Experience with website design optimized for long-term maintenance scenarios

M.Potekhin

BNL Nuclear and Particle Physics Software Group (NPPS)

The DAP Round Table

05/08/2025

An observation

- Most experiments need to have robust web presence for an extended period of time, to remain productive.
- Platforms that can be efficient for near term goals may prove difficult to maintain in the long term.
- In the following, we’ll take a look at the traditional content management systems and at the “static website generators”.

Content management systems (CMS)

- CMS’ rely on a backend DB server to store content and to (dynamically) render it into web pages, at run time. The front-end relies on executable code (e.g. PHP). Hence, potential security issues.
- Examples: Wordpress, MediaWiki, Drupal.
- Ability to store ad-hoc structured data is typically limited since databases are design for generic content storage, and requires additional or custom created plug-in software.
- Inclusion of a writable database creates security concerns and necessitates a long-term commitment for support and upgrades of the respective RDBMS.

Another look at Wiki and Drupal

- There is plenty of prior experience with Mediawiki (Wiki) and most people agree these resources don’t age well because of lack of management tools and only rudimentary access and version control.
- There are good content management systems there but they share some of the problems with the Wiki and in some cases (e.g. Drupal) long-term maintenance incurs a real cost in effort required due to the never ending software update cycle, which starts with mandatory PHP updates (either at 3rd party hosts or BNL). This is exacerbated if customized code was added to the CMS.
- Migration to a different server is far from trivial (requires a DB replication procedure of some sort).
- We need a portable system which requires a minimal effort to maintain.

Static websites vs content management systems

- By contrast with CMS, the static websites are compiled once and are deployed by copying the resulting HTML and other materials to the target server.
- There are multiple platforms to support this approach, such as Jekyll, Hugo, Gatsby and many others. This technology is mainstream.
- Similar to CMS, the goal is to simplify managing the content, although most often it is done not in the WYSIWYG manner, but rather be managing simple-to-read text files and links to uploads. And also –…
- Taken the well established Jekyll site builder as an example: structured data can be easily integrated into the website and rendered as needed, with an equivalent of a "join" operation, at compile time. More on that below.

The web technology choice

- HSF, NPPS, PHENIX, EICUG and now ePIC are all using a particular static website generator called “Jekyll”, an app written in Ruby (see next slide) which includes the Liquid templating language.
- A set of inputs (text, images, layouts, data) is converted into a collection of HTML files which form a complete website. Deployment then is effectively a copy of the compiled HTML collection to the target server. The result is high performance, security and ease of deployment.
- The Markdown format used for creation and management of the text content on the site is not difficult to learn, from our experience.
- The data content (e.g. working groups and conference info) is kept in YAML files, which are parsed as needed to render the content on the web pages, approximating the DB functionality.

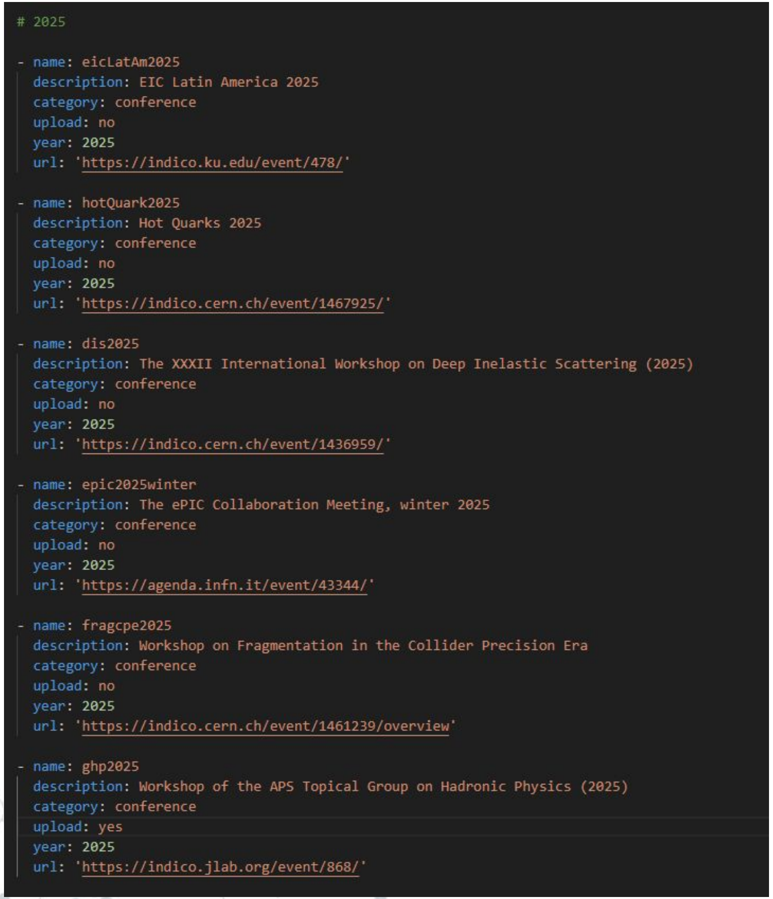
Parsing data

- Borrowing from industry — the “Liquid” templating/macro language was developed by the e-commerce company “Shopify”.
- This platform has been in continuous use at scale since 2006.
- Well documented and well understood.
- See https://shopify.dev/docs/api/liquid
- Powerful parsing and filtering features included. The code can be included in-line into page written in Markdown, or stored as macros.
- Macros can be chained and nested, and included as Markdown files into other Markdown files.

Structured data

- YAML has already been mentioned as the container for structured data, and CSV can be used as well if needed. These are simple, proven formats.
- It’s a good idea to identify components of web pages that can be factored into (a) a structured data part, and (b) the corresponding presentation layer. A natural example of this is tabulated data.
- An additional bonus is that the same data can be rendered on different pages according to the context. This ensures referential integrity i.e. you only edit and maintain the data in one location instead of tweaking multiple pages at once.

Example: the ePIC conference section in the keywords file



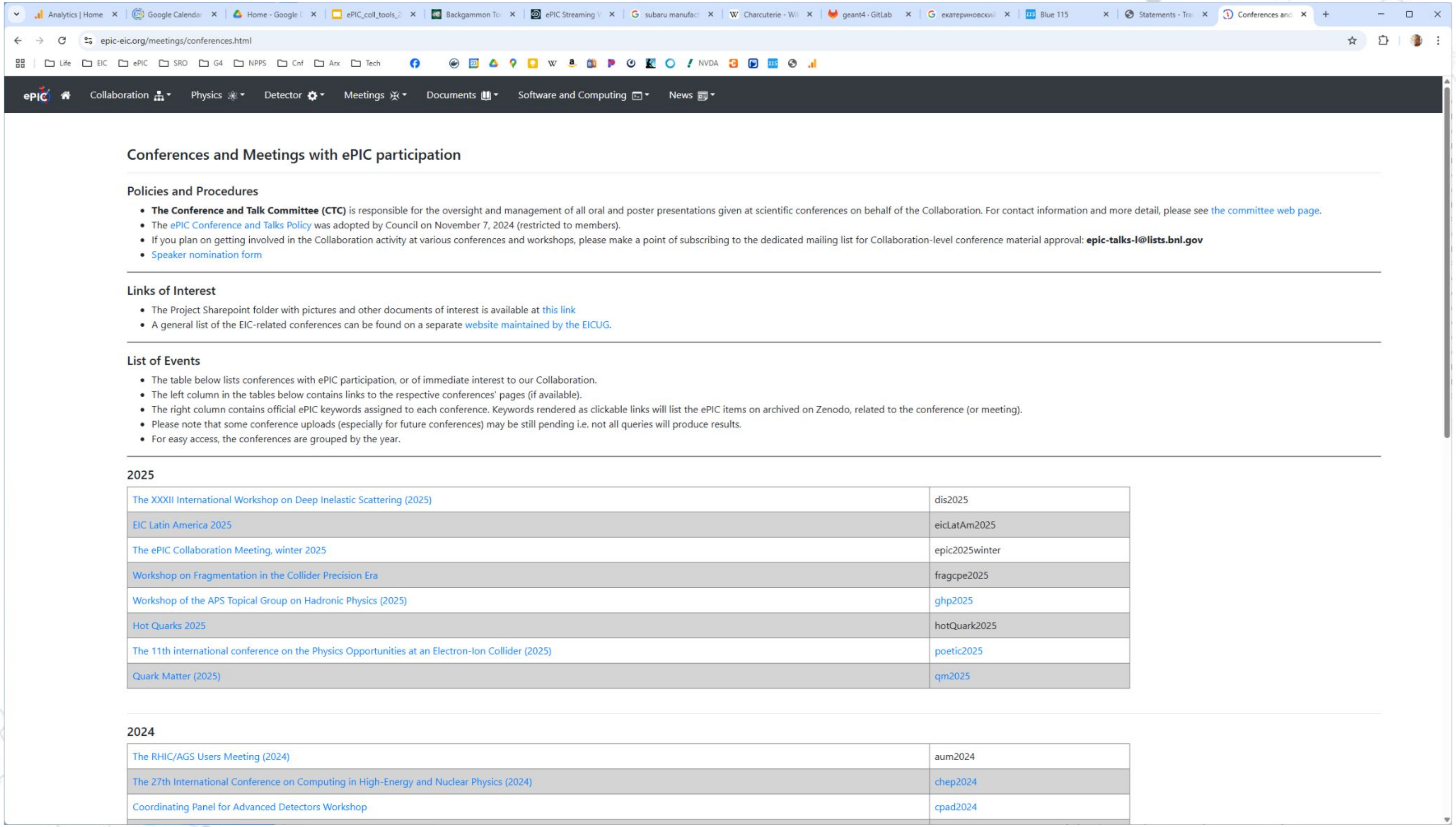
```
# 2025

- name: eicLatAm2025
  description: EIC Latin America 2025
  category: conference
  upload: no
  year: 2025
  url: 'https://indico.ku.edu/event/478/'

- name: hotQuark2025
  description: Hot Quarks 2025
  category: conference
  upload: no
  year: 2025
  url: 'https://indico.cern.ch/event/1467925/'

- name: dis2025
  description: The XXXII International Workshop on Deep Inelastic Scattering (2025)
  category: conference
  upload: no
  year: 2025
  url: 'https://indico.cern.ch/event/1436959/'

- name: epic2025winter
  description: The ePIC Collaboration Meeting, winter 2025
  category: conference
  upload: no
  year: 2025
  url: 'https://agenda.infn.it/event/43344/'

- name: fragcpe2025
  description: Workshop on Fragmentation in the Collider Precision Era
  category: conference
  upload: no
  year: 2025
  url: 'https://indico.cern.ch/event/1461239/overview'

- name: ghp2025
  description: Workshop of the APS Topical Group on Hadronic Physics (2025)
  category: conference
  upload: yes
  year: 2025
  url: 'https://indico.jlab.org/event/868/'
```

- These data are well organized, easy to read and can be rendered as needed on any page.
- Less work and maintenance than with the Wiki i.e. there are no concerns about the format of the presentation – here only the data need to be modified.
- There are automatic URLs included in tables based on the data, and automatic Zenodo links, with additional control via the “upload” attribute (e.g. to prevent unsuccessful queries in cases when materials are still pending the upload).
- Added an optional “nominations” functionality, currently not commissioned by the CTC decision.
- The updates page has been moved from the “Documents” section to “Meetings”, where is more organically belongs.

The ePIC website code management and features

- The ePIC website: https://www.epic-eic.org
- We leverage the useful features on GitHub, which works well for team effort. Recently there is an added a preview feature for the proposed changes.
- A managed list of keywords is used to achieve tight integration with our Zenodo repository e.g. automated searches. This is done using “Liquid” macros, which generate links automatically based on DOIs. Same approach can be used with Invenio RDM.
- Example: the WG info is stored in YAML, and parsed into a standard group template with a provision to add any custom content as needed.

The Updated ePIC Conferences Page



Conferences and Meetings with ePIC participation

Policies and Procedures

- **The Conference and Talk Committee (CTC)** is responsible for the oversight and management of all oral and poster presentations given at scientific conferences on behalf of the Collaboration. For contact information and more detail, please see the committee web page.
- The ePIC Conference and Talks Policy was adopted by Council on November 7, 2024 (restricted to members).
- If you plan on getting involved in the Collaboration activity at various conferences and workshops, please make a point of subscribing to the dedicated mailing list for Collaboration-level conference material approval: **epic-talks-l@lists.bnl.gov**
- Speaker nomination form

Links of Interest

- The Project Sharepoint folder with pictures and other documents of interest is available at this link
- A general list of the EIC-related conferences can be found on a separate website maintained by the EICUG.

List of Events

- The table below lists conferences with ePIC participation, or of immediate interest to our Collaboration.
- The left column in the tables below contains links to the respective conferences' pages (if available).
- The right column contains official ePIC keywords assigned to each conference. Keywords rendered as clickable links will list the ePIC items on archived on Zenodo, related to the conference (or meeting).
- Please note that some conference uploads (especially for future conferences) may be still pending i.e. not all queries will produce results.
- For easy access, the conferences are grouped by the year.

2025

The XXXII International Workshop on Deep Inelastic Scattering (2025)	dis2025
EIC Latin America 2025	eicLatAm2025
The ePIC Collaboration Meeting, winter 2025	epic2025winter
Workshop on Fragmentation in the Collider Precision Era	fragcpe2025
Workshop of the APS Topical Group on Hadronic Physics (2025)	ghp2025
Hot Quarks 2025	hotQuark2025
The 11th international conference on the Physics Opportunities at an Electron-Ion Collider (2025)	poetic2025
Quark Matter (2025)	qm2025

2024

The RHIC/AGS Users Meeting (2024)	aum2024
The 27th International Conference on Computing in High-Energy and Nuclear Physics (2024)	chep2024
Coordinating Panel for Advanced Detectors Workshop	cpad2024

The Website: content vs layout

- The content and the layout of the site, which defines the look and feel of the site, are not related.
- The former consists of a collection of Markdown, YAML and image files.
- The layout is defined in a collection of “templates”. The site can be given a new “skin” w/o changes in the content. More flexibility than in CMS.
- A template includes a HTML, CSS and Javascript components. JS is needed to provide more interactivity to the site, e.g. drop-down menus and other embellishments.
- There is a choice of Javascript libraries, the criteria for selection include stability, security, outlook for support. We chose Bootstrap since it matches well with these requirements.

Maintenance

- Once a website is set up, our experience indicates that the maintenance requires very little effort, and editing of materials can be done after a brief learning curve. So far it seems a good match for DAP purposes.
- Recently, there are GitHubs plugins like “Netlify” which allow a preview based on pull requests.
- …and, installing Jekyll on the user’s machine is usually not too hard, so the preview can happen locally as well.
- All of this is important because a syntax error can indeed interfere with an entire page, so it’s a good idea to verify edits before pushing to main.

This technology applied in the DAP context

- Some experiments adopted the static website generator technology early – like in ePIC.
- Others had to migrate later, e.g. PHENIX and EICUG.
- Easier compliance with Cybersecurity compared to CMS or legacy PHP-based sites — as experienced at BNL.
- Structured data provides many benefits of the database, without having one. It also works well for preserving information in a well organized and structured way.