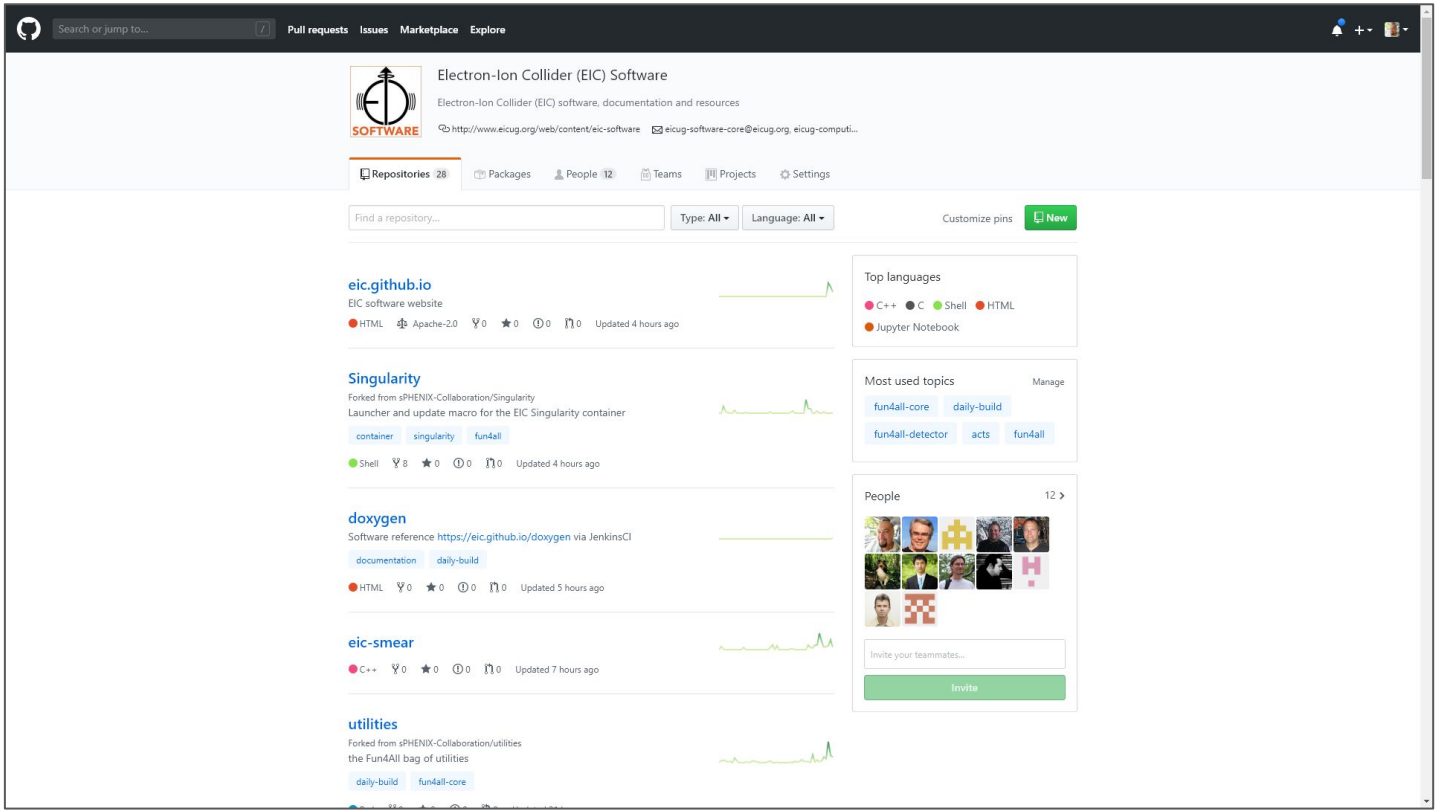
EIC Software on GitHub: The Repositories and the Website

Maxim Potekhin
(BNL, NPPS)
04/22/2020

The EIC Software on GitHub



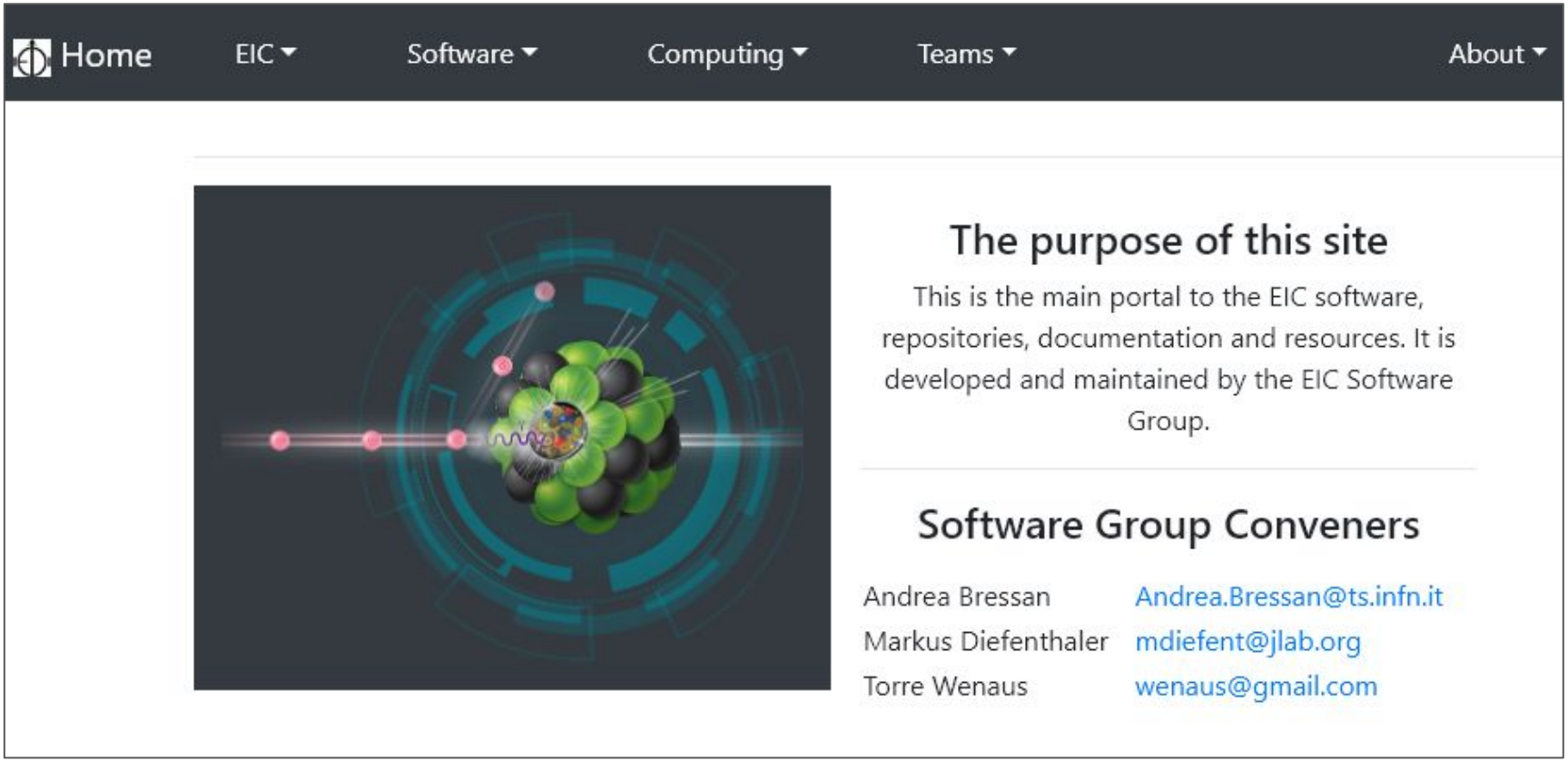
Overview

- Many thanks to Markus and others for getting the EIC Software GitHub organization in place and for the initial setup
- 28 repositories and 12 total participants as of April 22 2020
- It’s already being put to good use
  - Steady flow of updates
- The group is working on how to best keep organized in the GitHub environment e.g. leverage features like teams, projects, project boards etc
  - Comments, ideas and suggestions are welcome
- We started using the GitHub Pages platform to host the EIC Software Group Website: https://eic.github.io/

Migration

- Prior to the EIC Organization on GitHub, we had a mix of solutions which prominently included GitLab, which is also a good platform
- The idea was create a unified and future-proof resource for the EIC community to use
  - Avoid fragmentation of repos, documentation and tools (which are many)
  - Comments?
- Need a clear policy on migration to GitHub and notify the community
- The new website (next slide) is a good tool to guide the user through the repo, provide documentation etc

https://eic.github.io/



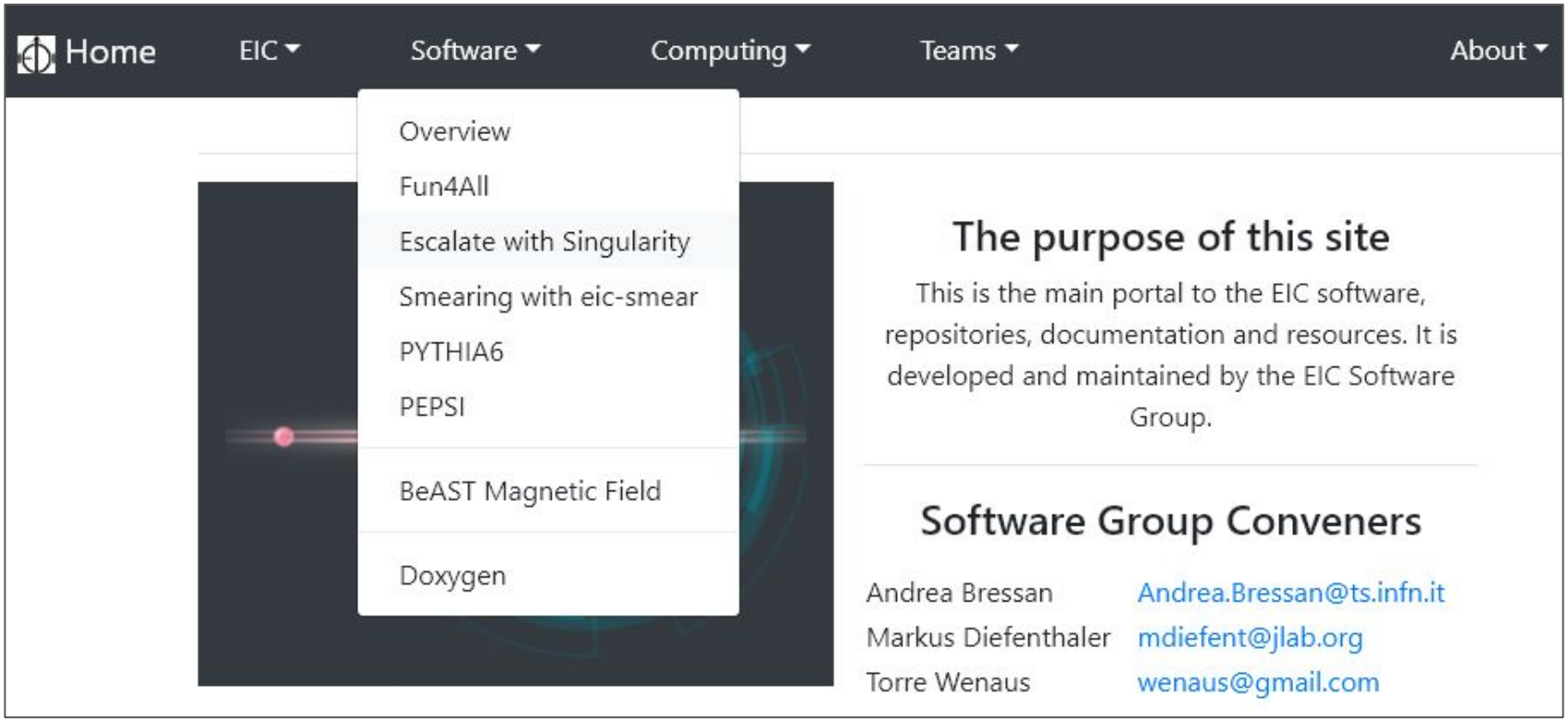
The EIC Software Website on GitHub

- Repo: https://github.com/eic/eic.github.io
- Inspiration for the toolkit and layout/navigation scheme came from the *HEP Software Foundation* website: https://hepsoftwarefoundation.org/
  - However the code of the EIC site is new and different from its predecessors with emphasis on ease of maintenance and facilitating users' contributions
  - Updated with the latest versions of third-party components
  - Purposely lean design
- Based on Jekyll/Liquid + Bootstrap for layouts and navigation
  - Users/contributors are **insulated** from intricacies of both unless they want to contribute to the mechanics of the site
  - Very basic familiarity with Markup is the only requirement for someone who wants to contribute

The Platform

- Advantages of static websites:
  - Security
  - Ease of maintenance (no dependencies on PHP etc)
  - Portability (down to a USB stick) and compatibility with data preservation
  - Performance (no DB queries)
  - Reliable version control
  - GitHub (and other sites) integration + free hosting
- Jekyll is a static website generator
- Key features
  - Structured storage of the content/data with consistent references across sites (YAML)
    - e.g. can help systematize simulations
  - DB-like features without using an actual database (i.e. do a “join” on data structures)
  - Solid web page templating functionality and filters, flow control, includes etc
  - No straight HTML - editing Markdown is a lot more palatable

Navigation (dropdowns)



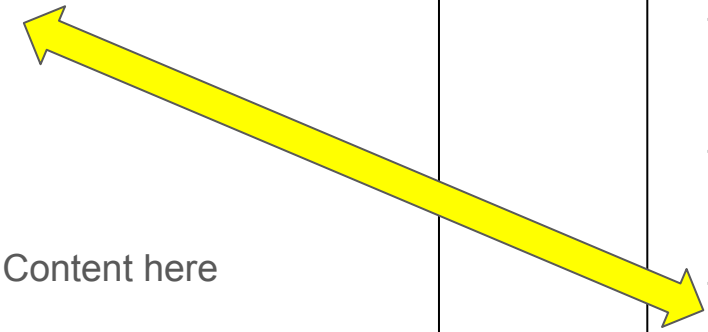
Contributing to the Website

- The navigation bar on top contains items named consistently with folders in the codebase, making it easier for contributors to get orientated:
  - *Software* → ”_software”
  - *About* → ”_about”etc
- To add a page one needs to do the following
  - Add a Markdown-formatted file (with the mandatory “front matter” section on top) to the appropriate folder e.g. if it’s a software-related page add it to the “_software”, if it’s about teamwork add it to “_teams” etc (see next slide)
  - Register your page in the file “_data/menu.yml” to position the link (created automatically) in the menu structure, with the “name” attribute matching one in the “front matter”
- Existing code is clear enough to provide working examples, no need to guess
- There is a “how-to” page with more information: https://eic.github.io/about/howto.html

Front Matter (actual example)

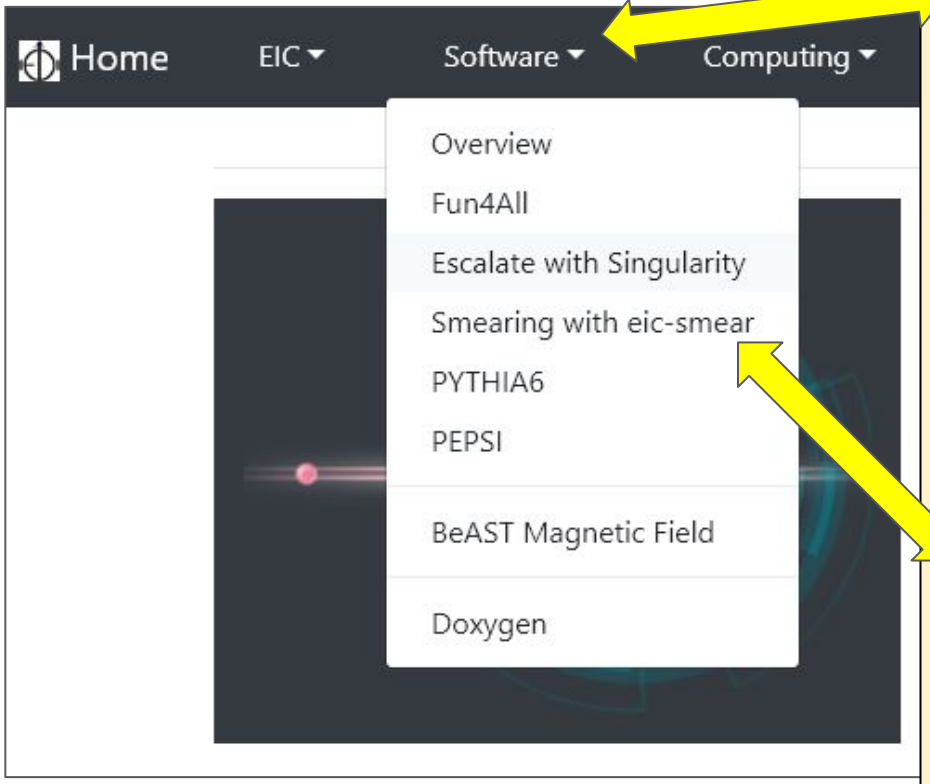
```
# Only two first attributes are mandatory
---
title: Smearing with eic-smear
name: eicsmear_1
category: eicsmear
layout: default
level: 0
---

### The EIC Smear Content here
```



```
# menu.yml (defines menu content)
- name: software
  full: Software
  submenus:
    -
      name: overview
      full: Overview
    -
      name: fun4all_tutorial_1
      full: Fun4All
    -
      name: escalate_singularity_1
      full: Escalate with Singularity
    -
      name: eicsmear_1
      full: Smearing with eic-smear
    -
      name: pythia6
      full: PYTHIA6
```

Menu rendering




```
- name: software
  full: Software
  submenus:
    -
      name: overview
      full: Overview
    -
      name: fun4all_tutorial_1
      full: Fun4All
    -
      name: escalate_singularity_1
      full: Escalate with Singularity
    -
      name: eicsmear_1
      full: Smearing with eic-smear
    -
      name: pythia6
      full: PYTHIA6
```

Contributing to the Website (cont'd)

- The site is automatically re-generated on GitHub on every "push"
  - Latency varies from seconds to minutes
- It is entirely possible to just edit a file and commit, or even do it right on the GitHub site in the Web UI - this is a plus
- However for optimal productivity it is recommended to install Jekyll (and before that a recent version of Ruby) on your machine - not too hard but will take a few minutes
- Once this is done, you get a development server on your machine on port 4000 which allows you to check all changes immediately
- Initial feedback from contributors is positive i.e. the system is fairly easy to use (~5 contributors active)

https://github.com/eic/eic.github.io

_about	modified "how to" to reflect recent changes in organization	18 hours ago
_computing	modified "how to" to reflect recent changes in organization	18 hours ago
_data	Added pythia-rad-corr	9 hours ago
_includes	Cleaned up retired MD files	19 hours ago
_layouts	created a basic cross-reference structure for sofware pages	15 days ago
_software	updated eicsmear	7 hours ago
_teams	converted the navbar center lines into a loop interating over YAML	19 hours ago
assets/images/site	Correct BeAST field map picture uploaded	8 days ago
.gitignore	Updated the "ignore"	18 days ago
Gemfile	Added all of the stubs (for now)	18 days ago
LICENSE	Initial commit	21 days ago
README.md	Update README.md	8 hours ago
_config.yml	Migrating more material to the new drop menu generation scheme	19 hours ago
index.html	Added all of the stubs (for now)	18 days ago

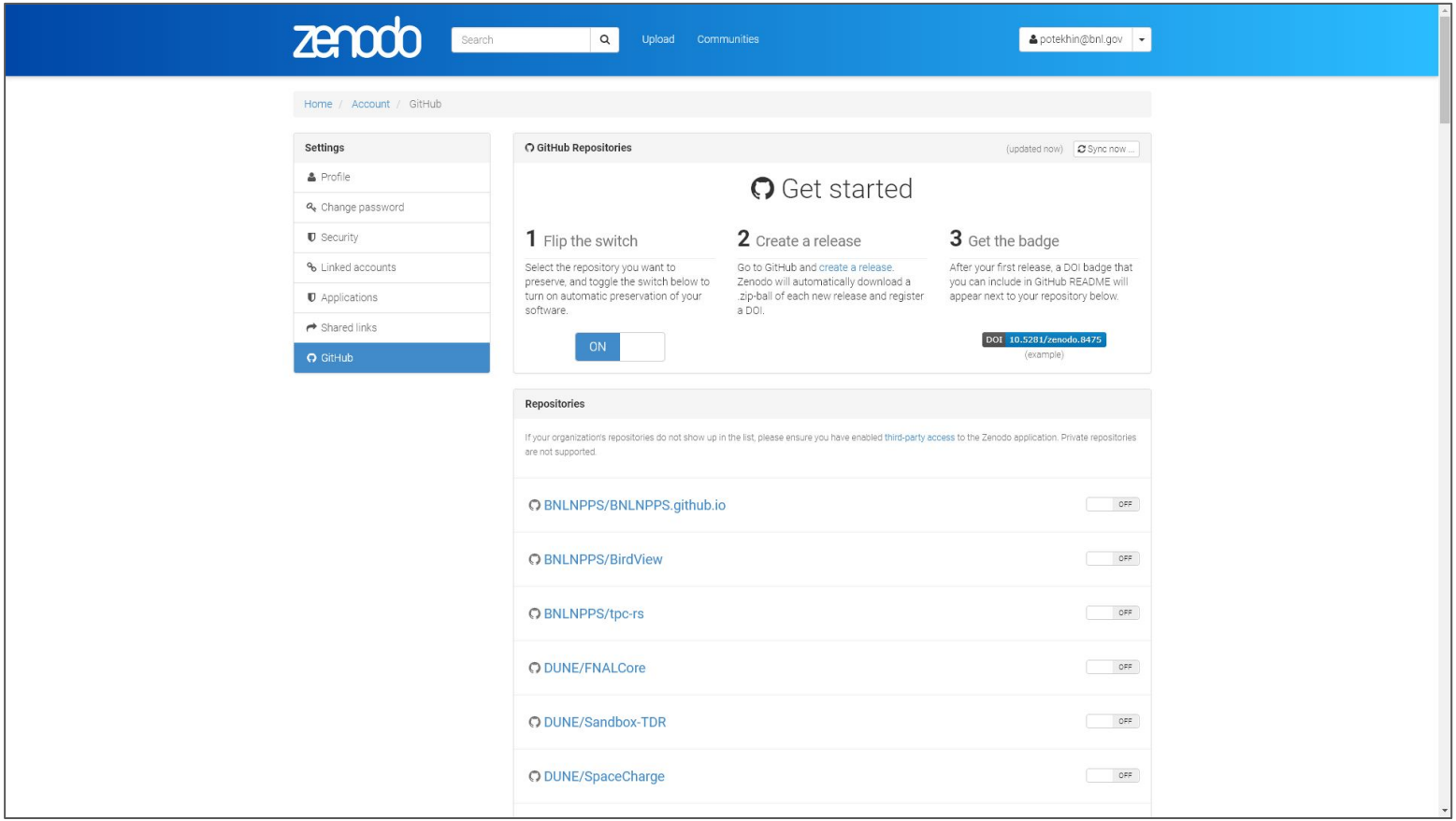
GitHub integration with Zenodo

- Zenodo is a CERN-based digital repository: https://zenodo.org/
  - It is not a new product but in fact an evolution of CDS i.e. one of the cornerstones of the CERN infrastructure and a myriad information services
  - “Invenio RDM” is in the works which is a portable version of this CERN-based system
  - Vibrant community, solid CERN affiliation, good support at many levels including the lead developer
- It can serve as a drop-in replacement for DocDB which is an aging product
  - Metadata support
  - But can also do so much more e.g. store datasets, code or any other digital products
  - Generates and supports official DOIs
- It definitely **deserves a separate discussion** about its potential use for EIC
  - The current need to manage documents in EICUG is not addressed properly
- Today’s focus is on one aspect of it: the GitHub integration

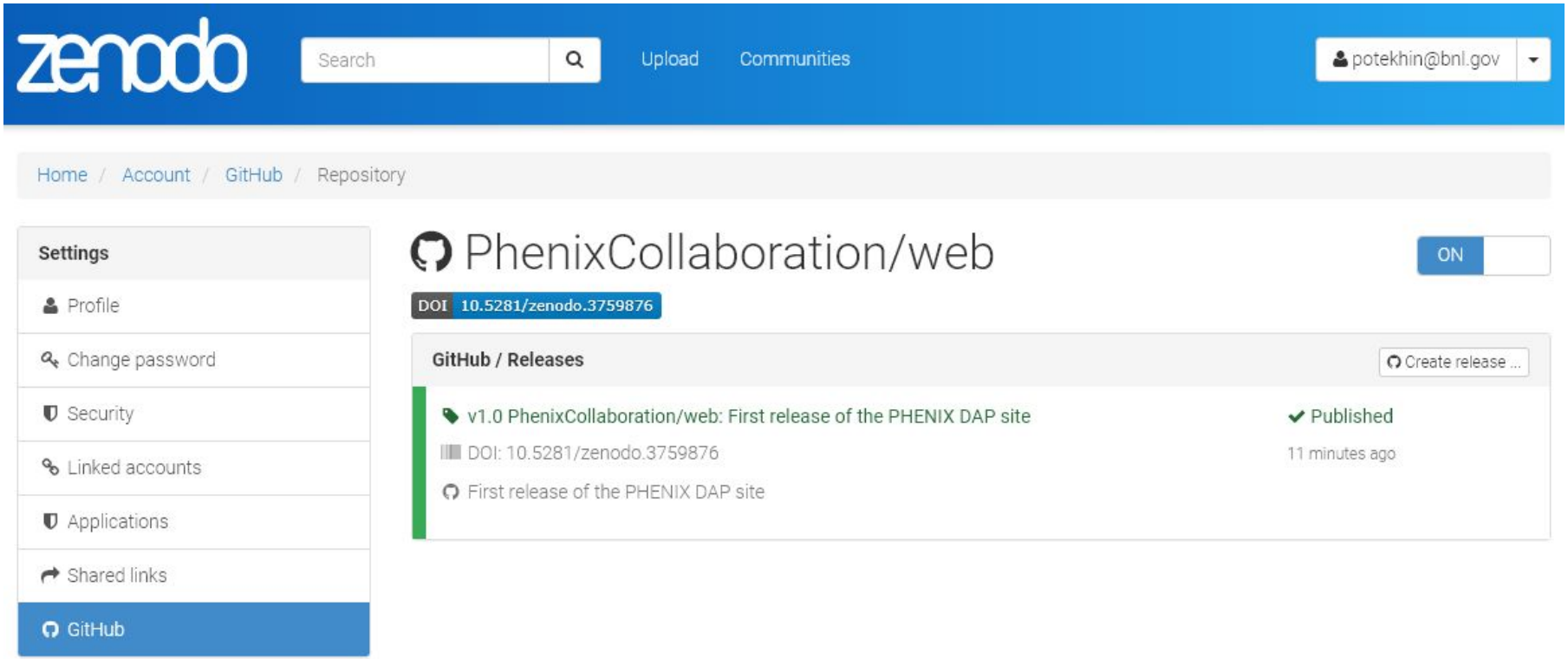
GitHub/Zenodo mechanics

- A snapshot of a GitHub repository can be included in Zenodo organically, and a DOI generated
  - Prepares and preserves tarballs of your releases
  - Makes your code easy to find (using the metadata) and to reference by a unique ID
  - Nice GUI
  - Potential utility for the Yellow Report and beyond
- Easy to use
  - I tested this functionality and it was quite simple
  - DOIs take some time O(10min) to propagate to the DOI.org system

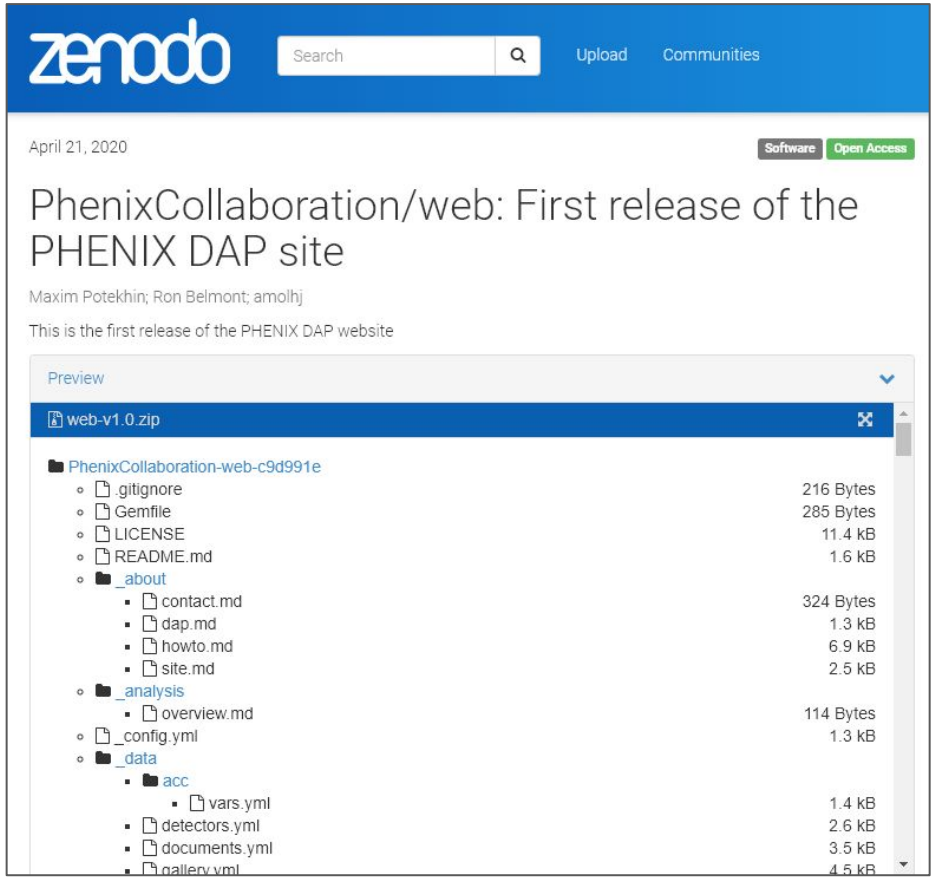
Zenodo - GitHub panel - repo selection



Zenodo - GitHub panel - published release



Zenodo - GitHub panel - published release browser



GitHub/Zenodo integration benefits

- Not a core functionality by a long shot, however...
- ...provides a uniform way to reference digital products using DOI
- ...metadata is a good thing to have - better discoverability!
- ...can leverage the Zenodo “community” feature to organize materials and increase visibility
  - Cf. simulated data and the code used to produce it can be kept under the same umbrella
- Longer term - Data and Analysis Preservation