

## InPhO for All: Why APIs Matter

Jaimie Murdock, Indiana University  
Colin Allen, Indiana University

### Abstract

The unique convergence of humanities scholars, computer scientists, librarians, and information scientists in digital humanities projects highlights the collaborative opportunities such research entails. Unfortunately, the relatively limited human resources committed to many digital humanities projects have led to unwieldy initial implementations and underutilization of semantic web technology, creating a sea of isolated projects with data that cannot be integrated. Furthermore, the use of standards for one particular purpose may not suit other kinds of scholarly activities, impeding collaboration in the digital humanities. By designing and utilizing an Application Platform Interface (API), projects can reduce these barriers, while simultaneously reducing internal support costs and easing the transition to new development teams. Our experience developing an API for the Indiana Philosophy Ontology (InPhO) Project highlights these benefits.

### Introduction

The unique convergence of humanities scholars, computer scientists, librarians, and information scientists in digital humanities projects highlights the collaborative opportunities such research entails. The digital humanities aspire to create, maintain, and deploy high integrity metadata that are derived from the activities and feedback of domain experts in the humanities, to support scholarly activities which meet the high standards of academic peer review. Unfortunately, the relatively limited human resources committed to many digital projects for the humanities have led to unwieldy initial implementations and underutilization of semantic web technology, with the result that most projects in this burgeoning field are standalone projects whose data cannot easily be integrated with others. In addition to the barriers arising from idiosyncratic implementations, the difficulties of integrating data from multiple sources are compounded by the use of standards that serve one particular purpose well but do not facilitate other kinds of scholarly activities, often making the combination of resources from different projects laborious and expensive. Thus, much of the potential for collaboration in the digital humanities still remains to be unlocked.

Even humanities scholars who are not programmers should care about the ad hoc nature of application integration, because so much of their time involves laboriously transferring what they learn in one digital context to what they do in another. For example, the Stanford Encyclopedia of Philosophy (SEP)<sup>1</sup> and PhilPapers<sup>2</sup> are the two most widely used online resources for philosophers. But if a PhilPapers user wishes to know which SEP entries cite an item listed in the PhilPapers bibliography (or elsewhere online), the citation's information must be manually copied and pasted from PhilPapers into the SEP search engine in order to perform the search. In the other direction, PhilPapers now provides a service to the SEP whereby a link in each SEP entry leads to a page at PhilPapers showing the items from the entry's bibliography as represented in PhilPapers. However, the idiosyncratic formats of both the SEP and PhilPapers mean that there is no corresponding

---

<sup>1</sup> <http://plato.stanford.edu>

<sup>2</sup> <http://philpapers.org>

service in the other direction, that there is only a partial correspondence between items in the SEP bibliography and PhilPapers, and that this special purpose software cannot be easily be redeployed by other developers of online resources for philosophers.

Without easy access to preferred representations, the social and semantic web cannot be quickly adapted to the needs of researchers in the humanities.<sup>3, 4</sup> And while the needs and goals of librarians have been important drivers of standards in the digital humanities, this represents just one aspect of the potential of the digital humanities to facilitate scholarly research. Humanities scholars need access to the data in many different representational formats: from HTML for the ordinary end user to fully integrated XML specifications and raw data dumps for the information scientist, and to lightweight JSON stores for the web programmer.

### The Indiana Philosophy Ontology Project

The Indiana Philosophy Ontology Project (InPhO)<sup>5</sup> aims to overcome barriers to broader collaboration by providing a simple, lightweight API (application programming interface) capable of serving a wide variety of data formats. APIs allow programmers to focus on the ‘what’ of computing rather than the ‘how.’ So, for instance, it is an API that allows programmers to tell the computer’s operating system to respond to a mouse click by opening a “window” on the screen, without the programmer having to worry about the graphics needed to produce a rectangle of a certain size, border, color, etc. Similarly, programmers can exploit databases on another server through an API without having to know what the underlying database model is on the remote server. APIs give power to programmers by allowing them to stand on the shoulders of others.

At the InPhO project, we have a vision of seamless integration among digital philosophy applications, and our API is a deliberate first step towards realizing that vision. The InPhO is a dynamic computational ontology which models philosophy using statistical methods applied to the entire SEP corpus,<sup>6</sup> as well as machine reasoning methods applied to feedback from experts in the field, particularly the editors and authors of SEP entries. Our approach<sup>7</sup> begins with a small amount of manual ontology construction and the development of an initial philosophical lexicon through collaboration with domain experts. We then build on this foundation through an iterative three-step process to create a taxonomic representation of philosophy. First, statistical inference over the SEP is used to generate hypotheses about the relations among various topics, including the relative

---

<sup>3</sup> Leonard Richardson and Sam Ruby, *RESTful Web Services* (O’Reilly Media, Inc.: 2007).

<sup>4</sup> Sinuhé Arroyo, Rubén Lara, Juan Miguel Gómez, David Berka, Ying Ding, and Dieter Fensel, “Semantic aspects of web services,” in *Practical Handbook of Internet Computing*, ed. Munindar P. Singh (Baton Rouge: Chapman Hall and CRC Press, 2004), pages 31–1 – 31–17.

<sup>5</sup> <http://inpho.cogs.indiana.edu>

<sup>6</sup> The SEP contains over 1,200 entries comprising more than 14 million words, maintained by over 120 volunteer subject editors, and accessed through more than 700,000 entry downloads per week.

<sup>7</sup> detailed in Cameron Buckner, Mathias Niepert, and Colin Allen, “From encyclopedia to ontology: Toward a dynamic representation of the discipline of philosophy,” *Synthese* (2010). Special issue on Representing Philosophy, in press. <http://dx.doi.org/10.1007/s11229-009-9659-9>.

generality of pairs of terms.<sup>8</sup> These hypotheses are then evaluated by domain experts through simple questions that do not require any knowledge of ontology design on the part of the experts. Finally, the expert responses are combined with the statistical measures as a knowledge base for a machine reasoning program, which uses answer set programming to output a taxonomic view of the discipline that synthesizes the sometimes inconsistent data obtained by querying experts.<sup>9</sup> This resulting representation can then be used to generate tools that assist the authors, editors, and browsers of the SEP, such as a cross-reference suggestion engine, access to bibliographic content, context-aware semantic search, and interfaces for exploring the relations among concepts, among philosophical thinkers, and between concepts and thinkers.

InPhO does not assume that a single, correct view of the discipline is possible, but rather takes the pragmatic approach that some representation is better than no representation at all.<sup>10</sup> Even if other projects do not agree with our final taxonomic projections, our statistical data and expert evaluations may still be useful. By exposing our data through the API at all three steps of the process outlined above, we encourage other projects to discover alternative ways to construct meaningful and useful representations of the discipline. Furthermore, by exposing our data in this way, others may explore try alternative methods for generating representations of the discipline.

## Design Considerations

The use of APIs by other projects requires our accountability and necessitates permanent availability. The high cost of redesign under these conditions implies that we have one chance to get the access-layer right.<sup>11, 12</sup> To do this, we used one of the most venerable and pervasive technologies—the hypertext transfer protocol (HTTP)<sup>13</sup> that is the foundation of the World Wide Web—to enable ease of use by scholars, programmers, and scientists through nearly any interface. Each entity in the InPhO knowledge base is exposed as a resource with a unique Uniform Resource Identifier (URI) which is accessed using the HTTP methods, providing a consistent interface for data retrieval and manipulation. This is known as the REpresentational State Transfer (REST) paradigm of web services, pioneered by HTTP inventor Roy Fielding.<sup>14</sup> The InPhO data can be explored via human-

---

<sup>8</sup> Mathias Niepert, Cameron Buckner, and Colin Allen, “A dynamic ontology for a dynamic reference work,” in *Proceedings of the 7th ACM/IEEE Joint Conference on Digital Libraries* (2007), 288–297.

<sup>9</sup> Mathias Niepert, Cameron Buckner, and Colin Allen, “Answer set programming on expert feedback to populate and extend dynamic ontologies,” in *Proceedings of the 21st International FLAIRS Conference* (Coconut Grove, Florida: AAAI Press, 2008), 500–505.

<sup>10</sup> Buckner et al., “From encyclopedia to ontology.”

<sup>11</sup> Joshua Bloch, “How to design a good API and why it matters,” in *OOPSLA '06: Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications* (New York: ACM, 2006), 506–507.

<sup>12</sup> Toby Segaran, Colin Evans, and Jamie Taylor, *Programming the Semantic Web*. O'Reilly Media, Inc., 2009.

<sup>13</sup> Roy Fielding, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and Tim Berners-Lee, “Hypertext transfer protocol - http/1.1. RFC 2616,” *Network Working Group*, June, 1999, accessed February 27, 2010, <http://www.w3.org/Protocols/rfc2616/rfc2616.html>.

<sup>14</sup> Roy Fielding, *Architectural Styles and the Design of Network-based Software Architectures*, PhD thesis, University of California, Irvine (2000).

friendly HTML or in machine-friendly JSON, selected by simply adding either .html or .json to the URI of each resource.

This approach has many advantages over previous attempts at integration. For example, data dumps provided using industry-standard Web Ontology Language files (idiosyncratically referred to as OWL files) reveal only certain types of relations and do not allow for read-write access to the underlying knowledge base. While OWL remains an important format for exchange of data, to limit oneself to that format would place significant limits on collaborative efforts, such as InPhO's partnership with Noesis<sup>15</sup> to power their domain-specific search engine. The Noesis project currently has no need to receive InPhO's entire ontology file when seeking to query specific pieces of information about a journal, a philosophical concept, or a thinker from the InPhO database. Instead, Noesis programmers may use InPhO's RESTful API to easily select only those entities and partitions of the InPhO which are relevant to the current query. Other projects are likely to have similar requirements—a project tracing the history of a specific philosopher could initially pull selected data only from the thinker database, but could easily branch out to other portions of the database as connections between that thinker and specific concepts become relevant to an end-user's online searching and browsing behavior. At the same time, this approach to data sharing protects data providers from the overexposure that may result from making large data dumps available to all comers, while easing the processing load for data consumers who might otherwise have to parse masses of unwanted data.

The design of the API also satisfies Crane's rubric for digital humanities infrastructure:<sup>16</sup> 1) By providing a unique URI, we have created canonical named entities for each concept within the ontology. These entities are aliased within our knowledge base with alternative spellings or abbreviations, increasing the likelihood of identifying objects correctly. This technology is being used by the Noesis project's journal search. 2) Our machine learning and data mining techniques create a co-occurrence graph which is exposed through the API as a dynamic cataloging service for philosophical concepts. 3) Structured user contributions are invited through secure write access to improve the quality of the knowledge base. Evaluations will be solicited throughout the SEP editorial process. 4) These are then used to provide custom, personalized data and tools for researchers, such as the SEP cross-reference engine. The design also satisfies the computer science community, by providing a concrete example of a semantic web portal, as envisioned by Stollberg et al.<sup>17</sup>

## Benefits

Our experience shows the development of an API is not just an exercise in enhancing collaboration with other projects, but can alleviate internal management concerns about sustainability and efficiency. Due to the nature of academia, turnover happens regularly on a three to five year cycle as students working as programmers and researchers on the project progress from matriculation to graduation. New project members must be quickly integrated with our development process and

---

<sup>15</sup> <http://noesis.evansville.edu>

<sup>16</sup> Crane et al., "The humanities in a global e-infrastructure."

<sup>17</sup> Michael Stollberg, Holger Lausen, Rubén Lara, Ying Ding, Sung-Kook Han, and Dieter Fensel, "Towards semantic web portals," in *Proceedings of the WWW2004 workshop on Application Design, Development and Implementation Issues in the Semantic Web*, eds. Christoph Bussler, Stefan Decker, Daniel Schwabe, and Oscar Pastor. (New York: 2004).

fluent in our existing code base. Our initial architecture consisted of a decentralized, uncoupled multitude of quick scripts and interfaces, driven by the necessity of having a proof of concept. This led to difficulty in turnover, and highlighted a need for maintainable, documented code. Additionally, this loosely coupled architecture was resistant to scalability. Many of our scripts required a sequence of coupled events and were often executed by hand. As evaluations continued to trickle in, parts of our database became inconsistent leading to integrity issues and requiring manual cleanup during the ontology extension process. With all data access occurring at a single point, IT demands were reduced, as maintenance of SQL data connections and secure data access tunnels was replaced with the maintenance of the website. By porting out internal tools to use the same API calls, we can use our internal code as public examples.

## Conclusions

While there exist other APIs for humanities computing, these have usually been developed by groups seeking to provide easy access to large cultural collections such as those held by libraries and museums. To our knowledge, we are the first project to have developed an API for access to information about the dynamically changing concepts, people, and institutions defining an academic discipline, and to create a mechanism for partner projects to contribute to our database, bridging the gap between social and semantic web. We are certainly the first to do this for the field of philosophy. The lessons learned in carrying out this project will, we hope, encourage other scholarly communities to pursue similar projects to make the conceptual structure and human capital of their field readily accessible for applications that have not yet been dreamt of, and will enable such projects to avoid some of the early problems with design that arose from an application-centric view of the web, as opposed to the service-oriented semantic web.

## Bibliography

- Arroyo, Sinuhé, Rubén Lara, Juan Miguel Gómez, David Berka, Ying Ding, and Dieter Fensel. "Semantic aspects of web services." In *Practical Handbook of Internet Computing*, edited by Munindar P. Singh, 31–1 – 31–17. Baton Rouge: Chapman Hall and CRC Press, 2004.
- Bloch, Joshua. "How to design a good API and why it matters." In *OOPSLA '06: Companion to the 21st ACM SIGPLAN symposium on Object-oriented programming systems, languages, and applications*, 506–507. New York: ACM, 2006.
- Buckner, Cameron, Mathias Niepert, and Colin Allen. "From encyclopedia to ontology: Toward a dynamic representation of the discipline of philosophy." *Synthese* (2010). Special issue on Representing Philosophy, in press. <http://dx.doi.org/10.1007/s11229-009-9659-9>.
- Crane, Gregory, Brian Fuchs, and Dolores Iorizzo. "The humanities in a global e-infrastructure: a web-services shopping-list." UK e-Science All Hands Meeting (2007).
- Fielding, Roy. *Architectural Styles and the Design of Network-based Software Architectures*. PhD thesis, University of California, Irvine (2000).
- Fielding, Roy, J. Gettys, J. Mogul, H. Frystyk, L. Masinter, P. Leach, and Tim Berners-Lee. "Hypertext transfer protocol - http/1.1. RFC 2616," *Network Working Group*, June, 1999. Accessed February 27, 2010. <http://www.w3.org/Protocols/rfc2616/rfc2616.html>.

- Niepert, Mathias, Cameron Buckner, and Colin Allen. "A dynamic ontology for a dynamic reference work." In *Proceedings of the 7th ACM/IEEE Joint Conference on Digital Libraries* (2007): 288–297.
- Niepert, Mathias, Cameron Buckner, and Colin Allen. "Answer set programming on expert feedback to populate and extend dynamic ontologies." In *Proceedings of the 21st International FLAIRS Conference*, 500–505. Coconut Grove, Florida: AAAI Press, 2008.
- Richardson, Leonard and Sam Ruby. *RESTful Web Services*. O'Reilly Media, Inc.: 2007.
- Segaran, Toby, Colin Evans, and Jamie Taylor. *Programming the Semantic Web*. O'Reilly Media, Inc.: 2009.
- Stollberg, Michael, Holger Lausen, Rubén Lara, Ying Ding, Sung-Kook Han, and Dieter Fensel. "Towards semantic web portals." In *Proceedings of the WWW2004 workshop on Application Design, Development and Implementation Issues in the Semantic Web* edited by Christoph Bussler, Stefan Decker, Daniel Schwabe, and Oscar Pastor. New York: 2004.