



Implementation Considerations

The Arches data management platform is robust enterprise-level software that organizations can freely download, install, configure, and if desired customize and extend to meet their data management needs.

We characterize Arches as “enterprise-level” to highlight how Arches is designed for deployment in organizational contexts with both **needs** and **capabilities** beyond those typical of an individual person. One practical aspect of enterprise-level software is that it typically needs to be hosted on a server or on cloud-based storage. This documentation helps to explain the types of capabilities an organization needs to successfully deploy and maintain Arches.

Overview

Organizations need to understand, for evaluation and planning purposes, what might be required to implement Arches according to their own requirements. The basic process for implementing Arches typically includes the following steps:

- Determine whether Arches is the right fit for your purposes, and whether you have access to the necessary technical expertise (in-house or externally) to implement Arches based on the steps below.
- Determine how you will host Arches (either on the cloud or your own servers).
- Install the Arches software and its dependencies.
- Decide how to model (i.e., determine content of and organize) your data. Modeling and organizing information is at the heart of every Arches implementation. It’s typically good practice to build upon wider community expertise and start by loading ontologies and controlled vocabularies into your Arches deployment. Rather than starting from scratch, Arches enables you to reuse and adapt existing resource models defined by others in the community. If those don’t work, the

“Arches Designer” provides user interfaces where you can define your own ways to organize your data without needing a programmer.

- While developing models to organize your data, consider the issues involved in migrating your legacy information into your new Arches deployment. How will any legacy data impact your modeling needs? You may want to experiment to get a better understanding of time and effort needed to clean and reorganize legacy data for import into Arches.
- Arches is a highly capable and flexible system. However, you may need additional features or integrations with other information systems requiring additional software development. Such extensions should be carefully planned and architected to be sustainable as your organization upgrades to future versions of Arches.
- Create a strategy for ongoing support and maintenance of your Arches implementation and participation in the larger Arches community.

The remainder of this document expands on the above steps and is organized into five main sections. Each section describes the considerations and the recommended technical skills needed to be successful.:

- Recommended Technical Capabilities
- Installation Considerations
- Data Considerations
- Ongoing Support and Community Participation
- More Detailed Technical Considerations

Recommended Technical Capabilities

To help guide your resource planning, here is an overview of the minimum required technical capabilities needed to deploy Arches:

- **Web Hosting and IT Systems Administration:** At a minimum, deploying Arches requires fluency with Web hosting, IT systems administration, and, if using a cloud service provider, cloud computing infrastructure. These skills are needed to install, configure, and (critically) maintain Arches. We also strongly recommend familiarity with

deploying Django/Python applications. See below for additional technical expertise required for more complex and/or customized deployments.

- **Information and Data Management:** Arches provides powerful capabilities in cultural heritage data management. However, to be used effectively, Arches needs to be well integrated into wider planning and workflows that comprise an organization's data management strategy. What workflows or insights need to be supported by your Arches deployment and how should these be reflected in data modeling? How will legacy data get mapped and migrated into Arches? What is the backup strategy for Arches? What are long term data preservation needs beyond the current Arches implementation? How should security, privacy, and confidentiality needs shape Arches data models and user permissions? What other systems need to interface with data in Arches? Addressing all of these questions requires certain expertise in information management.

Installation Considerations

Arches will need to reside on a server, either on an in-house server or on a cloud hosting service, or perhaps even a combination of the two. If this is not the type of software installation that you or your team are familiar with, please consider the following points:

Institutional hosting requirements and rules

Begin by establishing if your institution has rules you'll need to follow when hosting databases and websites. In some cases, an in-house server is the only option, and in others, no such option will exist. This is important to consider up front, as it will impact the overall cost of the project.

Which cloud hosting service will work best for Arches?

If you are going to use a cloud hosting service there are many good options—Google, Microsoft and Amazon all offer cloud hosting services, as well as smaller companies like DigitalOcean. Any of these will work for Arches. However, at this point, the most extensive tests and deployments have been done using Amazon Web Services (AWS). AWS is popular, well-documented, and widely understood by developers, but many other cloud service providers have very similar capabilities and costs.

Technical specifications

The [Arches documentation](#) recommends at least 4GB of RAM for evaluation and testing (8-32GB for production) and 2GB minimum disk space to install the code base. However, required disk space depends on the size and type of the data you'll be storing. Do you have a lot

of photos or videos? These types of files will use much more disk space than simple database records, for example. Such storage needs may also motivate your organization to explore cloud-storage services such as Amazon's S3 service.

Which operating system (OS) will work best for Arches?

Arches works on Linux, Windows, and Mac servers. However, we recommend deployment on Linux (especially Ubuntu) because it simplifies installation of Arches and the software dependencies used by Arches. A mapping library used by the tileserver included in Arches is not compatible with Windows (meaning you can't serve GeoTIFFs, for example, but otherwise everything else should work the same).

Data Considerations

Arches was designed to be flexible enough to accommodate data of many different types and formats. As such, implementing organizations have the flexibility to decide what data they want to manage with Arches and how that data is organized and accessed. The following are important points to consider when planning how to configure Arches based on your data requirements:

Legacy data

Most organizations that are implementing Arches will have legacy (existing) datasets that will be migrated into Arches. Legacy data types that can be imported into Arches include spatial data, tabular data, and any kind of digital file (e.g., PDFs, images, videos, sound recordings). What's important to note is that inspecting legacy data can help to determine if previous patterns of data collection and usage are going to be continued using Arches, or if changes will be made to your organization's data creation and management methodology. In any event, the first step is to make a thorough review of the legacy data you wish to migrate, including the identification of data fields in each dataset.

Data structure and organization

By examining and determining what legacy data you want to import into Arches, it may become clear that new data fields and reorganization of the existing data are needed to serve your institutional goals. Remember, migrating to a new data management system is the best time to update and improve on old procedures! Once your organization has determined what data and processes you need to manage with Arches, you can start to dynamically define your Arches database and data entry forms using the [Arches Designer](#). Arches allows you to organize the data for each resource type in a Resource Model, which is a data model based on a graph structure. How you organize the data in a Resource Model determines the data entry forms and the contents of reports, as well as how the data can be searched. Also note that, by default,

Arches uses the CIDOC Conceptual Reference Model (CRM), an ISO standard for cultural heritage information, as the semantic ontology for each Resource Model.

Controlled vocabularies

As part of your overall data strategy you may want to consider how to leverage the use of controlled vocabularies to enforce consistent use of the proper terminology during data entry, while also facilitating more accurate search results. Arches manages controlled vocabularies via its [Reference Data Manager \(RDM\)](#). In order to take full advantage of the RDM, which works in tandem with the Arches Designer, you may consider investing some time to create controlled vocabularies for data entry fields. We also encourage you to **benefit from the contributions of the wider community** by importing and using existing controlled vocabularies.

Migrating data

Arches has several tools to facilitate the bulk import of legacy data. The Arches user interface has powerful features to import data stored in CSV and Excel file formats. There are other command-line tools to import legacy CSV, JSON, or Shapefile data. In addition, Arches administrators proficient in SQL can map and import data by interacting with the Arches PostgreSQL database.

The best method to import data into Arches depends on the scale, complexity, and type of legacy data you have. If your data isn't particularly complex, then it may be easiest to convert your existing files to CSV. And if you are importing spatial data, you may either convert your spatial information to WKT (for import through CSV) or use a Shapefile to import the data into Arches. As with any data migration process, reformatting and cleaning your data will likely be an involved undertaking, so be sure to plan accordingly. See the above discussion of Data Management Expertise to understand some of the background required for migrating data into Arches.

Ongoing Support and Community Participation

As you are installing, configuring, and potentially customizing your Arches instance, you may come to the realization that the Arches platform is meant to support your ongoing work and the data that it produces. The following are some considerations to help you plan for the future of your Arches implementation:

Ongoing maintenance and server administration

It will be necessary to establish a system administrator, potentially train that administrator as needed, and determine who will provide ongoing technical support. Ongoing support will consist of basic server updates, as well as Arches upgrades and enhancements.

Data updates

In order to ensure that the data in your Arches instance is valid and authoritative, it is essential that a strategy for ongoing data updates is established. If new data will be entered using the Arches data entry interface or through bulk loading methods, then new or existing staff must be trained in their use and training materials may need to be created. If data will be updated through a linkage with another system, then you may need to plan for a customization that enables that.

Community Participation

The Arches open source community has many opportunities for involvement by Arches implementers, and these include, but are not limited to: sharing Arches Resource Models and packages; contributing new code as the result of customizing and extending Arches; helping other Arches implementers via the Discussion Forum and other channels; writing articles about your Arches experience or helping to improve Arches documentation; and taking part in the overall governance of the community and helping to determine the general developmental direction of the software. Being part of the Arches community ensures that you are up-to-date on the latest Arches news and developments, which help you to best maintain your Arches instance. Participating in the Arches community helps you to best use and shape the technology to meet your needs.

If you have questions or feedback regarding implementation considerations, please post on the [Arches Community Forum](#).

More Detailed Technical Considerations

Below we provide additional more detailed and technical discussion of Arches implementation considerations. These details should help guide decision makers in understanding the specific areas of expertise required to implement Arches for an organization.

Data Management Expertise Needed for Effective Use

One of the most important capabilities of Arches is that it puts data modeling and data organization into the hands of subject matter experts without requiring custom programming. The Arches user interface (“Arches Designer”) enables end users to quickly define and update data structures and schemas to meet an organization’s needs. This capability both reduces costs and allows for more experimentation and iteration in organizing information. To make best use of these features, we recommend familiarity with the following aspects of data management:

- **Controlled Vocabularies:** The selection and use of controlled vocabularies is a key aspect in making information classification appropriate and consistent.
- **Ontologies:** Ontologies, especially the CIDOC-CRM, provide a formally defined bedrock of concepts that are used to organize different kinds of entities (“Resource Models” in Arches) and how they relate.
- **Data Types:** Data modeling involves defining different kinds of attributes that may include dates, numbers, controlled vocabulary lists, spatial geometries, free-text strings, etc. Familiarity with the use of these different data types is a fundamental aspect of data modeling.
- **Data Formats:** Data are often expressed in tabular formats like CSV or Excel, in nested “tree” structures as in JSON, specialized geospatial formats like GEOJSON and Shapefiles, or in graph structures like different RDF formats. Familiarity with these different data can be useful to manage imports and exports of data from Arches.

Technical Expertise Needed for Customization and/or Large Data Scales

The core Arches application comes ready and bundled with a wide variety of powerful tools and features. However, should your organization need to customize Arches, there are certain areas of technical expertise that you will need.

Technical Expertise Required for Backend (server-side) Customization

- **Python Django:** Arches is built with the Django framework in the Python programming language. The open source ecosystem built around Django is large and very active, and offers many opportunities to extend the capabilities of Arches.

- **PostgreSQL/PostGIS:** Arches uses PostgreSQL as a backed database. While knowledge of SQL is by no means required to use Arches, it is helpful in certain scenarios where you need additional flexibility and precision in importing large sets of legacy data.
- **ElasticSearch:** Arches uses ElasticSearch to power fast and flexible querying of data. Knowledge of ElasticSearch and its configuration is useful in deployment scenarios involving some combination of: large scale, complex data modeling, and fast performance needs.
- **Cloud Computing:** Organizations often deploy Arches via commercial cloud computing services. Using these services, the Arches application, PostgreSQL database, and ElasticSearch components can be deployed on multiple instances to meet scale and performance expectations. However, such cloud deployments require technical experience with the specific cloud computing service provider used.
- **REST APIs:** Organizations seeking to integrate Arches with other information systems will likely need to tap into developers with experience using RESTful Web-Services (APIs) such as those supported by Arches.

Technical Expertise Required for Frontend (client-side, user interface) Customization

- **Javascript:** Arches supports a great deal of flexible customization of the “frontend” user-interface through a variety of widget and templating options. To customize the frontend, a developer will need to be proficient in Javascript and certain important Javascript libraries, including: Knockout (soon to be retired and replaced by Vue.js), RequireJS, and Mapbox GL JS.
- **HTML/CSS and Templating:** Knowledge of HTML and CSS web design and good practices is also required for customizing the frontend of Arches.

If your IT support doesn't have the skills and resources available to install, configure, customize, and/or maintain Arches, you may want to contract with a service provider experienced with implementing Arches. A listing of recognized Arches service providers is available on the Arches web site: <https://www.archesproject.org/service-providers>. It is important to note that some service providers offer Arches technical training and knowledge transfer, which can be an option for organizations with IT staff.

Configuration Considerations

Once you've installed Arches on a server and determined your organization's overall requirements, you can begin to configure Arches. This document defines configuration as any

activity implementers take to set up Arches according to their own needs without changing the core Arches software code. Customization of the Arches code to serve your specific use case is covered in the Customizing and Extending Arches section.

Localized settings and content

There are a number of settings and some content that will need to be localized. For example: home page content and branding, including the name you are giving the system and your organization's logo, saved searches, default map extent coordinates, additional map overlays, configuration of your basemaps including historic maps or satellite imagery. You may also be using a non-English language for your installation's UI—Arches uses Transifex to manage its ever-growing number of translations, so you can set your deployment to use one or more languages for which a translation is available. Changes to the settings or content can generally be made at any time during the configuration process, depending on how the data is organized.

Configuring Arches to handle your data

Arches allows you to design your database with no coding using interface tools, such as the [Arches Designer](#) and [Reference Data Manager](#). These tools enable administrators to either edit or create new Resource Models, with predefined data fields, data entry forms and reports, and permissions settings, along with the accompanying controlled vocabularies. You may use existing Resource Models, controlled vocabularies and project settings from an Arches implementation that has similar requirements or you may choose to create new Resource Models and controlled vocabularies from scratch, which would require some expertise in database design, semantic modeling, and vocabulary creation. The Arches community is currently assembling a **library of** Arches implementations that are making their Resource Models, controlled vocabularies and/or project settings available for reuse and adaptation by other organizations in the Arches community.

Users and Permissions

Arches allows you to manage individual users and user groups, and to define how both users and groups as well as the general public can interact with your data through permissions settings. For example, based on your organization's data access policies, you may decide that only certain staff members can edit geospatial data for specific types of resources or that your Arches administrator is the only person who can modify your controlled vocabularies. You may add new users and groups and modify permissions at any time, but you may want to create an initial plan regarding who has access to view, create, edit, and delete what types of data. This information will help you to fully define your Resource Model permissions settings using the Arches Designer.

Recommended skills/knowledge: The latest version of Arches allows you to do many of your configuration tasks using the Arches user interface. These include: establishing and applying

the name of your Arches instance; defining your Google Analytics key (if this tool aligns with your privacy policies), changing basic search and map settings; adding saved searches; styling map layers (e.g., color of icons, transparency of overlays); creating and editing data fields, data entry forms, reports, and permissions settings using the Arches Designer; and defining and adding new terminology using the Reference Data Manager. In addition, using the Arches Django administration panel, you can also manage users and add the map layers to be managed within the user interface. For these tasks, no software coding is required, and apart from learning the Arches interface, you will need some background knowledge on the concepts underlying each task. For example, the Reference Data Manager allows you to easily create controlled vocabularies, but you will need some background knowledge on best practices to do correctly.

Configuration tasks that require additional IT skills include: changing the Arches front-end —i.e. how Arches appears to the end user, adding custom map layers, and changing or adding to the languages that the interface displays. See the above discussion about Frontend technical expertise to review necessary skills.

Customizing and Extending Arches

The Arches platform source code is open, malleable and extensible, so your imagination is the limit of how your installation can be customized to support your needs. Here are some examples of customizing and extending Arches:

- Creating a unique user interface;
- Setting up e-mail reporting so admins get a daily summary of database activity;
- Integrating Arches with another system so that resources are synchronized between the two databases;
- Setting up the login page as the first thing that a user encounters;
- Incorporating 3D models or other types of viewers into resource reports;
- Implementing a draft → publish workflow for resource creation;
- Adding the ability to geocode addresses to create spatial coordinates.

Enhancements such as these can be seamlessly integrated into your Arches implementation. We only ask that you share your improvements with the rest of the Arches community if they are more broadly applicable! See the next section for more information on community participation. See the above discussion about Backend and Frontend technical expertise to review necessary skills for enhancing Arches.