

Indico: A Collaboration Hub

P Ferreira, T Baron, C Bossy, J B Gonzalez, M Pugh, A Resco, J Trzaskoma, C Wachter

513-1-005, CERN CH-1211, Genève 23, Switzerland

E-mail: indico-team@cern.ch

Abstract. Since 2009, the development of Indico has focused on usability, performance and new features, especially the ones related to meeting collaboration. Usability studies have resulted in the biggest change Indico has experienced up to now, a new web layout that makes user experience better. Performance improvements were also a key goal since 2010; the main features of Indico have been optimized remarkably. Along with usability and performance, new features have been added to Indico such as webchat integration, video services bookings, webcast and recording requests, designed to really reinforce Indico position as the main hub for all CERN collaboration services, and many others which aim is to complete the conference lifecycle management. Indico development is also moving towards a broader collaboration where other institutes, hosting their own Indico instance, can contribute to the project in order make it a better and more complete tool.

1. Introduction

The Indico (Integrated Digital Conferencing) Software was born of a European Project, a joint initiative of CERN, SISSA, University of Udine, TNO, and Univ. of Amsterdam. The main objective was to create a web-based, multi-platform conference storage and management system. This software would allow the storage of documents and metadata related to real events. The project started in May 2002, and ended 2 years later. After the end of the European Project, CERN took over the project's development, based on the core modules that had been developed by the organization. Indico started as an Open Source project and so it has remained to this day - the entirety of its source code is available online, through the project website¹.

Indico is an event management web application. Its initial goal was to provide conference organizers with a set of tools that could help them through the entire conference



Figure 1. Indico servers around the world

¹ <http://www.indico-software.org>

life cycle. This initial feature set was extended to other events (such as meetings and lectures) after the end of the European project phase, and has since grown to include other features such as a full-fledged Room Booking module. Over the last 3 years, Indico became CERN's official hub for collaborative tools, providing a common user interface for videoconferencing, chat and webcasting/recording systems. As the project grew, other organizations in the High Energy Physics realm started using it, effectively creating a network of more than 100 different servers, distributed across 4 continents. The Indico Community keeps growing day by day.

At the time of the writing of this article, Indico's latest release is 0.98.2.

Indico at CERN

At CERN, Indico is the standard tool for event management. Virtually every event that takes place at the organization (regular meetings, lectures, workshops and conferences) is scheduled through Indico. Over the course of the last 5 years the number of events that are stored in CERN's Indico instance has grown dramatically. This accentuated growth called for a faster and more scalable Indico than the one that was then available.

2. From 0.97 to 0.98

Version 0.97 saw the complete redesign of most of the application interface and the adoption of a new service architecture that could provide the necessary back end to facilitate such a change[1]. By that time, Indico was still encumbered by several weaknesses:

- (i) An aging job scheduler ("task daemon") that was unstable and hard to extend for other uses
- (ii) Indico had a rudimentary script (`export.py`) aimed at providing metadata to the outside world under several structured formats (XML, iCalendar, RSS). As users felt the need to start extracting more metadata from Indico, the usage of this script grew in an uncontrolled way, causing occasional performance problems
- (iii) An aging and unstable OAI gateway that caused sporadic performance issues.

In order to solve these issues, the decision was taken to address them through the development of new modules from scratch, that would provide the same functionality in slightly different ways, but in a controllable and scalable fashion:

- (i) A brand-new extensible job scheduler, which would allow the execution of arbitrary tasks according to scheduling rules
- (ii) A proper HTTP export API, web-oriented and exploitable using different technologies and protocols
- (iii) A *push-based*, close to realtime synchronization mechanism that could periodically export event data to other services, such as search engines, thus replacing the current use for the OAI gateway.

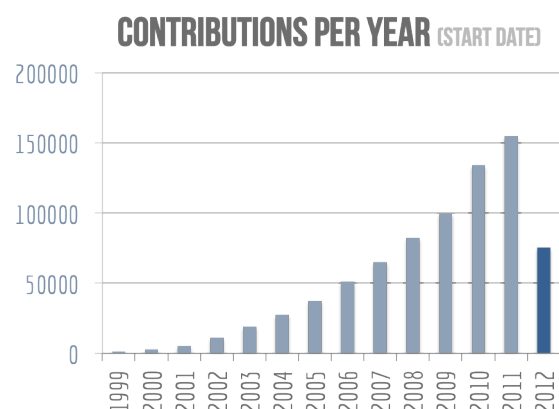


Figure 2. Number of contributions currently stored at CERN's Indico instance, grouped by year of their start date

In addition to these improvements, several “slow spots” were identified and optimized. A general caching mechanism was as well put in place, which is capable of communicating with `memcached`. `memcached`’s distributed nature opens the door to caching large volumes of data in a scalable and performant way.

3. A Collaboration Hub

Today’s highly distributed scientific community mandates the existence of collaborative tools. Projects such as the Large Hadron Collider require that researchers be connected through several means, such as videoconference and chat applications. However, if there is no place to centralize all those services, the coordination of meetings can be a complex job. Indico was a natural choice for such a role - the system already contained the information pertaining to all events, and adding a module that would provide all required features was a much easier task than developing a new system from scratch that would then, in some way, integrate with Indico. The latter would simply act as a hub, an intermediary between users and other applications, which would store metadata on bookings and bind them to specific events[2].

The following types of collaborative service are currently supported by Indico:

- Videoconferencing tools - The first interface to be implemented made the bridge with EVO² (the then standard videoconferencing system at CERN). EVO was recently replaced with Vidyo³ as CERN’s main system and, as such, an appropriate plugin was developed for Indico. Besides those two services, a third service called “CERN MCU” (room to room conferencing) is supported
- Recording/Webcast Requests - Indico also works as a workflow tool for those two systems, that allow users to, respectively, request the recording of specific lectures and the broadcast, over the internet. Authorization is automatically requested to speakers, who can sign an electronic form confirming that they accept that their talks be published on the web
- XMPP/Jabber chat - The XMPP[5] standard is currently used by large-scale services (such as Google Talk and Facebook Chat) in order to provide millions of users with a reliable and fail-safe chat infrastructure. It is the natural choice for the cornerstone of a chatting infrastructure, which is why it is the chosen chat system for Indico to integrate with. An experimental XMPP service has been running at CERN for more than two years, for which Indico acts as a management tool. Through Indico, organizers are able to create chat rooms for their events, which get advertised on the event page. With the help of a Jabber web client (Jappix⁴), users are able to join the discussion with a single click. A standalone XMPP client can be used, and XMPP federation is supported by the server (people may log in using their GTalk/Facebook/jabber.org accounts).

4. HTTP API

In today’s web, every major web application needs a proper HTTP API. Regardless of the openness (or lack thereof) of an application, the ability to make use of the data that it provides is instrumental in creating derivative products that will cater to additional features and an influx of users. The existence of Web APIs seem to be an important matter for large and medium scale services. The development of Indico’s own HTTP API was triggered not only by the aforementioned need for a mechanism that could replace the existing export script with a sounder and more performant solution, but also by the desire for a clean, extensible and

² <http://evo.caltech.edu>

³ <http://www.vidyo.com>

⁴ <https://project.jappix.com/>

maintainable module for metadata extraction - it was designed as a gateway to the web of data, one that can be adapted to new concepts, formats and paradigms, as Indico and the web itself change and evolve. The objective was to make available the same information that the web interface provides in a “computer-friendly” way. The following formats are currently supported:

- XML
- JSON (as well as JSONP)
- HTML (with basic CSS classes, as to allow easy styling/customization)
- iCalendar
- Atom.

The presence of these formats ensures that virtually any web application can easily make use of the information provided by Indico.

Currently, the following data is available through the API:

- List of events inside a category
- Details of a single event (dates, description, speakers, contributions)
- Rooms that are registered in the Room Booking system
- Room bookings
- Recording/Webcast request bookings.

The level of detail of the returned data can be chosen using a query parameter, thus allowing API users to choose making a request that returns only minimal information instead of one that returns further detail at the cost of longer waiting time. As such, information about events can go from the basic “direct” attributes of said event down to the subcontribution level. The API is made available using a hierarchical and intuitive URL scheme, which requires at least an identifier for the entity that is being requested and the format that information should be returned in. As an example, suppose a user wants to obtain the data concerning the event with id 314, in JSON format - that would mean URL `’/export/event/314.json’`, relative to the server in question. Another example would be the list of events inside a given category with id 42, in XML - `’/export/categ/42.xml’`. The result set can then be narrowed down through the use of filters such as date (whether the start date of the event is located between two given dates), location, maximum number of results and others. More information can be found in the official documentation⁵.

Authentication

Data protection is a priority in an application like Indico, and developing a new HTTP interface meant finding an authentication scheme that could accomplish all the security requirements that were presented not only by general security practices, but by CERN’s internal regulations (since CERN is Indico’s main “client”). At the same time, the ease of use of an application tends to be affected by security mechanisms that are too rigid - a balance had to be found, something that would be secure enough and easy to work on at the same time.

After investigating current practices in web APIs, a scheme that involves an API key (unique user identifier) and a secret token was devised, which uses HMAC[6] as a way to guarantee data integrity and authenticity of each API request. The implementation of said mechanism was straightforward, as is the development of applications that can make use of it, since there are HMAC-enabled libraries available for several platforms and programming languages.

⁵ <http://indico.cern.ch/ihelp/html/ExportAPI/>

The future adoption of a more complex authentication protocol such as OAuth is not out of question. The main limitation of the current authentication scheme is that users need to provide trusted applications that make use of the API with their secret token - this may be undesirable in some cases. OAuth solves this problem by allowing different applications to request their own access tokens, while doing it in a standard and user-transparent way.

The security level at which an Indico server's API works may be tuned by the administrator. It can range from public access to every resource to a completely close scheme in which HMAC-signed requests are required for any resource. The recommended setting is actually a compromise between the two, allowing access to unprotected content through the use of the API key, and to protected data with a HMAC signature based on the security token.

5. Better User Experience

After the success of Indico's first experience with usability testing, back in 2007/2008[1], it was clear that the benefits that were being drawn from such a procedure had become too many to be ignored. User experience testing is an example of a skill that has been acquired, over time, by the Indico Team. This procedure has taken the form of regular discussions (at the Indico meetings, in which peers evaluate the quality of created interfaces and provide feedback) and periodical, more detailed, less informal tests. Such an example was that of Giuseppe Certo, who in 2011, in the context of his Master Project at EPFL⁶/CERN re-applied the procedure that had been undertaken 4 years before, in an improved and more mature form. This included, as it had already years before, actual "lab tests" with real Indico users, selected from different "strata" of the project's user community (frequent users, infrequent users, experienced, unexperienced...).

The outcome of Giuseppe's research has shown that, even though there is room for improvement (and the development plan for versions 0.99 and 1.0 includes several of his suggestions), Indico is on the right track - global user satisfaction has seen a significant improvement since 2007 (Fig. 3), from almost 20% negative evaluations in 2007 to around 4% in 2011. Likewise, the number of users giving Indico the maximum rating in satisfaction grew from less than 9% to 15% over those same almost 4 years. This is, undoubtedly, good news, and it would be good that the trend remains the same. Detailed figures and findings can be found in Giuseppe's Master Thesis[3].

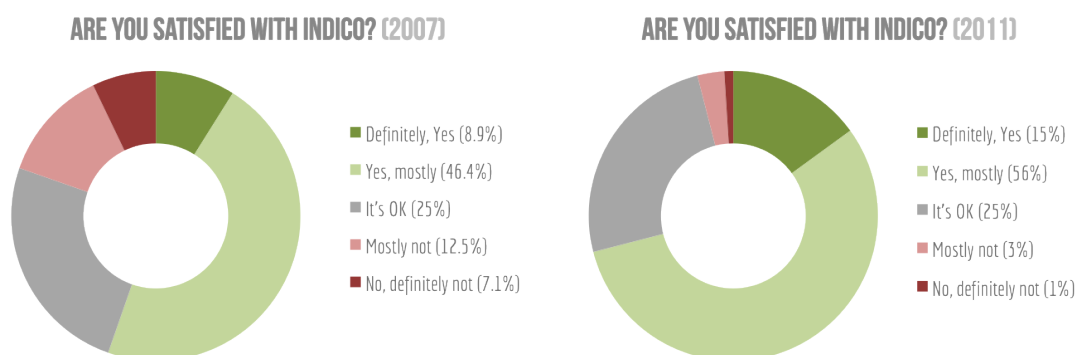


Figure 3. Comparison between the usability surveys of 2007 and 2011.

In parallel with the aforementioned usability evaluation, several parts of the interface were improved, some of them designed prior to 2007, some few others reworked even further. Such is the case of the timetable, that having been completely redesigned throughout 2008 (in favor

⁶ École Polytechnique Fédérale de Lausanne

of a JavaScript-based, fast response, intuitive version) has finally reached the point at which users can just drag and drop time blocks as if it were in a desktop application. This was a long time ambition and feature request, which further optimizations in the back end code and the adoption of lightweight, powerful JavaScript libraries such as jQuery and jQueryUI made possible.

Other very important and useful additions to Indico's feature set are:

- A Room Map - a Google Maps-based interface displaying rooms based on their geographical location and allowing different filtering criteria
- Per-event statistics - event managers can now know exactly how many people visit their event sites, and some "demographic" indicators. This was built in the form of a plugin that is capable of importing this data from an external analytics app (a default implementation for Piwik⁷ is provided)
- Paper Reviewing workflow - conference organizers can now set up a workflow that lets authors submit papers and designated referees to evaluate them according to predefined questions and criteria.

6. The Future

It is undeniable that one of the markets that is currently in explosive growth is that of handheld devices, such as smartphones and tablets. In line with this demand, which affects Indico users in the same way as the general public, a new mobile Indico application is in development. It will be an autonomous add-on layer on top of Indico, powered by the HTTP API, which will provide a simple and functional mobile-friendly interface. A first version should be available by the end of the year.

At the time of writing, the next Indico release is foreseen to be 0.99, due in July 2012. Indico 1.0 should be available for use at CERN in January 2013 and released shortly thereafter. The latter will include several more user experience improvements as well as a substantial amount of refactored code. Its main focus will be uniformizing the remaining parts of Indico that are still using old technologies and interfaces. In spite of the significant change in the release number, 1.0 will be, like many others, a normal Indico release, with the particularity of marking the affirmation of Indico as a mature web application with sound development process, proper documentation and a growing community that actively uses it. More than a revolution, it will be yet another step in an evolution that has been gradual and accelerated at times, but that has brought it to be the tool of election for event organizers in the HEP community and a piece of technology that owes nothing to comparable players from the industrial sector.

7. Conclusion

The Indico Project has come a long way since its inception ten years ago. After roughly eight years during which the main objective was the growth of the application's feature set, priorities shifted to user experience and, subsequently, performance. In the course of the last three years, a significant amount of the project's budget was spent as well in the creation of modules that integrate with different collaborative tools in the videoconferencing and chatting realms. This has enabled Indico to keep its standing as the number one collaborative tool in High Energy Physics, and to spill over into other scientific and computing-related communities⁸. We hope that the coming release of Indico 1.0 will allow the application to cross the boundaries of the scientific/academic community and to acquire a larger following in the industrial environment. The investments that have been and are currently being made in a better integration with other

⁷ <http://piwik.org/>

⁸ List of known Indico instances: <http://indico-software.org/wiki/IndicoWorldWide>

applications (HTTP API) and support for different devices (Mobile Indico) are aimed at keeping pace with the current trends in terms of web technology, and making sure that Indico will remain a valid and feature-rich option for event organization in any particular environment. The future of Indico depends not only on the effort that is invested by its core team, at CERN, but more and more on the contributions of its users, server administrators and developers world wide. A solid community of users already exists around the project, but growing demand for new features and improvements (and their maintenance) requires a comparable increase in human capital, which can only be achieved through everyone's collaboration. From translation to development of small features, fixes or just bug reporting, any contribution will for sure be welcomed by the community.

References

- [1] Gonzalez Lopez, J B and Ferreira, J P and Baron, T 2010 Indico Central - Events Organisation, Ergonomics and Collaboration Tools Integration *J. Phys.: Conf. Ser.* 219 (2010) 082002 doi:10.1088/1742-6596/219/8/082002
- [2] Fernandes, J and Bjorkli, K and Martin Clavo, D and Baron, T 2010 Virtuality and Efficiency - Overcoming Past Antinomy in the Remote Collaboration Experience *J. Phys.: Conf. Ser.* 219 (2010) 082006 doi:10.1088/1742-6596/219/8/082006
- [3] Certo G 2011 Usability Study of Indico *Master Thesis - École Polytechnique Fédérale de Lausanne*
- [4] MySQL AB 2008 Designing and Implementing Scalable Applications with Memcached and MySQL *MySQL White Paper*
- [5] Saint-Andre P 2011 RFC 6120: Extensible Messaging and Presence Protocol (XMPP): Core *Internet Engineering Task Force (IETF)* ISSN: 2070-1721
- [6] Krawczyk H, Bellare M, Canetti R 1997 RFC 2104: HMAC: Keyed-Hashing for Message Authentication *Internet Engineering Task Force (IETF)*