",
            "carrier": "setOrcid"
        },
        ...
}
```
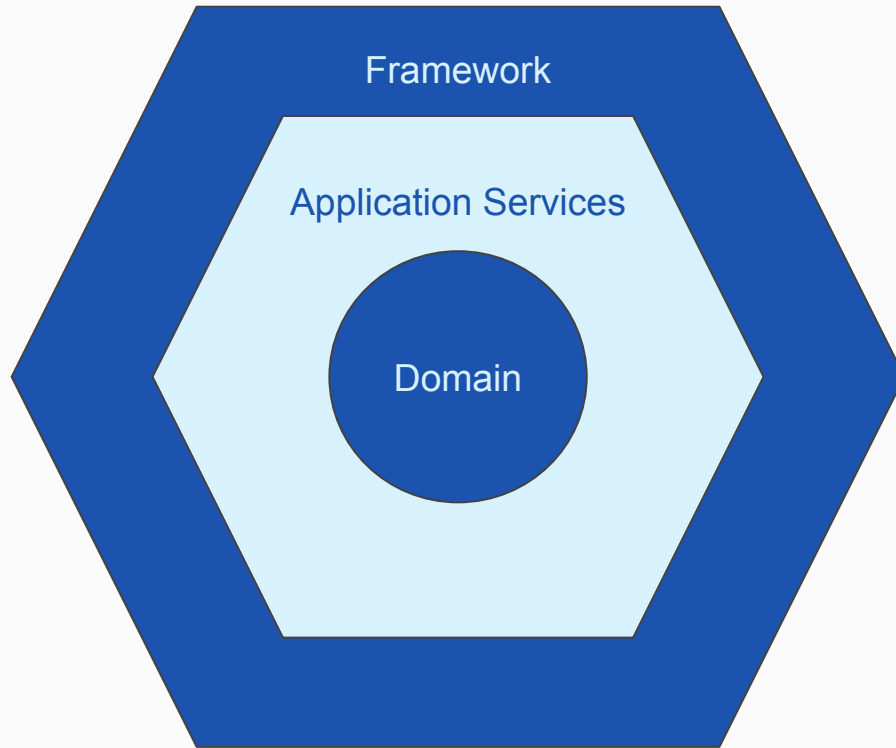
⚠️ Issues

- Glance's in-house built Object-Relational-Mapper (ORM): FacTree

New architecture

- Hexagonal Architecture
- Backend:
  - REST API
- Frontend:
  - Modular and reusable components

New architecture

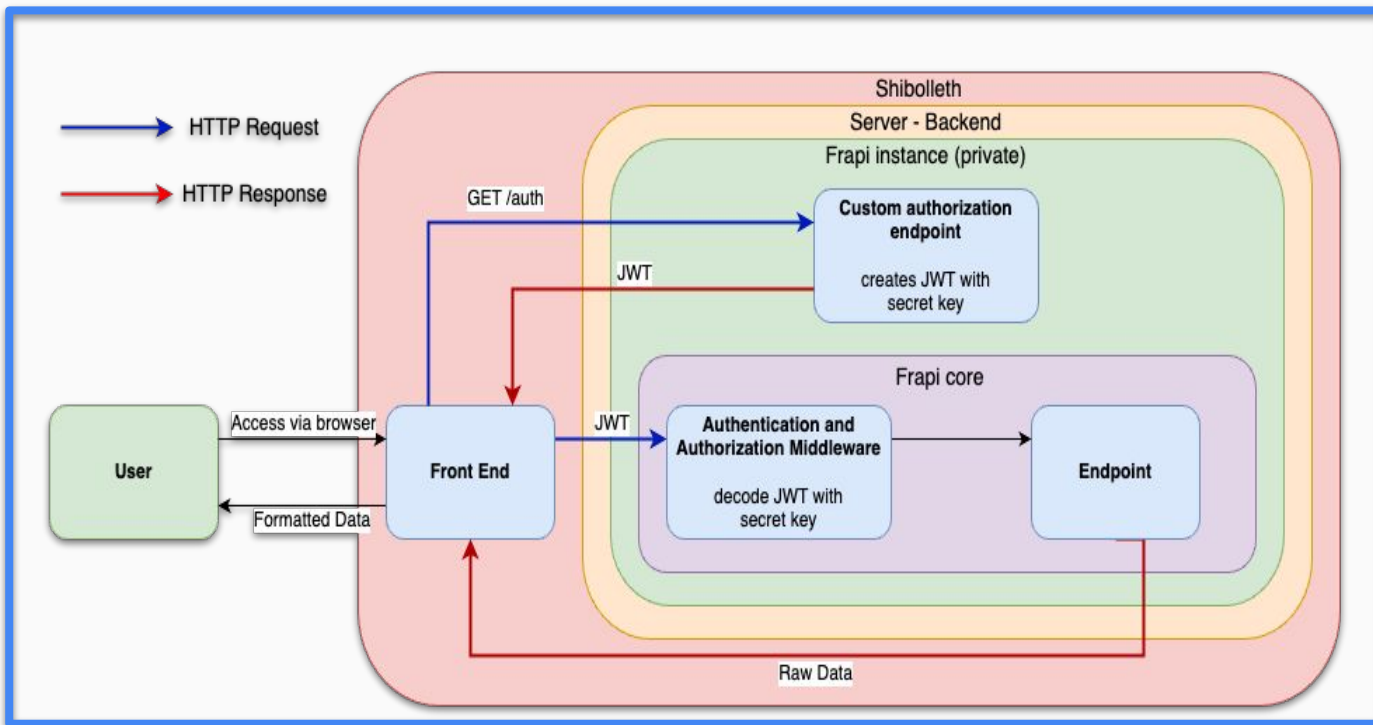Hexagonal architecture
- Loosely coupled concerns
- Flexibility
- Scalability

Framework

Application Services

Domain

Backend: FENCE REST API (FRAPI)
- Handling authorization with CERN's services
- Building endpoints
- Logging HTTP requests

FENCE REST API (FRAPI)
- Handling authorization with CERN's services
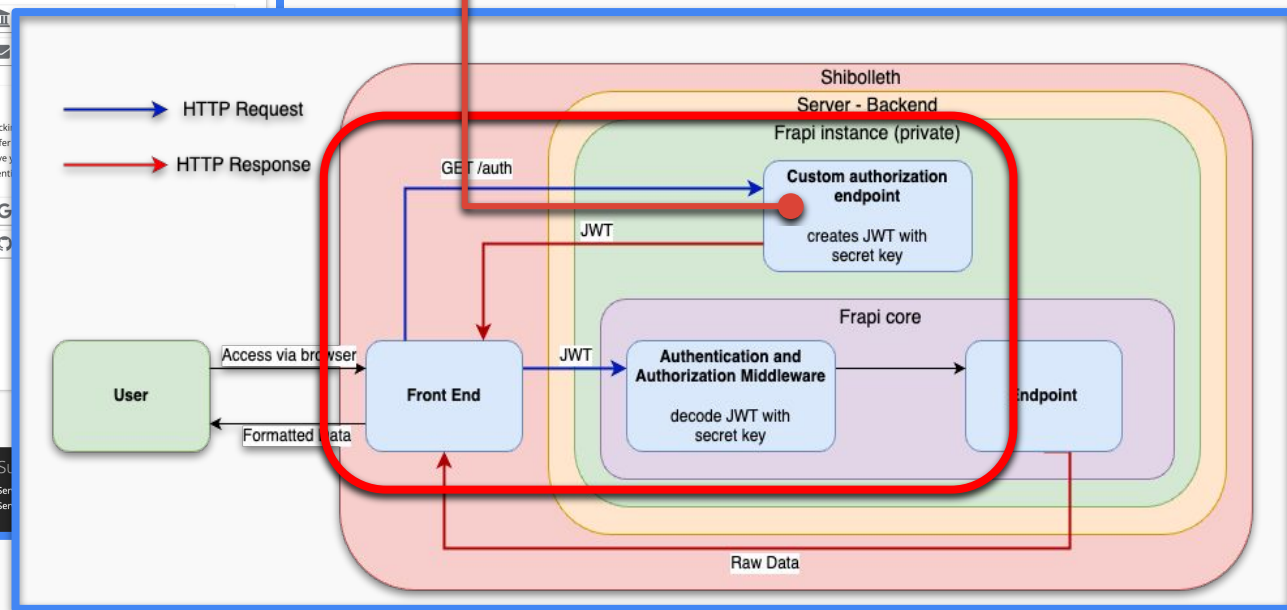
FRAPI endpoints

```php
use Fence\Frapi\Api;

$api = new Api("configuration/api.json");
$api->setContainer($container);
$api->run();
```

FRAPI setup

FENCE REST API (FRAPI)
- Building endpoints
  - Easy setup
  - Highly configurable

```json
{
    "auth": {
        "sso": "keycloak"
    },
    "user": {
        "class": "LHCb\\Membership\\User"
    },
    "endpoints": {
        "folders": [
            "configuration/routes"
        ]
    }
}
```

FRAPI configuration file

```json
{
    "type": "controller",
    "class": "\\Member\\Infrastructure\\Web\\MemberController",
    "paths": {
        "/members": {
            "GET": {
                "method": "findAllMembersBasicInformation"
            },
            "PATCH": {
                "method": "insertMember",
                "schema": "resources/schemas/member-update.json"
            }
        }
        ...
    }
}
```
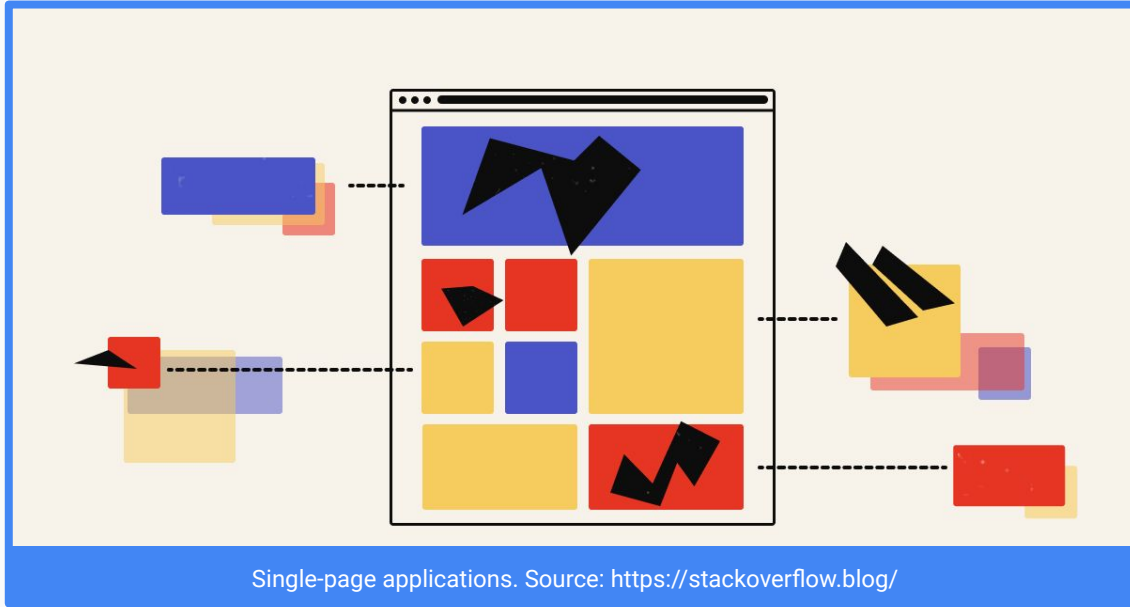
FRAPI endpoint example

FRAPI logging

FENCE REST API (FRAPI)
- Logging HTTP requests

Frontend

Single-page applications. Source: https://stackoverflow.blog/

Qualities sought:
- Open-source
- Well documented
- Reusable components

Frontend technologies

Frontend technologies

- Vue.js
- Vue libraries
  - Vuetify
  - Vuex
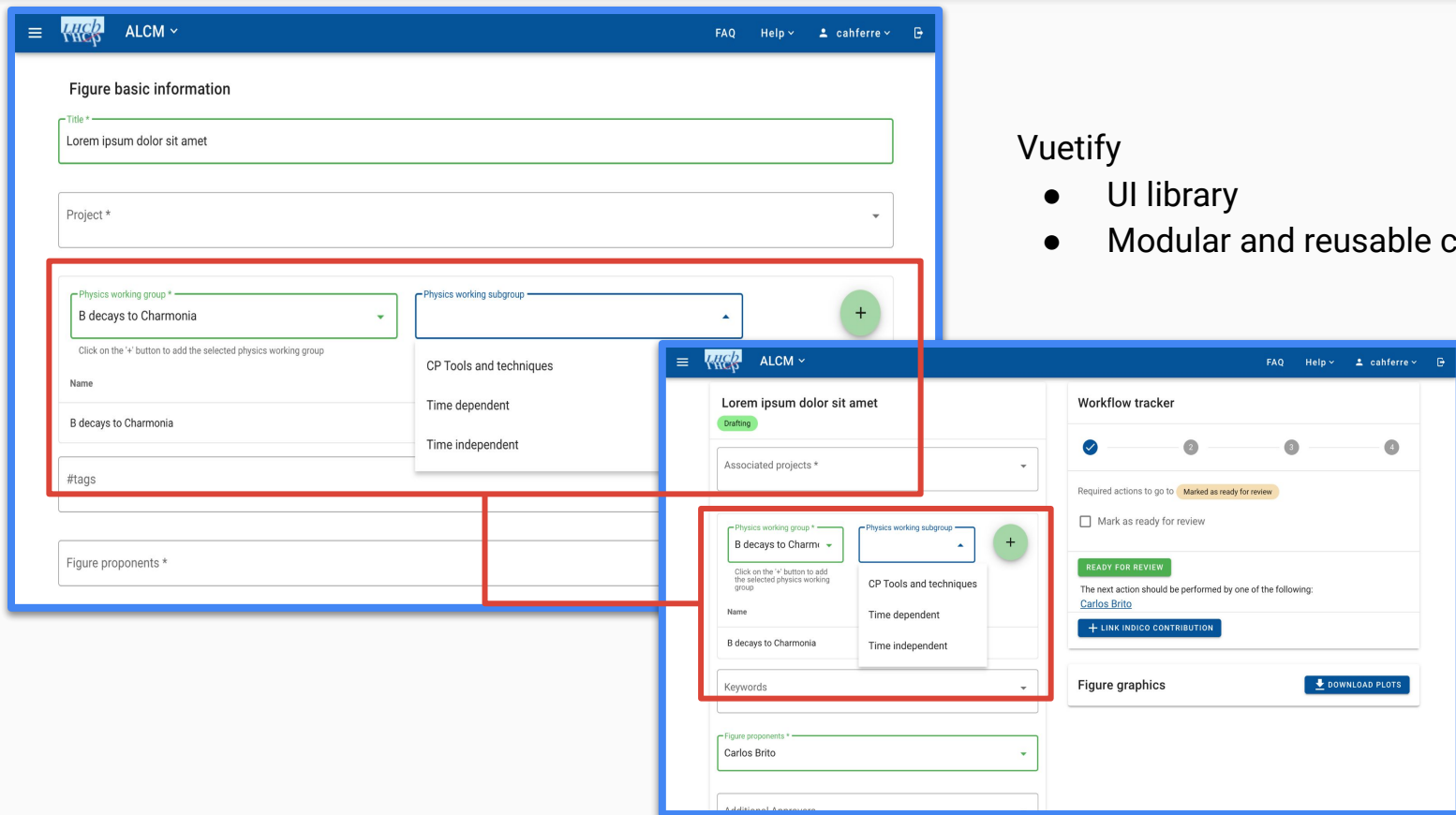
Vuetify

Vuetify
- UI library
- Modular and reusable components

Vuex

```
const getters = {
    member: state => state.member,
    loadingMember: state => state.loadingMember,
    ...
};
```

```
const actions = {
    async fetchMember({ commit }, memberId) {
        commit('SET_LOADING_MEMBER', true);
        const response = await memberApi.getMember(memberId);
        commit('SET_MEMBER', response.data.member);
        commit('SET_LOADING_MEMBER', false);
        return response;
    },
    ...
};
```

```
const mutations = {
    SET_MEMBER(state, member) {
        state.member = member;
    },
    SET_LOADING_MEMBER(state, loadingMember) {
        state.loadingMember = loadingMember;
    },
    ...
};
```

Vuex
- State management library
- Reactive container that holds the application state

Glance in the future

Past Events	Currently	Future

2020:
- Started developing the new architecture

2021:
- Launch of the first system in the new architecture: LHCb authorship

2023:
- Adopting the standard across the experiments
- Removing FENCE as a dependency from our applications

Future:
- Implement new systems with the new architecture
- Implement a more robust logging solution

Thanks!

Glance project presentations at CHEP 2023:
- The ALICE Glance Membership system
    - Poster session, Poster #13, 15:30 - 16:30
- The ALICE Glance Service Work system
    - T8 May 8th, 2023, 15:00 - 15:15
- Iterative and incremental development of the ATLAS Publication Tracking system
    - T5 May 9th 2023, 14:00 - 14:15
- Enhancing data consistency in ATLAS and CERN databases through automated synchronization
    - T5 May 9th 2023, 14:15 - 14:30
- Glance Search Interface
    - T5 May 9th 2023, 14:30 - 14:45

Contact: carlos.brito@cern.ch