Abstract

"Local Caching of External Linked Data Authorities in Hydra"

Linked data allows libraries to build more manageable and interoperable datasets, thereby providing the opportunity to connect their collections to similar content that exists across the web. Yet in order for libraries to fully realize the possibilities linked data affords, they must develop new technical infrastructure and workflows. The Chemical Heritage Foundation (CHF) has joined a growing community of libraries that are beginning to implement linked data in the open-source platform Hydra. Hydra's Sufia 6 models information about its objects using the Resource Description Framework (RDF), bringing metadata in-line with semantic web standards. To streamline cataloger workflow, CHF has integrated Application Program Interfaces (APIs) into its metadata creation template to offer autocomplete functionality that suggests terms from linked data vocabularies maintained by external authorities such as OCLC or Library of Congress.

However, in trying to implement linked data in Hydra's Sufia 6, CHF has encountered several problems that are indicative of larger issues faced by the Hydra community and beyond. The default setting in Sufia is to store only the human-readable text without its associated machine-readable Uniform Resource Identifier (URI), thereby missing a key component for linked data functionality. Furthermore, an autocomplete thesaurus API may only be offered for one controlled vocabulary per metadata element, resulting in catalogers supplementing the primary authority by manually searching other external thesauri, a time-consuming workflow that increases the possibility of human error. Finally, when URIs are used, load times for collection webpages can be slow due to the number of additional calls the page must make to external servers responsible for hosting the authorities.

From Oct. 2016 – Sept. 2017, CHF will work with the consultants Data Curation Experts (DCE) to integrate an RDF triplestore (a database designed to store RDF triples) as an optional additional component in the Hydra architecture in order to locally cache externally maintained linked data authorities. CHF and DCE will also produce open-source software code that will provide enhanced functionality in the default metadata template, allowing a cataloger to begin typing a name into a field and see results from multiple authorities. When a name is selected from the autocomplete options, Hydra would then store both the human-readable text and the machine-readable URI. The new functionality proposed will be limited to autocompleting and locally caching vocabularies in the Creator element in Hydra's Sufia metadata template by using name authorities found in OCLC's Virtual International Authority File (VIAF) and Faceted Application of Subject Terminology (FAST). However, the design will allow for easy integration of additional linked data vocabularies in the future. The ability to locally cache results for data maintained by external authorities will also enhance the end user experience by speeding up delivery and ensuring reliability.

Our primary audience comprises librarians, archivists and museum professionals who manage their collections with Hydra, and potential future adopters of the IMLS-funded Hydra-in-a-Box application. However, we expect this project to attract attention outside the Hydra community, as it will also serve as a best practice model for how libraries can still exert some local control over the linked data and services they rely on others to maintain and provide. This project is aligned with the IMLS's National Digital Platform and its agency-level goal to improve management of content and collections by providing a strategic solution to implementing and managing linked data.

"Local Caching for External Linked Data Authorities in Hydra"

Statement of Need

Linked data allows libraries to build more manageable and interoperable datasets, thereby providing the opportunity to connect their collections to similar content that exists across the web. The Chemical Heritage Foundation (CHF) has joined a growing community of libraries that are beginning to implement linked data in the open-source platform Hydra (a technology stack comprising Fedora 4, Solr, Blacklight and Ruby on Rails).[1] Hydra relies on Fedora 4 as an open-source repository layer for managing and preserving collection data. Instead of creating locally-maintained bespoke configurations of EXtensible Markup Language (XML) (which was used in Fedora 3 and continues to be the structure behind many other collections management systems), Fedora 4 follows the Resource Description Framework (RDF) to bring metadata in-line with semantic web standards as set by the World Wide Web Consortium (W3C).

CHF currently uses Hydra's Sufia 6.4.0, the most robust and popular Hydra repository solution currently available. The forthcoming Sufia 7.0 will form the basis of the IMLS-funded Hydra-in-a-Box project.[2] Because Hydra's Sufia uses Fedora 4, which models information about its digital objects in RDF, the dream of linking collection metadata to data authorities is an exciting and tangible possibility. Furthermore, through the use of an Application Program Interface (API), a linked data thesaurus can be pulled from an external authority such as those provided by the Online Computer Library Center (OCLC) or Library of Congress (LC) where that data is hosted and maintained,[3] thereby offering the cataloguer autocomplete functionality in the Hydra metadata template that streamlines workflow.

However, in trying to implement linked data in Hydra, CHF has encountered several problems that are indicative of larger issues faced by the community. While linked data is possible in Fedora 4, Sufia's default setting is to only store the human-readable RDF string value or label ("Benjamin Franklin") but not with its associated machine-readable Uniform Resource Identifier (URI): <http://id.worldcat.org/fast/34115>. As such, a key component for linked data functionality is missing. Further, while the name may be entered in accordance with external controlled vocabulary standards, without the associated URI there is no way to identify from which authority that name derived, resulting in a loss of data provenance that is important for quality control and long-term maintenance. In trying to integrate linked data functionality into Sufia, CHF has experienced many challenges aligned with use cases that have been presented across the wider Hydra community. Hydra Connect 2015 presentations on linked data by Trey Terrell (Princeton University)[4] and Hector Correa (Penn State University)[5], as well as Steven Anderson's work-in-progress Metadata Enrichment Initiative for the Boston Public Library,[6] all identified similar unexpected metadata challenges and limitations in

[1] Project Hydra: http://projecthydra.org; Hydra Community: http://projecthydra.org/community-2-2/partners-and-more/
[2] Hydra-in-a- Box FAQ: http://hydrainabox.projecthydra.org/faq.html
[3] Library of Congress Linked Data Service - Authorities and Vocabularies: http://id.loc.gov/
[4] Trey Terrell, "Linked Data, Labels, URIs": http://www.slideshare.net/JosephSchmoseph1/linked-data-labels-uris
[5] Hector Correa, "Introduction to the Linked Data Platform (LDP)": http://www.slideshare.net/hectorwashere/introduction-to-linked-data-platform-ldp
[6] The Metadata Enrichment Initiative for Hydra: https://github.com/boston-library/mei. While extraction and Ruby gemification is expected to take place over February 2016, some code currently exists in the Digital Transgender Archive: https://github.com/CollegeOfTheHolyCross/dta_sufia

Sufia’s data model. What has become apparent is the need to supplement Fedora 4 with a local database for caching RDF “triples” (the three components <subject> <predicate> <object>), but no turn-key solution is available and potentially valuable code is dispersed among existing Hydra projects.

Similarly, CHF’s attempts to integrate autocomplete functionality into Hydra’s Sufia metadata creation template resulted in a number of challenges. Presently, the application uses OCLC’s API for Faceted Application of Subject Terminology (FAST) to make controlled vocabulary suggestions from the FAST thesaurus once a cataloger starts typing a name into a field in the metadata template. However, the default functionality stores only that name, but not the URI associated with it. Furthermore, the default interaction in the metadata template is limited to displaying results from a single API. For example, the Creator field can be linked to FAST or OCLC’s Virtual International Authority File (VIAF), but not to both. If the name does not exist in the primary thesaurus used for autocomplete, then the cataloger would need to look up the name in another external source and manually enter or paste it into the system. It can be difficult to motivate librarians to integrate an additional manual entry step into an already labor-intensive workflow. Autocomplete functionality speeds up the time-consuming task of creating metadata, and catalogers want to easily look up authorities in multiple sources without adding steps to their workflow and increasing the chance of human error. At an institution like CHF that has diverse collections drawn from its Library, Archive, Museum and Center for Oral History, the ability to easily manage multiple authority options is critical.

In addition to complications in metadata standards and staff workflows, linked data has the potential to introduce new challenges to the end user experience once URIs are incorporated. If the external server maintaining the metadata is down, a search result on an institution’s digital collection website could be slow or even deliver blank fields for that metadata content. Yet even if all external hosts are working, relying on linked data can still result in slower webpage load times because the page needs to make calls to additional servers to load the content. Further, some hosts even impose usage limits on the number of calls their server can receive per minute and many of these are surprisingly low, such as one call per three seconds.[7] The speed of loading an item page can be slowed even further if the library chooses to store HTTP URIs linking to multiple authorities (for example, FAST, VIAF and the Getty’s Art and Architecture Thesaurus (AAT) for different elements in a single item’s record) because these would all require the page to make calls to additional servers. Without the ability to locally cache metadata, an institution cannot guarantee the speed, delivery or service of the content being served on their own collection website.

As noted in the summary document for the “2015 IMLS Focus: The National Digital Platform”, linked data provides a way to make connections between different platforms, such as the DPLA and Europeana, but it also “requires a shift in thinking about metadata as human-readable records to thinking about it as machine-readable entities”.[8] This shift in thinking also necessitates adjustments to established institutional workflows and, like any emerging technology, the actual implementation of it can present unexpected challenges. Jon Voss from Historypin has already advised the IMLS National Digital Platform to consider that linked data will not just easily slot into existing tools and systems, but must instead be built with new standards and shared protocols that can help reimagine workflows and

[7] This was a topic of concern at a recent Applied Linked Data Working Group meeting held on January 7, 2016. Minutes may be viewed here: https://wiki.duraspace.org/display/hydra/Applied+Linked+Data+2015-01-07

[8] “IMLS Focus: The National Digital Platform for Libraries, Archives and Museums” April 28, 2015. Washington, DC, p. 8: https://www.imls.gov/sites/default/files/publications/documents/2015imlsfocusndpreport.pdf

processes.[9] Across the Hydra community, the implementation of linked data has produced new challenges and engendered the creation of community groups dedicated to solving them. The annual Hydra Connect meet-up that was held in Minneapolis in September 2015 had a well-attended breakout session specifically entitled "Linked Data: What are the tools we need in Hydra?"[10] Many new and existing groups in the Hydra community have been actively engaging in these questions, such as the Hydra Triple Store Interest Group,[11] the Applied Linked Data Working Group,[12] and the Descriptive Metadata Working Group.[13]

The proposed solution to these problems is to build and document a best practice solution for adding an RDF triple store (a database designed to store RDF triples) into the Hydra technology stack, and to create open-source code that will add linked data functionality into Sufia's metadata template. The cataloger will be able to begin typing a name into a field and get results from multiple authorities, initially VIAF and FAST. When a name is selected from the autocomplete options, Hydra would store both the human-readable metadata authority text and the machine-readable URI, though only the former would be displayed to the end user. On the backend, the metadata will not only be stored in Fedora, but will also be cached in an RDF triple store built for maintaining external linked data. To begin with a manageable scope and budget, the new functionality proposed by this project will be limited to autocompleting and locally caching vocabularies in the Creator element in Hydra's Sufia metadata template by using name authorities found in VIAF and FAST. However, the design will allow for easy integration of additional linked data vocabularies in the future. The ability to locally cache results for data maintained by external authorities will also serve as a model for how libraries can still exert some control over linked data and the services they provide on their own website.

The infrastructure will be designed with an eye towards future expansion that will later offer additional functionality outside the scope of this grant, but these design choices will be finalized once the initial product has been implemented and more actual use cases have been collected from librarians using the improved metadata template in their workflows. Some future work could include the ability for the autocomplete options to show the number of times an entry has already been used locally, and the option to create new local authority entries for names which could not be found in external vocabularies. Once a triple store is in place, its functionality could be expanded to handle data about data that should be stored outside of Fedora 4. For example, disambiguation between multiple authorities (reconciling "Benjamin Franklin" URIs in FAST and VIAF as the same entry) and local corrections to external authorities (changing "Franklin, Benjamin" to "Benjamin Franklin") could all be managed in the triple store.

Impact

To ensure broad impact, CHF has already been talking with existing working groups in the Hydra community who are collecting applied linked data use cases and experimenting with new code. We will continue working with the wider Hydra community to ensure functionality and adoption

[9] Ibid., p. 8.

[10] E. Lynette Rayle, "Linked Data: What are the tools we need in Hydra?" Hydra Connect 2015. September 24, 2015. https://wiki.duraspace.org/display/hydra/Hydra+Connect+2015

[11] Hydra Triple Store Interest Group, "Challenges for Linked Data": https://wiki.duraspace.org/display/hydra/Challenges+for+Linked+Data

[12] Applied Linked Data Working Group: https://wiki.duraspace.org/display/hydra/Applied+Linked+Data+Working+Group

[13] Descriptive Metadata Working Group: https://wiki.duraspace.org/display/hydra/Descriptive+Metadata+Working+Group

beyond CHF, including posts on the Hydra-Tech and Hydra-Community Google Groups and via informal paths targeting Hydra software developers, including IRC and Slack channels. CHF intends to regularly attend monthly Skype meetings with related Hydra working groups throughout the course of the project. Updates and results will be shared throughout the development process and CHF will demo its implementation at the annual Hydra Connect conference that will be held in September 2017 in order to promote the codebase and engage potential new users and developers. The software code and documentation for the project will be made freely and publicly available on GitHub via an Apache 2.0 license, the Hydra community's standard open source license.[14]

To the fullest extent possible, this project will propose new functionality that can be built within or on top of existing Ruby gems in order to ensure the long-term maintenance of the code and encourage new commits from the Hydra community beyond the timeline for this project. For example, "Questioning Authority" provides a REpresentational State Transfer (REST) API for querying authorities and controlled vocabularies, and it might make sense to build on this project so that it can also query a local triple store.[15] "LinkedVocabs" provides tools for connecting to and loading terms into a local triple store from external linked data sources, and we may want to build on this gem to load single terms in addition to entire vocabularies.[16] "Oargun" uses "LinkedVocabs" and organizes external linked vocabularies used by Oregon Digital into different "authority" categories such as "creator," "subject," and "geographic".[17] Emulating this design choice could help make our project easier to build on when we want to extend to more types of fields. Collating disparate work and creating extensive documentation will offer a focal point around which the community can concentrate future development.

The project team will also keep an eye towards future code because developing something that is modular and extensible for use within the wider Hydra community is important to CHF. Sufia 7 will have a migration path to the forthcoming turn-key Hydra-in-a-Box application. Anna Headley, CHF's Library Applications Developer, was nominated for and currently serves on the Technical Advisory Group for Hydra-in-a-Box. As development continues, CHF will remain committed to maintaining our relationships with the Hydra-in-a-Box team and to updating the code associated with this additional software component, ensuring that it is an optional feature compatible with the new software. While the functionality we propose to create is not currently within scope for the larger Hydra-in-a-Box initiative, it is recognized as a highly desirable feature, as noted in the attached letter of support from Michael Giarlo, Technical Manager on the project.

Immediate success will be measured by the ability to create functional, modular code that can successfully be deployed. CHF will conduct the first test implementation, beginning by deploying the code to its staging server and checking for bugs and compatibility with the greater Hydra stack using Hydra's Sufia 7. Once all tests pass, CHF will deploy to its production server, access to which is currently limited to staff with IP addresses within CHF's Philadelphia office. We will then capture and share five use cases for how diverse in-house staff integrate the improved autocomplete metadata template into their workflows and how the new linked data functionality enhances our collection

[14] Chemical Heritage Foundation, code repository: https://github.com/chemheritage

[15] "Questioning Authority" code repository: https://github.com/projecthydra-labs/questioning_authority

[16] "LinkedVocabs: Linked Data Controlled Vocabularies for ActiveFedora::RDF" code repository: https://github.com/projecthydra-labs/linked_vocabs

[17] "Oargun: A Gem Created from Oregon Digital's controlled vocabulary" code repository: https://github.com/curationexperts/oargun

management abilities. With these initial measures, we will widely promote the new functionality as a successful linked data implementation model that other organizations can consider. We hope that other Hydra adopters will also choose to use our code, but we expect these long-term measures of success will not become apparent until Hydra-in-a-Box is fully realized. Community acceptance, reuse, and adaptation of our code would be difficult to measure within the one year grant period, but we will rely on communication with Hydra community groups throughout the project to ensure broader use later.

This innovative project also has the ability to attract attention far beyond the Hydra community, including any library professionals exploring how to best implement linked data for their collections. The integration of a triple store for locally caching external data authorities is also being discussed in the Islandora open source community, which is looking to integrate a triple store into its architecture with its Fedora 4 release.[18] While linked data is a buzz word in libraries, examples and best practice models of organizations that have actually implemented these new APIs and workflows are still hard to find, especially in small- to medium-sized cultural heritage institutions like CHF. As such, CHF staff members will also look for opportunities to connect with other organizations interested in implementing linked data, such as Linked Open Data in Libraries Archives and Museums (LODLAM).

Project Design

The project will be managed, developed and implemented by CHF's digital collections team with the help of the esteemed Hydra consulting company Data Curation Experts, comprised of three developers each with 10-20 years of software experience. Project management and budgetary oversight will be done by CHF's Michelle DiMeo, Curator of Digital Collections, who previously served these roles when she oversaw grants awarded to the Medical Heritage Library in her former role as Director of Digital Library Initiatives at the College of Physicians of Philadelphia. Upon notice of an IMLS award, Dr. DiMeo would announce the project to the wider Hydra community via its list servers and invite participation in providing use cases and considering design options. CHF team members Anna Headley, Library Applications Developer, and Cathleen Lu, Digital Projects and Metadata Librarian, will target specific Hydra partners already working on linked data via the communication channels with which they are already engaged, including the Applied Linked Data Working Group (Headley) and the Descriptive Metadata Interest Group (Lu). Ms. Headley will also serve as the project's Technical Lead and will communicate directly with Michael Giarlo, Technical Manager on the Hydra-in-a-Box project, to ensure that they maintain periodic check-ins throughout the course of software development. The team will also begin coordinating efforts in-house via CHF's Digital Collections Committee, comprised of representatives from its Library, Museum, Archives, and Center for Oral History, to enlist volunteers to test demos and provide feedback on functionality and workflows.

After two months of building teams and collecting use cases both in-house and within the wider Hydra community, CHF will work with Data Curation Experts to begin software development. CHF and DCE have already established a successful methodology for working together, as Ms. Headley previously worked with DCE to produce Ansible scripts that would automate CHF's deployment of Hydra on Amazon Web Services (AWS). Ms. Headley collaboratively wrote code with a consultant developer at DCE, using Skype and GitHub for communication. This methodology resulted in CHF using expert advice from consultants to build in-house staff expertise, ensuring sustainability beyond the period when consultants can be afforded by enriching local knowledge and empowering staff.

[18] Islandora 7.x-2.x Architecture: http://islandora-claw.github.io/CLAW/technical-documentation/architecture/

Software development is anticipated to take two months, as outlined by the attached Quote for Services by DCE, but will be spread over three months to integrate feedback and iterations. Planning will begin by researching triple stores and identifying Hydra code that may be relevant. DCE's consultant developers will then work with Ms. Headley to generate mock-ups and wireframes of the proposed new metadata template and functionality which Ms. Lu will share with CHF's in-house user group and interested parties in the wider Hydra community. This feedback will then be shared with Ms. Headley and DCE, and they will begin writing new code that builds upon the existing codebase for Sufia 7's metadata template. Using an iterative design approach, this functionality will then be tested by users to provide feedback. This information will help DCE and Ms. Headley to finalize their code on the metadata template and begin working on integrating the local triple store database. Ms. Headley will work primarily on how this database interacts with the Hydra application, and Daniel Sanford, CHF's Systems Administrator, will set up and configure the server infrastructure. Once in place, Ms. Headley and Mr. Sanford will work with DCE on configuration and deployment management scripting. Like all code produced for this project, CHF will work with DCE on testing and validation.

After development, CHF expects to use months six and seven of the grant period for implementation and testing. This will begin with in-house efforts, as CHF will deploy this code to its production server and ask catalogers to integrate the new metadata template into their workflow. Once satisfied with the in-house response, Ms. Headley and Ms. Lu will share screenshots and reports on internal workflows to targeted Hydra partners working on linked data and descriptive metadata with the hope that they might consider implementing this code. When we are confident in the performance of the code, Ms. Headley will work on extraction and consolidation with other relevant codebases and will write extensive documentation in order to make it easy for others to implement the code. Documentation will be found in a ReadMe file that will be available with the code on GitHub. The ReadMe file will include a description of the problem as well as technical notations related to dependencies, installation, and configuration. It will also include usage examples and development information, including how to run tests and how to contribute code back into the project.

We expect to use the final quarter of the project period to define more specific requirements and priorities for additional functionality based on user feedback, and to engage in an aggressive promotion and outreach effort. CHF plans to deliver a poster and possibly present a paper at the annual Hydra Connect conference in September 2017. However, CHF's digital collections team also plans to promote the project well beyond the Hydra community via its already established networks in digital library and digital humanities circles. Dr. DiMeo and Ms. Lu have been active in organizing, presenting at, and attending local conferences such as the Philadelphia Area Consortium for Special Collections Libraries (PACSCL) and THATCamp Philly (which has been held at CHF for 5 consecutive years), but also national conventions including MashCat (for catalogers and developers), Digital Library Federation (DLF), and the Keystone Digital Humanities Conference. Ms. Headley has also been an active member of Code4Lib for several years, even serving on the committee responsible for hosting the national convention in Philadelphia in March 2016, and she will promote the project formally and informally through her communications with and involvement in this community of library software developers. CHF's annual operating budget allows for staff to travel to conferences and CHF will contribute the costs of travel and attendance so that it may successfully promote the project and share discoveries with other library staff members interested in implementing linked data. While Hydra adopters (present and future) are our target audience, we expect to reach a diverse audience of digital library and linked data professionals.

	Oct. 2016	Nov. 2016	Dec. 2016	Jan. 2017	Feb. 2017	Mar. 2017	Apr. 2017	May. 2017	Jun. 2017	Jul. 2017	Aug. 2017	Sep. 2017
Build in-house user test group	■	■										
Work with related Hydra Working Groups	■	■	■	■	■	■	■	■	■	■	■	■
Research triple stores and existing Hydra code		■	■									
Software development			■	■	■							
Server Config. and Deployment Scripts					■	■						
User Testing and Implementation						■	■					
Software Documentation							■	■	■			
Collect use cases for future development								■	■	■	■	■
Outreach and Promotion										■	■	■

DIGITAL STEWARDSHIP SUPPLEMENTARY INFORMATION FORM

Introduction
The Institute of Museum and Library Services (IMLS) is committed to expanding public access to federally funded research, data, software, and other digital products. The assets you create with IMLS funding require careful stewardship to protect and enhance their value, and they should be freely and readily available for use and re-use by libraries, archives, museums, and the public. However, applying these principles to the development and management of digital products is not always straightforward. Because technology is dynamic and because we do not want to inhibit innovation, we do not want to prescribe set standards and best practices that could become quickly outdated. Instead, we ask that you answer a series of questions that address specific aspects of creating and managing digital assets. Your answers will be used by IMLS staff and by expert peer reviewers to evaluate your application, and they will be important in determining whether your project will be funded.

Instructions
If you propose to create any type of digital product as part of your project, complete this form. We define digital products very broadly. If you are developing anything through the use of information technology (e.g., digital collections, web resources, metadata, software, or data), you should complete this form.

Please indicate which of the following digital products you will create or collect during your project (Check all that apply):

	Every proposal creating a digital product should complete …	Part I
	If your project will create or collect …	**Then you should complete …**
□	Digital content	Part II
☒	Software (systems, tools, apps, etc.)	Part III
□	Dataset	Part IV

PART I.

A. Intellectual Property Rights and Permissions

We expect applicants to make federally funded work products widely available and usable through strategies such as publishing in open-access journals, depositing works in institutional or discipline-based repositories, and using non-restrictive licenses such as a Creative Commons license.

A.1 What will be the intellectual property status of the content, software, or datasets you intend to create? Who will hold the copyright? Will you assign a Creative Commons license (http://us.creativecommons.org) to the content? If so, which license will it be? If it is software, what open source license will you use (e.g., BSD, GNU, MIT)? Explain and justify your licensing selections.

We will release this code under the Apache 2.0 software license, http://www.apache.org/licenses/LICENSE-2.0. This permissive license is used across the Project Hydra code base. Copyright will be asserted on behalf of Chemical Heritage Foundation and the implementing consulting firm, Data Curation Experts, but royalty-free reuse is permitted.

A.2 What ownership rights will your organization assert over the new digital content, software, or datasets and what conditions will you impose on access and use? Explain any terms of access and conditions of use, why they are justifiable, and how you will notify potential users about relevant terms or conditions.

The Apache 2.0 software license grants "a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form." Redistributions of this work must including the same copyright license. The license will be included in the README documentation file that lives with the code in GitHub.

A.3 Will you create any content or products which may involve privacy concerns, require obtaining permissions or rights, or raise any cultural sensitivities? If so, please describe the issues and how you plan to address them.

No. All available code on which we plan to build is already available through Project Hydra via the same Apache 2.0 software license.

Part II: Projects Creating or Collecting Digital Content

A. **Creating New Digital Content**

A.1 Describe the digital content you will create and/or collect, the quantities of each type, and format you will use.

A.2 List the equipment, software, and supplies that you will use to create the content or the name of the service provider who will perform the work.

A.3 List all the digital file formats (e.g., XML, TIFF, MPEG) you plan to create, along with the relevant information on the appropriate quality standards (e.g., resolution, sampling rate, or pixel dimensions).

B. Digital Workflow and Asset Maintenance/Preservation

B.1 Describe your quality control plan (i.e., how you will monitor and evaluate your workflow and products).

B.2 Describe your plan for preserving and maintaining digital assets during and after the award period of performance (e.g., storage systems, shared repositories, technical documentation, migration planning, commitment of organizational funding for these purposes). Please note: You may charge the Federal award before closeout for the costs of publication or sharing of research results if the costs are not incurred during the period of performance of the Federal award. (See 2 CFR 200.461).

C. Metadata

C.1 Describe how you will produce metadata (e.g., technical, descriptive, administrative, or preservation). Specify which standards you will use for the metadata structure (e.g., MARC, Dublin Core, Encoded Archival Description, PBCore, or PREMIS) and metadata content (e.g., thesauri).

C.2 Explain your strategy for preserving and maintaining metadata created and/or collected during and after the award period of performance.

C.3 Explain what metadata sharing and/or other strategies you will use to facilitate widespread discovery and use of digital content created during your project (e.g., an API (Application Programming Interface), contributions to the Digital Public Library of America (DPLA) or other digital platform, or other support to allow batch queries and retrieval of metadata).

D. **Access and Use**

D.1 Describe how you will make the digital content available to the public. Include details such as the delivery strategy (e.g., openly available online, available to specified audiences) and underlying hardware/software platforms and infrastructure (e.g., specific digital repository software or leased services, accessibility via standard web browsers, requirements for special software tools in order to use the content).

D.2 Provide the name and URL(s) (Uniform Resource Locator) for any examples of previous digital collections or content your organization has created.

Part III. Projects Creating Software (systems, tools, apps, etc.)

A. **General Information**

A.1 Describe the software you intend to create, including a summary of the major functions it will perform and the intended primary audience(s) this software will serve.

Our software will add linked data functionality into Hydra's Sufia, whereby a cataloger can begin typing a name into a metadata field and view matching results from VIAF and FAST authorities. When a name is selected from the autocomplete options, our software will store both the human-readable text and the machine-readable URI. Our software will store the metadata in Fedora 4 and maintain a cache of the external data in a local triple store database. End-users will see only the authority term itself, not the URI. Hydra adopters (present and future) are our target audience, but we expect to reach a diverse audience of digital library and linked data professionals.

A.2 List other existing software that wholly or partially perform the same functions, and explain how the tool or system you will create is different.

Currently there is no Hydra software that performs this function, nor is there a best practice model for how to integrate a triple store database into the Hydra technology stack. Our work will draw on and integrate designs and proofs-of-concept developed by members of Hydra's Applied Linked Data Working Group. Our software will be easy to integrate into an existing Sufia-based Hydra application, and we will work with the Hydra-in-a-Box development team to ensure forward-compatibility with the new turn-key Hydra solution.

B. **Technical Information**

B.1 List the programming languages, platforms, software, or other applications you will use to create your software (systems, tools, apps, etc.) and explain why you chose them.

The basis of this project will be the Hydra stack, specifically the Sufia engine, which will be used as the basis for Hydra-in-a-Box. Sufia is a well-regarded and widely-supported focus of current Hydra development, and Hydra itself is a well-established and respected digital repository framework. This stack includes Ruby, Ruby on Rails, Fedora 4, Solr, Blacklight, and numerous Project Hydra gems. Our software will also require a triple store such as Marmotta or Blazegraph to contain the cache of external metadata.

B.2 Describe how the intended software will extend or interoperate with other existing software.

Our software will extend Sufia, allowing Sufia-based applications to easily begin integrating true Linked Data into their workflows. We expect to use existing Project Hydra gems such as Questioning Authority, which provides a uniform REST API for multiple data sources, and Linked Vocabs, which provides tools for loading external vocabularies. During planning we will continue to watch other work in the hopes of collaborating or integrating with additional projects.

B.3 Describe any underlying additional software or system dependencies necessary to run the new software you will create.

Running this software will require an existing Hydra application (specifically a Sufia application, but possibly also a Curation Concerns application) and all its dependencies.

B.4 Describe the processes you will use for development documentation and for maintaining and updating technical documentation for users of the software.

Documentation will be presented in a README.md file which will live with the code itself on GitHub. This makes documentation maintenance straightforward and convenient. The ReadMe file will include a description of the problem as well as technical notations related to dependencies, installation, and configuration. It will also include usage examples and development information, including how to run tests and how to contribute code back into the project.

B.5 Provide the name and URL(s) for examples of any previous software tools or systems your organization has created.

CHF's Digital Collection (built on Hydra's Sufia). Code: https://github.com/chemheritage/chf-sufia/
[Website itself is limited to IP addresses within CHF's Philadelphia office while we're testing in Beta.]

CHF and DCE's Ansible automation scripts: https://github.com/chemheritage/sufia-ansible

C. Access and Use

C.1 We expect applicants seeking federal funds for software to develop and release these products under an open-source license to maximize access and promote reuse. What ownership rights will your organization assert over the software created, and what conditions will you impose on the access and use of this product? Identify and explain the license under which you will release source code for the software you develop (e.g., BSD, GNU, or MIT software licenses). Explain any prohibitive terms or conditions of use or access, explain why these terms or conditions are justifiable, and explain how you will notify potential users of the software or system.

We will release this code under the Apache 2.0 software license: http://www.apache.org/licenses/LICENSE-2.0. This permissive license is used across the Project Hydra code base. Copyright will be asserted on behalf of Chemical Heritage Foundation and the implementing consulting firm, Data Curation Experts, but royalty-free reuse is permitted. The Apache 2.0 software license grants "a perpetual, worldwide, non-exclusive, no-charge, royalty-free, irrevocable copyright license to reproduce, prepare Derivative Works of, publicly display, publicly perform, sublicense, and distribute the Work and such Derivative Works in Source or Object form." Redistributions of this work must including the same copyright license.

C.2 Describe how you will make the software and source code available to the public and/or its intended users.

The software will be made freely available to the public on GitHub.com. CHF will also promote this code to its target audience of Hydra adopters by posting to Hydra's list serves, engaging with Hydra community working groups, and presenting at the annual Hydra Connect conference, among other outreach avenues.

C.3 Identify where you will be publicly depositing source code for the software developed:

Name of publicly accessible source code repository:
GitHub.com

URL: https://github.com/chemheritage/

Part IV. Projects Creating a Dataset

1. Summarize the intended purpose of this data, the type of data to be collected or generated, the method for collection or generation, the approximate dates or frequency when the data will be generated or collected, and the intended use of the data collected.

2. Does the proposed data collection or research activity require approval by any internal review panel or institutional review board (IRB)? If so, has the proposed research activity been approved? If not, what is your plan for securing approval?

3. Will you collect any personally identifiable information (PII), confidential information (e.g., trade secrets), or proprietary information? If so, detail the specific steps you will take to protect such information while you prepare the data files for public release (e.g., data anonymization, data suppression PII, or synthetic data).

4. If you will collect additional documentation such as consent agreements along with the data, describe plans for preserving the documentation and ensuring that its relationship to the collected data is maintained.

5. What will you use to collect or generate the data? Provide details about any technical requirements or dependencies that would be necessary for understanding, retrieving, displaying, or processing the dataset(s).

6. What documentation (e.g., data documentation, codebooks, etc.) will you capture or create along with the dataset(s)? Where will the documentation be stored, and in what format(s)? How will you permanently associate and manage the documentation with the dataset(s) it describes?

7. What is the plan for archiving, managing, and disseminating data after the completion of the award-funded project?

8. Identify where you will be publicly depositing dataset(s):

Name of repository:
URL:

9. When and how frequently will you review this data management plan? How will the implementation be monitored?