FERMILAB-CONF-20-642-CCD

EPJ Web of Conferences **245**, 08028 (2020) https://doi.org/10.1051/epjconf/202024508028
CHEP 2019

Development of a Versatile, Full-featured Search Functionality for Indico

Penelope Constanta[1,*], *Ofer* Rind[2], *Jose* Caballero Bejar[2], *Pedro* Ferreira[3], *Adrian* Mönnich[3], *Pablo* Panero[4], *Carina Rafaela* De Oliveira Antunes[5], and *Aristofanis* Chionis Koufakos[3]

[1]Fermi National Accelerator Laboratory, CCD, P.O. BOX 500, Batavia IL 60510-5011, USA
[2]Brookhaven National Laboratory, Physics Dept., P.O. Box 5000, Upton, NY 11973-5000, USA
[3]CERN, Esplanade des Particules 1, IT-CDA-IC, P.O. Box 1211 Geneva 23, Switzerland
[4]CERN, Esplanade des Particules 1, IT-CDA-DR, P.O. Box 1211 Geneva 23, Switzerland
[5]CERN, Esplanade des Particules 1, IT-CDA-WF, P.O. Box 1211 Geneva 23, Switzerland

Abstract. Indico, CERN's popular open-source tool for event management, is in widespread use among facilities that make up the HEP community. It is extensible through a robust plugin architecture that provides features such as search and video conferencing integration. In 2018, Indico version 2 was released with many notable improvements, but without a full-featured search functionality that could be implemented easily outside of CERN. At both Fermi and Brookhaven National Laboratories, the user community viewed the lack of this popular feature as a significant impediment to deployment of the new software. In the meantime, CERN embarked upon a major redesign of their core search service, one that would also necessitate a rewrite of the Indico search interface. Seeing this pressing need, the two US labs decided to collaborate, with assistance from the CERN development team, on a project to develop the requisite search functionality for the larger user community. The resulting design exploits the simplified schema defined in the new CERN Search micro-service, based on Invenio and Elasticsearch, while still providing a flexible path to implementation for alternative backend search services. It is intended to provide a software package that can be installed easily and used out of the box, by anyone at any site. This paper will discuss the design choices and architectural challenges, and provide an overview of the deployment and use of these new plugins.

1 Overview

Indico is an open-source event management system, popular in the High Energy Physics (HEP) community. Indico's extensibility through its plugin architecture provides integrated event fee payment (via PayPal), video conferencing, search, room booking etc. that can be modified and/or extended to suit the specific needs of a particular institute. [1][2]

The latest version of Indico v2.x [4][5] has been re-written to use modern Python packages (Flask, SQLAlchemy, marshmallow etc.) and to use PostgreSQL for its database

[*] Corresponding author: penelope@fnal.gov

needs. However, Indico's popular search plugins have not been upgraded and the search functionality is limited to CERN's eco-system that currently uses SharePoint. The necessity for developing new plugins that are not tightly coupled with CERN's eco-system, is twofold: CERN is moving away from SharePoint to the new Invenio based CERN Search microservice; and the Fermi, BNL user communities requested a full functional search before deploying the new version 2.x of Indico.

The Fermi-BNL-CERN collaboration was formed to build the search plugins utilizing the new CERN Search microservice and make them available to the community as an easily deployable solution.

1.1 Indico's search plugins history

Indico v0.98 – v1.2: the search plugins utilize Invenio (v1.1) as their search engine. The plugins send search metadata and receive search results using the MARCXML Standards. The framework (Indico and Invenio) are not specific to CERN's infrastructure and can be used outside CERN's environment.

Indico v1.9 – v2.2: CERN itself moved towards a solution which uses an internally-available custom solution based on SharePoint, while the officially available Invenio plugin fell into disuse for lack of a maintainer. The CERN solution plugins send search metadata using the MARCXML Standards and search results are displayed by SharePoint.

Future Indico v2.x: the search plugins, as developed in this collaboration effort, will utilize the Invenio's (v3) "CERN Search API" component, built upon Elasticsearch as its search engine. The plugins will send search metadata and receive search results in JSON format. The framework is being developed so that it can be used both internally by CERN as well as outside CERN's environment.

2 Indico's code architecture

Indico's application code consists of the core system that provides the main functionalities of the application and the plugin subsystem which extends the application's functionality. The plugins are standalone packages of code that do not require any modifications to the Indico core itself. A plugin can perform something very simple such as adding a new command to the Indico CLI or more complex functionalities like introducing new payment methods, chat integration, etc. [3]

Indico's search functionalities are implemented by a set of plugins. The *livesync* and *search* plugins are core Indico plugins that require the implementation of corresponding *agent* plugins to provide connection with a particular search engine.

Indico's search metadata are collected by the core *livesync* plugin. Livesync collects metadata from events and their contributions, subcontributions, attachments and notes. The collected metadata are sent to the *livesync agent* plugin. The livesync agent is responsible for formatting appropriately and transmitting the search metadata to its associated search engine.

Indico's user search interface is implemented by the core *search* plugin. The search plugin accepts the user's search criteria, sends them to a *search agent* plugin and then receives and displays the search results. The search agent plugin is responsible for formatting and transmitting the search criteria as well as receiving the search results and sending them to the core search plugin. The Indico search architecture and the Fermi-BNL-CERN effort architecture are illustrated in Figures 1 and 2.

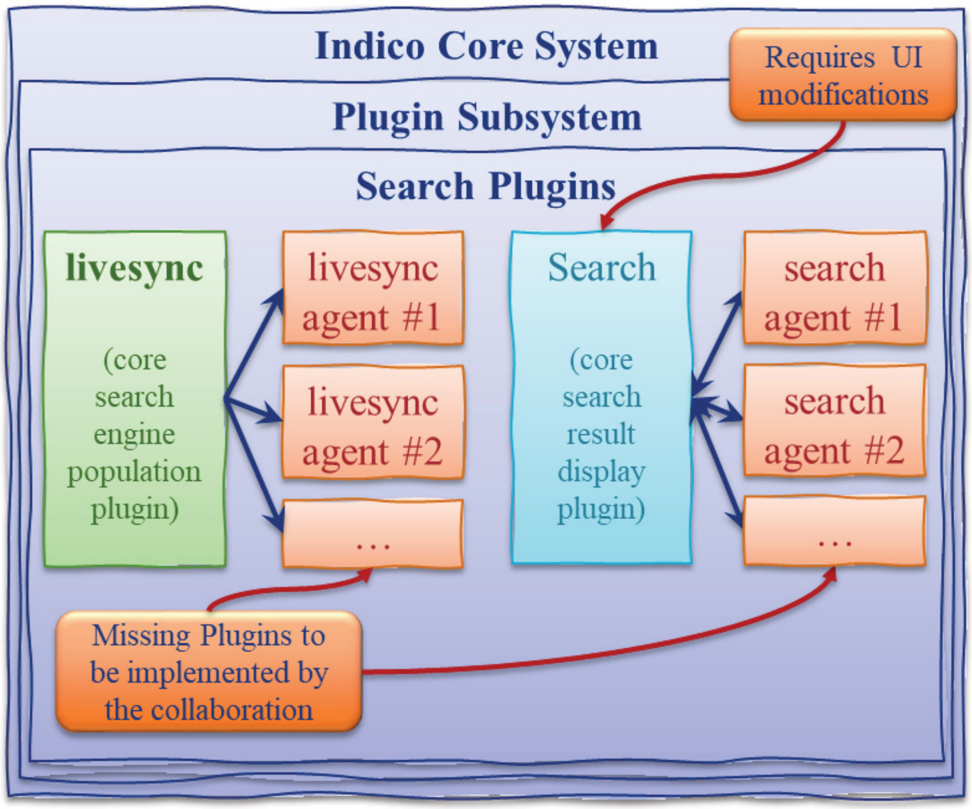

Indico Core System
Requires UI modifications
Plugin Subsystem
Search Plugins
livesync
(core search engine population plugin)
livesync agent #1
livesync agent #2
...
Search
(core search result display plugin)
search agent #1
search agent #2
...
Missing Plugins to be implemented by the collaboration

Figure 1. Indico code architecture

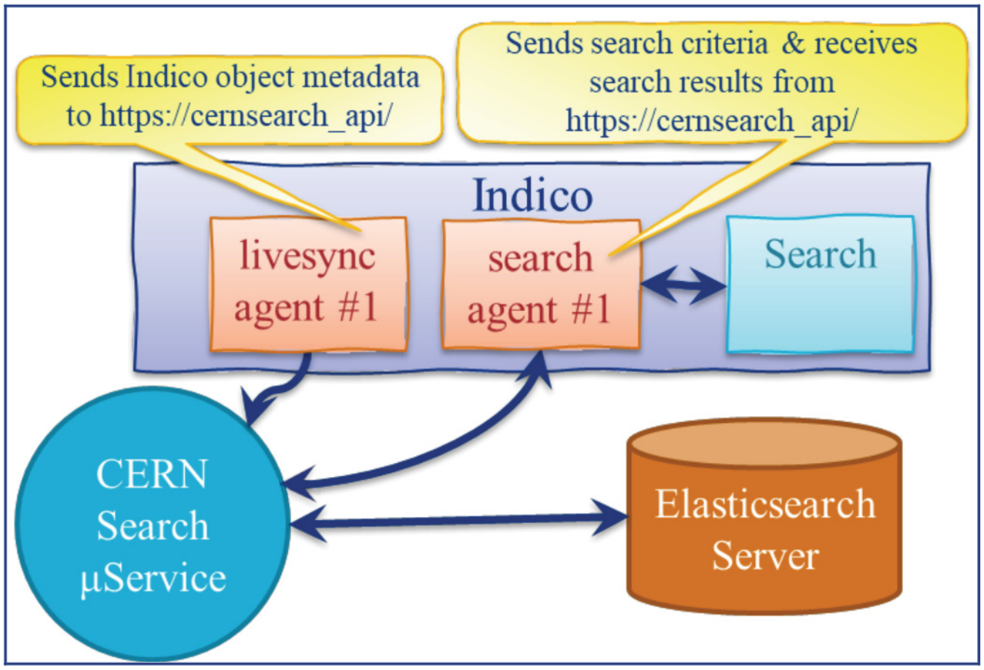

Sends Indico object metadata to https://cernsearch_api/
Sends search criteria & receives search results from https://cernsearch_api/
Indico
livesync agent #1
search agent #1
Search
CERN Search μService
Elasticsearch Server

Figure 2. Indico proposed search system architecture

3 Implementation challenges

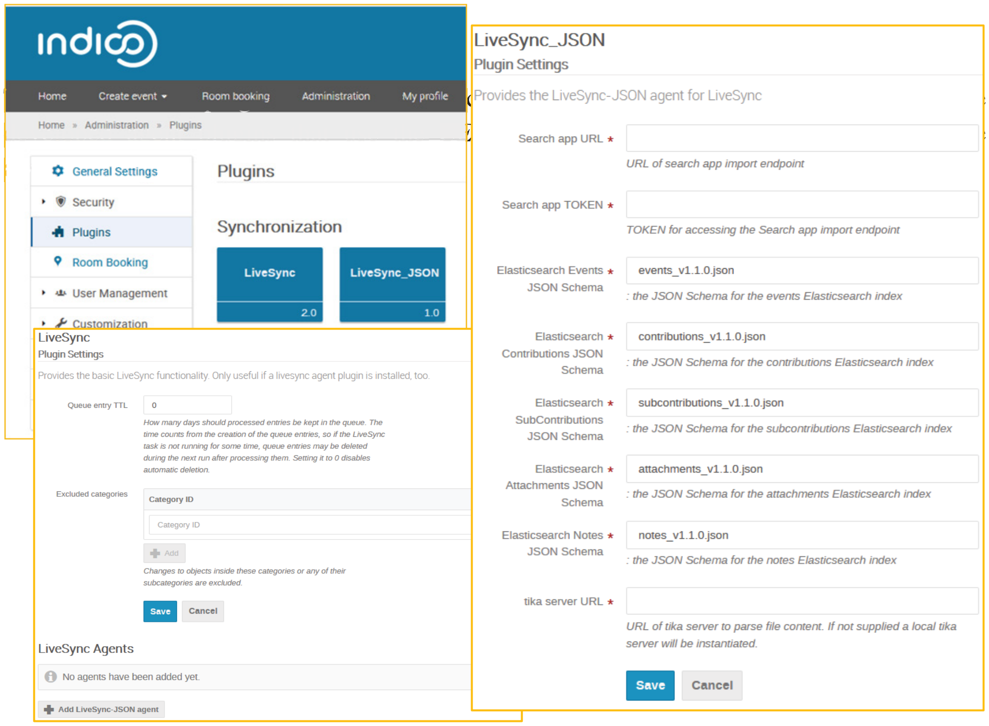
Indico v2.x moved away from the ZODB to PostgreSQL and almost the entire Indico code was re-written and restructured. For an outside developer, any familiarity with the previous versions' code is not useful. The new version provides an easier plugin development method, but the trade-off is that one now needs to understand all the internals of the new Indico plugin system, as well as the interface with the core plugins and the core code, along with numerous new Python packages.

The agreed upon development of the search plugin agents requires interfacing with the CERN Search microservice which is very new, and documentation has thus far been targeted for CERN's internal use. The deployment of the microservice through docker-compose also proved to be more challenging than expected. The CERN provided `docker-compose.yml` that deploys the necessary docker containers for the microservice, worked as expected, but the configuration to populate the Elasticsearch mappings was not that obvious.

Apart from the coding difficulties, the FNAL and BNL developers worked for a fraction of their time on the Indico project and were not familiar with the new Python packages.

4 Implementation status

The existing core *livesync* plugin will be used as is and will be communicating with the new livesync agent, *LiveSync_Json*, that was built by the collaboration. A new core *search* plugin, that includes the user interface, has been implemented by CERN and is in place.



LiveSync
Plugin Settings
Provides the basic LiveSync functionality. Only useful if a livesync agent plugin is installed, too.

Queue entry TTL: 0
How many days should processed entries be kept in the queue. The time counts from the creation of the queue entries, so if the LiveSync task is not running for some time, queue entries may be deleted during the next run after processing them. Setting it to 0 disables automatic deletion.

Excluded categories: Category ID
Changes to objects inside these categories or any of their subcategories are excluded.

LiveSync Agents
No agents have been added yet.

LiveSync_JSON
Plugin Settings
Provides the LiveSync-JSON agent for LiveSync

Search app URL *
URL of search app import endpoint

Search app TOKEN *
TOKEN for accessing the Search app import endpoint

Elasticsearch Events JSON Schema *: events_v1.1.0.json
: the JSON Schema for the events Elasticsearch index

Elasticsearch Contributions JSON Schema *: contributions_v1.1.0.json
: the JSON Schema for the contributions Elasticsearch index

Elasticsearch SubContributions JSON Schema *: subcontributions_v1.1.0.json
: the JSON Schema for the subcontributions Elasticsearch index

Elasticsearch Attachments JSON Schema *: attachments_v1.1.0.json
: the JSON Schema for the attachments Elasticsearch index

Elasticsearch Notes JSON Schema *: notes_v1.1.0.json
: the JSON Schema for the notes Elasticsearch index

tika server URL *
URL of tika server to parse file content. If not supplied a local tika server will be instantiated.

Figure 3. Livesync plugins setup pages

4.1 Indico livesync agent plugin configuration

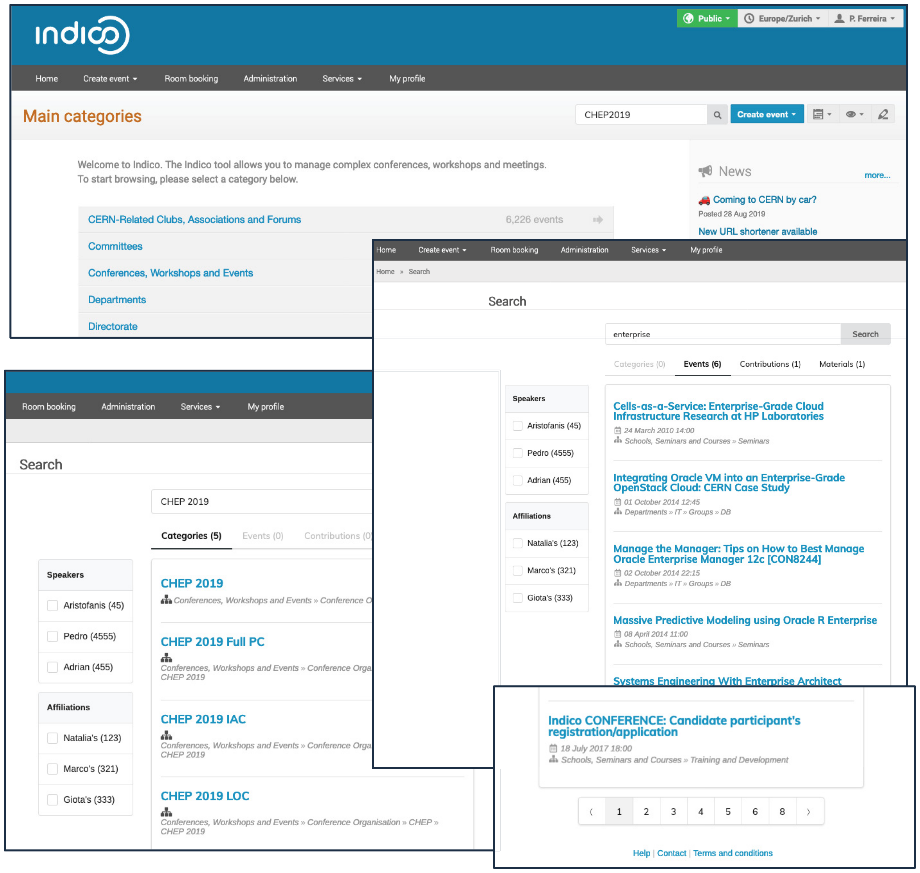
Figure 3 contains screenshots of the *livesync* and *LiveSync_Json* configuration.

The plugins are listed under the Plugins administration page and each is configured separately. The *livesync* plugin configuration is simple and not significantly changed from previous versions of Indico. In contrast, the *LiveSync_Json* configuration is now much more elaborate. It requires not only the URL and authorization token needed to communicate with the *CERN Search microservice*, but also the Elasticsearch schema file names, used in the initialization of the microservice for building the json structures needed by the search engine.

4.2 Indico search user interface

Figure 4 contains screenshots of Indico's new search user interface as implemented by CERN for Indico v2.x.

The search results are paginated using the plugin's configuration of search results number per page and categorized by Categories, Events, Contributions, Sub-Contributions, Notes, Attachments. The search results can be further filtered by Speaker Name and/or Affiliation.



Welcome to Indico. The Indico tool allows you to manage complex conferences, workshops and meetings.
To start browsing, please select a category below.

CERN-Related Clubs, Associations and Forums 6,226 events
Committees
Conferences, Workshops and Events
Departments
Directorate

Cells-as-a-Service: Enterprise-Grade Cloud Infrastructure Research at HP Laboratories
24 March 2010 14:00
Schools, Seminars and Courses » Seminars

Integrating Oracle VM into an Enterprise-Grade OpenStack Cloud: CERN Case Study
01 October 2014 12:45
Departments » IT » Groups » DB

Manage the Manager: Tips on How to Best Manage Oracle Enterprise Manager 12c [CON8244]
02 October 2014 22:15
Departments » IT » Groups » DB

Massive Predictive Modeling using Oracle R Enterprise
08 April 2014 11:00
Schools, Seminars and Courses » Seminars

Systems Engineering With Enterprise Architect

Indico CONFERENCE: Candidate participant's registration/application
18 July 2017 18:00
Schools, Seminars and Courses » Training and Development

CHEP 2019
Conferences, Workshops and Events » Conference O

CHEP 2019 Full PC
Conferences, Workshops and Events » Conference Orga
CHEP 2019

CHEP 2019 IAC
Conferences, Workshops and Events » Conference Orga
CHEP 2019

CHEP 2019 LOC
Conferences, Workshops and Events » Conference Organisation » CHEP »
CHEP 2019

Figure 4. Indico's search UI and search results display

5 Future Development

All the agent plugins developed by the collaboration will be integrated into Indico and CERN will take ownership. These plugins will be available to the Indico community to use along with the CERN Search microservice.
Further development may include improved resilience and recovery for the livesync agent; extensions to the search UI, if needed; and last, but not least, improved developer documentation and deployment instructions for non-CERN environments.

References

1. https://getindico.io/
2. https://docs.getindico.io/en/latest/
3. https://docs.getindico.io/en/latest/plugins/
4. https://iopscience.iop.org/article/10.1088/1742-6596/664/5/052012
5. https://iopscience.iop.org/article/10.1088/1742-6596/898/10/102017