Arches-HIP Documentation

Release 1.0

Farallon Geographics, Inc.

November 12, 2015

Contents

Contents:

Overview

1.1 What is Arches-HIP?

Arches-HIP (Heritage Inventory Package) is a web application designed to make it easy to create and maintain an inventory of cultural heritage resources at a local, regional, or national scale.

Arches-HIP supports several cultural heritage data types; these data types are called "Resources". Resources allow you to create data records and describe important, culturally signficant objects, activities, events, and people.

Arches-HIP supports the following resource types:

- **Heritage Resources:** These are culturally significant objects such as buildings and monuments. However, Heritage Resources may include any object that an organization regards as important. Examples include air raid sirens, statues, canals, roads, gates, bridges, etc.

- **Heritage Resource Groups** These are collections of heritage resources that are best regarded as a group. In Europe, such resources are often referred to as ensembles. In the United States, districts or landscapes may be inventoried as Heritage Resource Groups.

- **Activity** In Arches-HIP, activities are resources that occur over a period of time. Examples include surveys, investigations, excavations, literature review, and other actions.

- **Historic Event** These are culturally significant occurrences. Examples include the 1906 San Francisco Earthquake, Coronation of Napoleon, and the Normandy Beach landings in WWII.

- **Actor** In Arches-HIP, actor resources are persons, groups of people, or organizations.

- **Information Objects** These are objects that encode information such as an image (either digital or a traditional photograph), documents, signs, inscriptions, or 33 1/3 rpm records.

A key design consideration for the Arches-HIP web application is to allow you to quickly create and easily manage cultural heritage data. To achieve this goal, Arches-HIP allows you to create relationships between resource types to more precisely describe the significance of cultural heritage objects. For example, in Arches-HIP you can:

- Create an Actor Resource, such as "Frank Lloyd Wright"

- Create a Heritage Resource, such as "Hollyhock House"

- Link the resources in a way that describes their relationship: "Frank Lloyd Wright" "architected" "Hollyhock House"

Arches-HIP's ability to track relationships between resources is a key capability that can support a richer cultural heritage inventory and more intuitive discovery of heritage records.

Arches-HIP uses the Reference Data Manager (RDM), a thesaurus management model included in the Arches web framework, to let you manage the terms that you will need to describe heritage resources and the relationships between resources (e.g.: "architected").

Version History

> v1.0 April, 2015: Initial Release

License

> Arches-HIP is free software and is licensed under the terms of the GNU Affero General Public License (http://www.gnu.org/licenses/agpl-3.0.html)

1.2 Who is Arches-HIP for?

Arches-HIP is primarily intended for organizations that are responsible for creating cultural heritage inventories as an essential step in the preservation and conservation of heritage.

Getting Started

2.1 Introduction

Welcome to the **Arches HIP Installation Guide**. This document is intended to help you quickly get HIP running on your server.

2.1.1 What do I need to know to use this guide?

Arches HIP is designed to let you manage many types of cultural heritage data. And while we've tried to make it easy to install HIP, you'll need to make some decisions on how you wish to describe and manage the information that goes into Arches HIP. But don't worry, we've also included lots of examples and well-thought-out defaults for setting up the software.

So, installing HIP means loading software and deciding how you want to describe the cultural heritage information that's important to you. We've written this guide assuming that you are comfortable working with your computer's command line, and that you can edit text files with ease.

Arches HIP is built on the open source Arches framework. The Arches documentation can be found here: http://arches3.readthedocs.org/en/latest/ This document may be useful for understanding Arches-HIP and will be referenced throughout the Arches HIP documentation.

That's it! Feel free to tap a colleague on the shoulder if you need a little help along the way, or check out the resources at http://www.archesproject.org for more support.

2.2 System Requirements

There are a few things to know about Arches HIP before we get started.

Please note that HIP has been developed for modern browsers. It supports:

- Firefox
- Chrome
- Safari
- Opera
- Internet Explorer 10 or higher.

Minimum system requirements:

- At least 4GB of RAM for evaluation and testing, or 8 to 16GB for production.

- 10GB minimum to install the code base and test dataset, but disk space requirements will vary greatly depending on the size of your dataset

It's important to recognize that Arches HIP is designed to manage large, enterprise-scale Cultural Heritage inventories. The software is sophisticated and is designed to be support extensive customization. This means that there are many ways to deploy Arches HIP.

We've written this guide to streamline the installation of Arches Server and the in the more straight-forward manner possible. These installation instructions provide step-by-step guidelines for installing Arches HIP.

2.3 Dependencies

Arches requires the following applications/libraries:

- PostgreSQL relational database (version 9.3) (http://www.postgresql.org/download/)

- PostGIS (version 2.x) spatial module for PostgreSQL (http://postgis.net/install)

- Python (version 2.7.6 - there seem to be issues with later versions of python) (https://www.python.org/downloads/)

- GEOS (http://trac.osgeo.org/geos/)

You'll need to install each of these components on your computer before you can install Arches. Luckily, each of these technologies provide comprehensive installation instructions.

Note For the installation process you will need **pip** installed on your system. If you don't already have it, you can find instructions to install it here: https://pip.pypa.io/en/latest/installing.html

2.4 Installating Arches-HIP

These installation instructions will guide you through the installation and configuration proceess for the Arches-HIP application. When we're done, you'll have the following folders on your system:

```
/Projects
  /ENV
  /my_hip_app
```

The **/ENV** folder is where the Arches framework and the Arches-HIP applications will be installed

The **/my_hip_app** folder is a folder that will hold your custom settings for the HIP.

Custom settings include such things as the text and logos on the HIP home page, your map settings, and other information needed to ensure that the HIP meets your specific requirements.

With your Arches dependencies installed, you are now ready to install HIP.

#1. Create an Arches folder:

Create a folder called 'Projects' (or any other name that you prefer) on your computer.

#2. Install virtualenv:

Open a command prompt and type:

```
$ pip install virtualenv
```

Note: For Linux users: Using apt-get to install virtualenv will install an older version that will result in errors in your Arches installation. Be sure to use pip for your virtualenv installation.

---

virtualenv creates a directory with it's own installation of Python and Python executables.

#3. Create a virtual environment:

Arches and Arches-HIP will be installed within a virtual environment. To do so, navigate to your Projects directory (or wherever you created your Arches root folder) and create your virtual environment with the following command:

```
$ virtualenv ENV
```

Now activate your virtual environment with the following command:

On Linux (and other POSIX systems):

```
$ source ENV/bin/activate
```

On Windows:

```
> \path to 'Projects'\ENV\Scripts\activate
```

#4. Install the arches_hip module:

For this step your virtual environment must be activated. You will know that's the case if you see the name of your virtual environment in parentheses proceeding your command prompt like so (ENV):

```
(ENV)$ pip install arches_hip
```

#5. Create the folder for your HIP customizations:

Navigate to the **Projects** directory run the following command replacing 'my_hip_app' with the name you want to use for your version of Arches-HIP. For example, the City of Los Angeles customization of the HIP is called 'HistoricPlacesLA'

```
(ENV)$ arches-app create my_hip_app --app arches_hip
```

Note: Windows users may get an error like: **'arches-app' is not recognized as an internal or external command, operable program or batch file.** If that occurs, you may need to specify the path to the arches-app command. Be sure to add **'python'** to the beginning of the command to ensure that you use your virtual environment's Python. For example:

```
(ENV)> python path\to\ENV\Scripts\arches-app create my_hip_app --app arches_hip
```

#6. Windows users need to explicity point their application to a valid gdal dll file:

Open the `settings.py` file in `my_hip_app/my_hip_app` and add the following setting

GDAL_LIBRARY_PATH = '{path to OSGeo4W bin directory}/gdalxxxx.dll'

For example, your setting might look like this:

GDAL_LIBRARY_PATH = 'C:/OSGeo4W64/bin/gdal111.dll'

#7. Install ElasticSearch:

Navigate to the directory containing `manage.py`:

```
(ENV)$ cd my_hip_app/
(ENV)$ python manage.py packages -o setup_elasticsearch
```

Your project directory should now look something like this:

```
/Projects
  /ENV (virtual environment where arches_hip gets installed)
  /my_hip_app
    manage.py
    wsgi.py
    README.txt
    /my_hip_app
        settings.py
        setup.py
        urls.py
        /elasticsearch
        /logs
        /media
        /models
        /source_data
        /templates
        /tests
        /views
```

2.5 Settings

Open the `settings.py` file in `my_hip_app/my_hip_app`

There are a few changes that you may need to make in order for your application to install and run properly.

Database Settings

Be sure that the name of your postgis template in your settings matches your postgis template in postgres. If it doesn't, uncomment the line (remove the # at the start of each line) below and replace 'template_postgis_20' with the name of your template:

```
#DATABASES['default']['POSTGIS_TEMPLATE'] = 'template_postgis_20'
```

The default password used in HIP is 'postgis'. You may have already set your Postgres password to something more secure. If so, uncomment the line below and replace 'postgis' with your Postgres connection password:

```
#DATABASES['default']['PASSWORD'] = 'postgis'
```

Map Settings

The map center, zoom and extent will likely need to be updated to reflect your study region:

```
#DEFAULT_MAP_X = 0
#DEFAULT_MAP_Y = 0
#DEFAULT_MAP_ZOOM = 0
#MAP_MIN_ZOOM = 0
#MAP_MAX_ZOOM = 19
#MAP_EXTENT = '-13228037.69691764,3981296.0184014924,-13123624.71628009,4080358.407059081'
```

To use these settings, you will need to 'uncomment' them, and then change their values to meet your needs.

`DEFAULT_MAP_X` and `DEFAULT_MAP_Y` are the x and y (respectively) coordinate values which all maps will be centered on by default. `MAP_EXTENT` is a string 'bounding box' [min x, min y, max x, max y] representing the extent to which all maps will be constrained.

All of these coordinates are expressed in the web standard Spherical Mercator projection (http://spatialreference.org/ref/sr-org/6864/). A simple converter tool to go between Lat/Long and Spherical Mercator coordinates can be found here: http://mal2.ch/stuff/latlontomercator.html

`DEFAULT_MAP_ZOOM` is the default zoom level that all maps will start on. `MAP_MIN_ZOOM` and `MAP_MAX_ZOOM` are the minimum and maximum zoom levels that all maps will allow a user to access. More on zoom levels can be found here: http://openlayers.org/en/v3.3.0/doc/tutorials/concepts.html

Basemap and Geocoding Settings

By default, Arches-HIP will use Bing services for basemaps and geocoding; if you want to use these, you will need to get a Bing Maps Key (https://msdn.microsoft.com/en-us/library/ff428642.aspx) and add it to your `settings.py` file like so:

```
BING_KEY = 'my-bing-key'
```

2.6 Running Arches-HIP

1. **Start ElasticSearch**:

   Open a new command terminal and run the following command:

   ```
   $ my_hip_app\my_hip_app\elasticsearch\elasticsearch-1.4.1\bin\elasticsearch
   ```

   You can minimize this terminal and let it run in the background.

   Note: If you'd like to run ElasticSearch as a daemon process, just add the argument `-d` to the end of the command. You can check to make sure the process is running correctly by using:

   ```
   (ENV)$ curl localhost:9200
   ```

   where 9200 is the default port for ElasticSearch. You'll get a JSON response containing the phrase "You Know, for Search". To shut down the process, use this command:

   ```
   (ENV)$ curl -XPOST localhost:9200/_shutdown
   ```

   ---

2. **Create your database**:

   Using **your previous terminal**, run:

   ```
   (ENV)$ python manage.py packages -o install
   ```

   Note: This step will drop and recreate a database with the name "arches_my_hip_app", as well as load any concept schemes or business data that you may have defined in your settings.

   Note: If you get the following error *'django.db.utils.OperationalError: FATAL: database "arches_my_hip_app" does not exist'* It is probably because Arches is unable to find your postgis database template to create your new application's database. By default, Arches will look for a template called 'template_postgis_20'. If this does not match the name of your postgis template database, open your new applications `settings.py` file and define the name of your postgis template:

   ```
   DATABASES['default']['POSTGIS_TEMPLATE'] = 'the name of your postgis template'
   ```

   Manage Data Using pgAdmin:

Once you have successfully created your database(s), it is important to manage them effectively. You can do this by using pgAdmin, an open source and graphical database management interface for PostegreSQL. It is easy to use, supports all PosetgreSQL features, contains effective server-side code and SQL syntax editing tools, and can be used on several different platforms including: Linux, Windows, Max OSX, Solaris, and FreeBSD.

3. **Run arches**:

   Open a new terminal. **Be sure to activate your virtual environment for this terminal as well**.

   Navigate to your application's manage.py file and run:

   ```
   (ENV)$ python manage.py runserver
   ```

   Now you can open a browser window and see your application running at:

   ```
   http://localhost:8000/
   ```

Next steps:

   Now that Arches is running, you will need to load your concept scheme and resource data. You can get started on that here: Loading Data

2.7 Before you go live...

We use the Django development server to make it easy for you to confirm that Arches has installed properly.

You WILL NEED to configure Arches to use a production quality web server such as Apache if you want to use Arches in production. See the Django documentation for more on using Apache and mod_wsgi: https://docs.djangoproject.com/en/1.6/howto/deployment/wsgi/modwsgi/

Some other things you'll want to make sure that you do:

   • change the admin user name and password

   • Disallow access to port 9200 on prod machine

   • Set DEBUG = False on prod machine

   • Make sure that your Elasticsearch instance is production ready: http://www.elastic.co/guide/en/elasticsearch/guide/master/deploy.h

Arches-HIP Data Resources

Arches-HIP is designed to allow you to create cultural heritage inventories that consist of 6 primary resource types:

- **Heritage Resources:** These are culturally significant objects such as buildings and monuments. However, Heritage Resources may be any object that an organization regards as important. Examples include air raid sirens, statues, canals, roads, gates, bridges, etc.

- **Heritage Resource Groups** These are collections of heritage resources that are best regarded as a group. In Europe, such resources are often referred to as ensembles. In the United States, districts or landscapes may be inventoried as Heritage Resource Groups.

- **Activity** In Arches-HIP, activities are resources that occur over a period of time. Examples include surveys, investigations, excavations, literature review, and other actions.

- **Historic Event** These are culturally significant occurrences. Examples include the 1906 San Francisco Earthquake, Coronation of Napoleon, and the Normandy Beach landings in WWII.

- **Actor** In Arches-HIP, actor resources are persons, groups of people, or organizations.

- **Information Objects** These are objects that encode information such as an image (either digital or a traditional photograph), documents, signs, inscriptions, or 33 1/3 rpm records.

Each HIP resource provides a rich list of attributes that can be used to describe a cultural heritage feature in detail. The set of attributes that make up a resource is defined using a Graph.

Arches-HIP implements one Graph for each resource type. A key capability of Arches-HIP is that each Graph implements the CIDOC CRM (http://www.cidoc-crm.org/) ontology to impart meaning to each attribute in a resource graph.

Note: Certain heritage objects like an archaeological site can be modeled as either a resource or a group depending on your preference or implementation decision.

3.1 HIP Graphs

You can get a sense of the type of information associated with each resource by looking at each graph (details about the graphs attributes can be found in Appendix A):

To learn more about resource graphs check out the general Arches Documentation . For a closer and more in-depth look at each of these graphs, click on their title links.

3.1.1 Heritage Resource.E18
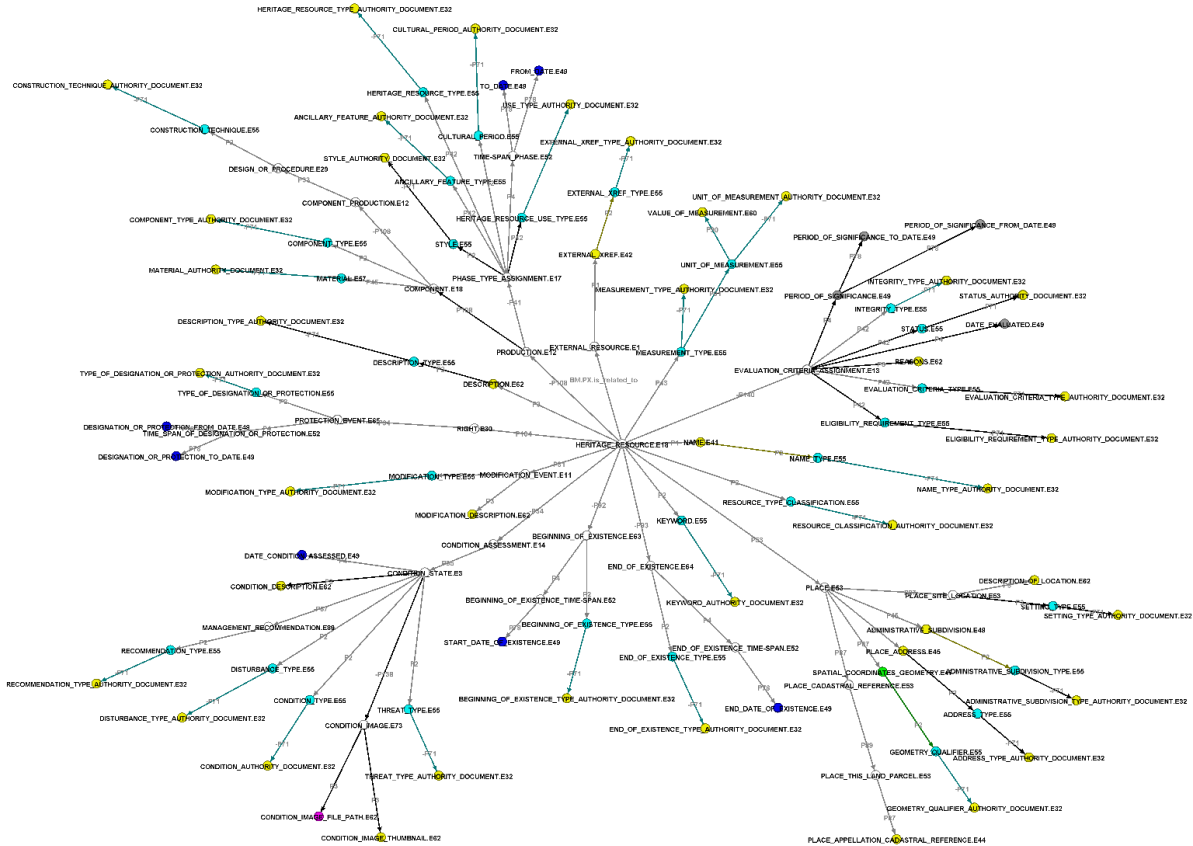
3.1.2 Heritage Resource Group.E27

"Place". One way to describe place is with an address. Another way to describe place is with spatial coordinates. Arches-HIP provides data entry screens that make it easy to enter these data.

In many cases, Arches-HIP assumes that you will want to limit the values for an attribute to a list of approved terms. For example, you will likely want to limit the terms used to describe the condition of a resource to values such as "good", "fair", "poor". Arches-HIP allows you to define the exact terms you wish to use for these kinds of attributes.

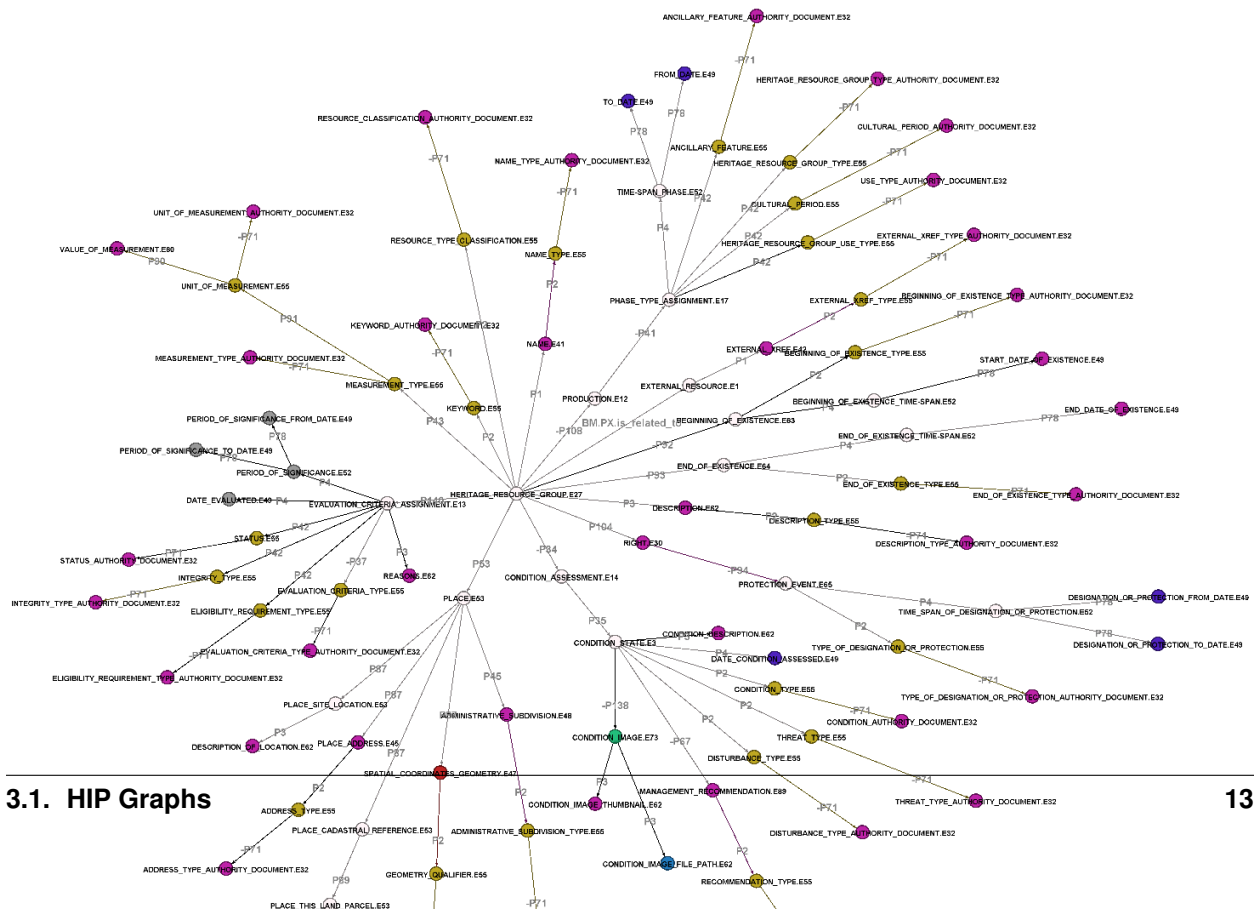Arches-HIP includes an "Authority File" for every attribute that is constrained by a controlled vocabulary. Note that a "controlled vocabulary" is nothing more than a listing of terms that will be used to describe an attribute of a resource. HIP gives you a lot of flexibility in defining your controlled vocabularies. They may be simple word lists (like the example for describing the condition of a resource) or a full-flegded thesaurus.

3.2 Included Authority Files

HIP comes with the following authority files included. It is also important to note that one entity type can apply to multiple resource graphs.

Entity Type	Authority Document
ACTIVITY_TYPE.E55	ACTIVITY_TYPE_AUTHORITY_DOCUMENT.csv
ACTOR_TYPE.E55	ACTOR_TYPE_AUTHORITY_DOCUMENT.csv
ADDRESS_TYPE.E55	ADDRESS_TYPE_AUTHORITY_DOCUMENT.csv
ADMINISTRATIVE_SUBDIVISION_TYPE.E55	ADMINISTRATIVE_SUBDIVISION_TYPE_AUTHORITY_DOCUMENT
BEGINNING_OF_EXISTENCE_TYPE.E55	BEGINNING_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.csv
COMPONENT_TYPE.E55	COMPONENT_TYPE_AUTHORITY_DOCUMENT.csv
CONDITION_TYPE.E55	CONDITION_AUTHORITY_DOCUMENT.csv
CONSTRUCTION_TECHNIQUE.E55	CONSTRUCTION_TECHNIQUE_AUTHORITY_DOCUMENT.csv
CULTURAL_PERIOD.E55	CULTURAL_PERIOD_AUTHORITY_DOCUMENT.csv
DESCRIPTION_TYPE.E55	DESCRIPTION_TYPE_AUTHORITY_DOCUMENT.csv
DISTURBANCE_TYPE.E55	DISTURBANCE_TYPE_AUTHORITY_DOCUMENT.csv
ELIGIBILITY_REQUIREMENT_TYPE.E55	ELIGIBILITY_REQUIREMENT_TYPE_AUTHORITY DOCUMENT.csv
END_OF_EXISTENCE_TYPE.E55	END_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.csv
EVALUATION_CRITERIA_TYPE.E55	EVALUATION_CRITERIA_TYPE_AUTHORITY_DOCUMENT.csv
EXTERNAL_XREF_TYPE.E55	EXTERNAL_XREF_TYPE_AUTHORITY_DOCUMENT.csv
GEOMETRY_QUALIFIER.E55	GEOMETRY_QUALIFIER_AUTHORITY_DOCUMENT.csv
HERITAGE_RESOURCE_GROUP_TYPE.E55	HERITAGE_RESOURCE_GROUP_TYPE_AUTHORITY_DOCUMENT.c
HERITAGE_RESOURCE_TYPE.E55	HERITAGE_RESOURCE_TYPE_AUTHORITY_DOCUMENT.csv
HISTORICAL_EVENT_TYPE.E55	HISTORICAL_EVENT_TYPE_AUTHORITY_DOCUMENT.csv
IDENTIFIER_TYPE.E55	IDENTIFIER_TYPE_AUTHORITY_DOCUMENT.csv
INFORMATION_CARRIER_FORMAT_TYPE.E55	INFORMATION_CARRIER_FORMAT_AUTHORITY_DOCUMENT.csv
INFORMATION_RESOURCE_TYPE.E55	INFORMATION_RESOURCE_TYPE_AUTHORITY_DOCUMENT.csv
INTEGRITY_TYPE.E55	INTEGRITY_TYPE_AUTHORITY_DOCUMENT.csv
KEYWORD.E55	KEYWORD_AUTHORITY_DOCUMENT.csv
LANGUAGE.E55	LANGUAGE_AUTHORITY_DOCUMENT.csv
MATERIAL.E57	MATERIAL_AUTHORITY_DOCUMENT.csv
MEASUREMENT_TYPE.E55	MEASUREMENT_TYPE_AUTHORITY_DOCUMENT.csv
MODIFICATION_TYPE.E55	MODIFICATION_TYPE_AUTHORITY_DOCUMENT.csv
NAME_TYPE.E55	NAME_TYPE_AUTHORITY_DOCUMENT.csv
RECOMMENDATION_TYPE.E55	RECOMMENDATION_TYPE_AUTHORITY_DOCUMENT.csv
ANCILLARY_FEATURE_TYPE.E55	RELATED_FEATURE_AUTHORITY_DOCUMENT.csv
RESOURCE_TYPE_CLASSIFICATION.E55	RESOURCE_CLASSIFICATION_AUTHORITY_DOCUMENT.csv
RIGHT_TYPE.E55	RIGHT_TYPE_AUTHORITY_DOCUMENT.csv

Table 3.1 – continued from previous page

Entity Type	Authority Document
SETTING_TYPE.E55	SETTING_TYPE_AUTHORITY_DOCUMENT.csv
STATUS.E55	STATUS_AUTHORITY_DOCUMENT.csv
STYLE.E55	STYLE_AUTHORITY_DOCUMENT.csv
THREAT_TYPE.E55	THREAT_TYPE_AUTHORITY_DOCUMENT.csv
TITLE_TYPE.E55	TITLE_TYPE_AUTHORITY_DOCUMENT.csv
TYPE_OF_DESIGNATION_OR_PROTECTION.E55	TYPE_OF_DESIGNATION_OR_PROTECTION_AUTHORITY_DOCUM
UNIT_OF_MEASUREMENT.E55	UNIT_OF_MEASUREMENT_AUTHORITY_DOCUMENT.csv
HISTORIC_DISTRICT_USE_TYPE.E55	USE_TYPE_AUTHORITY_DOCUMENT.csv
HISTORIC_RESOURCE_USE_TYPE.E55	USE_TYPE_AUTHORITY_DOCUMENT.csv

More information on the entity types these authority documents describe can be found in Appendix A

Loading Data

As you saw in the Overview Chapter, Arches-HIP is designed to allow you to create inventories of cultural heritage. With HIP, you can import heritage data from a simple (but powerful) text file or from your GIS using a shapefile. If you prefer, you can use the HIP's data entry forms to create heritage records.

4.1 Before You Start

There are a just few things to understand before you start building your cultural heritage inventory. First, Arches-HIP implements the following six resources for cataloging cultural heritage:

- Activity.E7
- Actor.E39
- Heritage Resource.E18
- Heritage Resource Group.E27
- Historical Event.E5
- Information Resource.E73

You saw in the Chapter on Arches-HIP Data Resources that each of these resources (or inventory record types, if you prefer) comes with a set of attributes, some of which are treated as domains or "Controlled Vocabularies".

4.1.1 A Note on Controlled Vocabularies and Authority Files

Let's look at the idea of controlled vocabularies in a bit more detail. Consider the following problem:

> You have people creating cultural heritage data, specifically information about buildings. You want to ensure that all your data are created consistently and wish to use just an approved list of possible terms to describe the architectural style of a building. For example, you want all the data creators to select a style from the following list: "Craftsman", "Beaux Arts", "Bauhaus". How do you do this?

Arches-HIP requires that you decide on the list of terms (we call them controlled vocabularies) that you will use to ensure your heritage data are created and managed using just the terms that you approve of. In Arches, controlled vocabularies may be simple word lists such as "Good, Fair, Poor" to describe the condition of a building. Or a Controlled Vocabulary could be a poly-hierarchical, multi-lingual thesaurus. You may decide just how sophisticated your controlled vocabularies need to be to support your cultural heritage data management needs.

Arches-HIP supports several methods for implementing your controlled vocabularies. You can define controlled vocabularies using an Authority File. As you saw in the last chapter, Arches-HIP includes a number of Authority Files,

one for each controlled vocabulary. Alternatively, you can build your controlled vocabularies interactively using Arches' Reference Data Manager (RDM)

4.2 Preparing Arches: Creating and Loading Controlled Vocabularies

Before you can start creating or importing cultural heritage data with Arches-HIP you will need to create and load the controlled vocabularies that you'll use to ensure consistent data entry and retrieval. You have several options to create the controlled vocabularies:

1. Review the authority files that come with Arches-HIP. These files provide very general terms for each controlled vocabulary and show how you to format a term for loading into HIP.

2. Use the Reference Data Manager (RDM) to create the terms you wish to use for each controlled vocabulary. The RDM provides easy-to-use data entry forms that will let you quickly create the terms you want to use for each controlled vocabulary.

You may wish to check the Arches Project website (http://www.archesproject.org) to see if other Arches users have shared their authority files. If so, you can simply download an authority file and replace the default files that come with Arches-HIP with the authority file that you prefer.

4.2.1 Create or Modify an Authority File

The following steps outline how to modify the authority documents before you load them into HIP. These steps can be followed using excel as your text editor but the final file must be saved as a UTF-8 formatted CSV file.

Note: You can find a summary of authority documents, their format, and steps for modifying them at Arches Documentation under the **Reference Data Graph** section

The steps for modifying authority documents BEFORE you load them into Arches-HIP is:

1. Navigate to your authority files directory. Open the file you wish to modify.

2. If you would like to edit an existing concept simply change the PrefLabel, AltLabels, ParentConceptid, ConceptType, Provider for that concept as necessary. Be sure to adhere to the rules for concept creation outlined in the Arches Documentation .

3. If you would like to add a concept, create a unique `conceptid` for the concept (remember this id must be unique both internal to the authority file and between authority files) and then add your desired PrefLabel, AltLabels, ParentConceptid, ConceptType, Provider.

4. Make sure to review the guidelines of creating a concept in the 'Loading Reference Data' section of the Arches Documentation before modifying the HIP documents to ensure your edited documents will function with HIP.

When you're happy with the contents of each authority file, navigate to `/my_hip_app` and run the following command to load your authroity files into Arches-HIP:

```
$ python manage.py packages -o load_concept_scheme -s 'path to authority files directory'
```

During load the authority files go through a set of validations. If there are errors in your authority documents load will fail and you will have to correct the errors before you try to load the authority files again.

Note: A detailed log of any errors detected in the authority files can be found here: `/my_hip_app/logs/authority_file_errors.txt`

4.2.2 Create Controlled Vocabularies Using the Reference Data Manager

If you wish, you can use the RDM to create each controlled vocabulary interactively. Before you start, You may wish to review a general overview of the RDM (http://http://arches3.readthedocs.org/en/latest/rdm/)

To use the RDM to create your controlled vocabularies, log in to Arches-HIP and click the RDM link. Click on the the "Arches" button on the right side of the screen. This will show you a listing of all the attributes in the six resource graphs that require a controlled vocabulary.

Click on the button labeled "Activity Type". Your screen should look something like this:

What you're seeing is the data entry screen for the terms that you will create for the Activity Type controlled vocabulary.

What is an Activity TYpe? Refer to Appendix A and find the entry for Activity Type. It says that Activity Type is the list of activities that you will associate with the Activity Resource. Examples might include: Investigations, Excavations, Heritage Surveys, Literature Review, or Designation for Protection. Really, the list of activity types is up to you. You may add as many activities as you need to describe your organization's efforts.
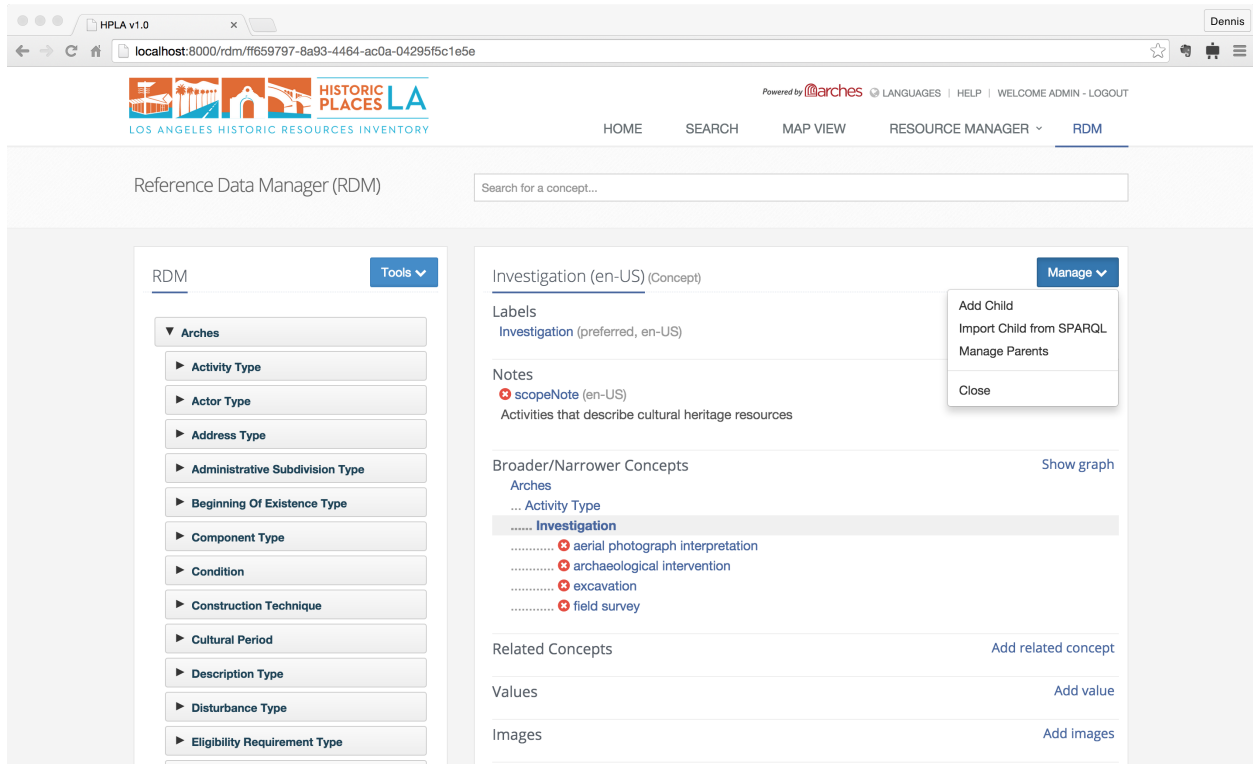
Click on the "Manage" button in the upper right of the form and select "Add Child" from the list of actions. You see a form appear that asks you to enter a label and note for a term that you want to add to your controlled vocabulary. Technically, you are entering a concept into the RDM and providing a preferred label and scope note (in the language that you specify) into the RDM, with is Arches' thesaurus management system.

Here's how you could create an entry for an "Investigation" activity:
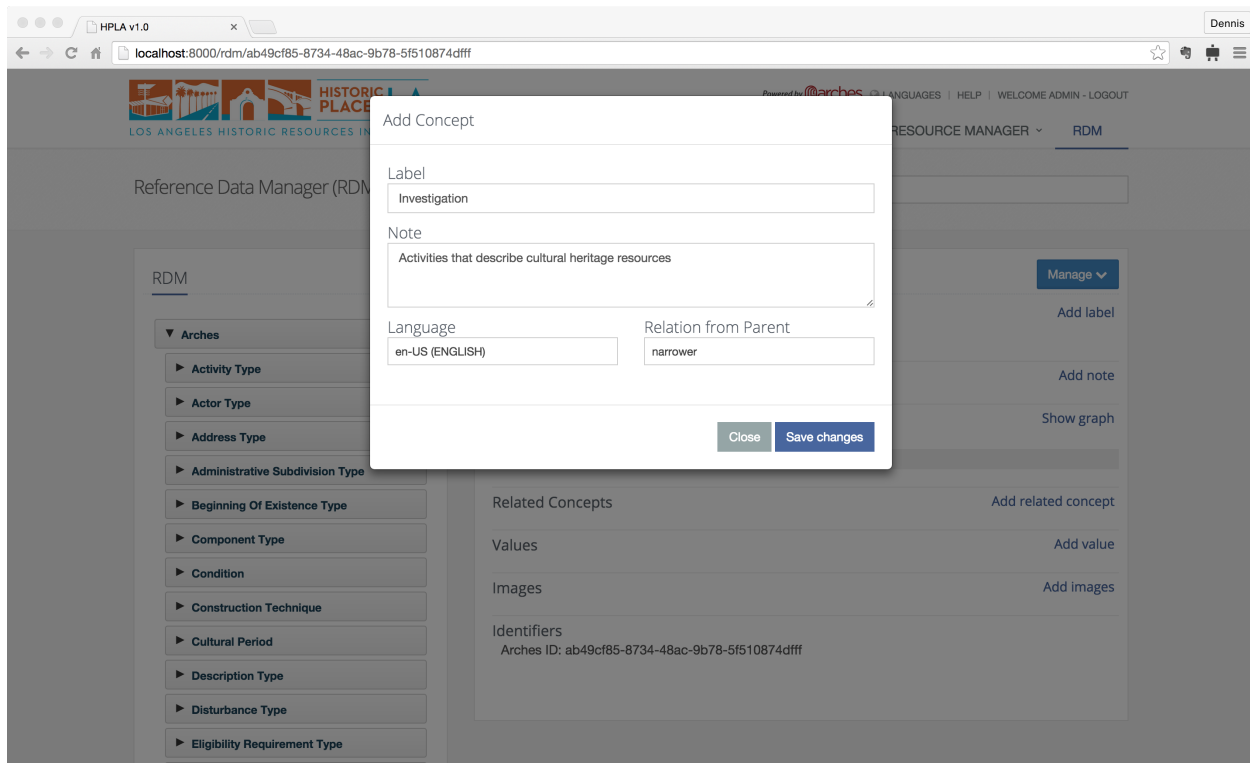
Click "Save Changes" when you're done and you'll see your entry listed under the "Activity Type" concept. You can add as many activities as you like, and you may organize them hierarchically if you want. Simply click on an existing entry in the list and then add a child (like you did for your first entry) to add one level of nesting to your hierarchy. In this way, you could group several terms under "Investigation" to keep them nicely organized and conceptually distinct from a list of Restoration activities.

Here's what a set of activities grouped under the Investigation activity might look like:

You're free to build simple list of activities, or a deeply nested hierarchy. Its up to you to decide how you want to organize your vocabulary. If you haven't already, it's worth reading the Arches RDM documentation to get a more complete description of how the RDM works.

When you're satisfied with your list, click on the button labeled "Dropdowns" (you may need to scroll down the list of vocabularies or click on the "Arches" button again before you see it). Click on the "Activity_Type.E55" button

This form allows you configure which of the terms in your controlled vocabulary will be shown in the dropdown element on the appropriate data entry form for this attribute. Arches introduces this form so that you can decide stop using terms in the future (or add new terms) without compromising your data.

Click the "Add dropdown entry" and type in a term from your activity type controlled vocabulary. For example: "investigation". This will link your term in the controlled vocabulary with the form's dropdown. Note that Arches will automatically add any children that may exist under the term you selected. Go ahead and add all the term you want your data entry staff to see when they create an Activity.
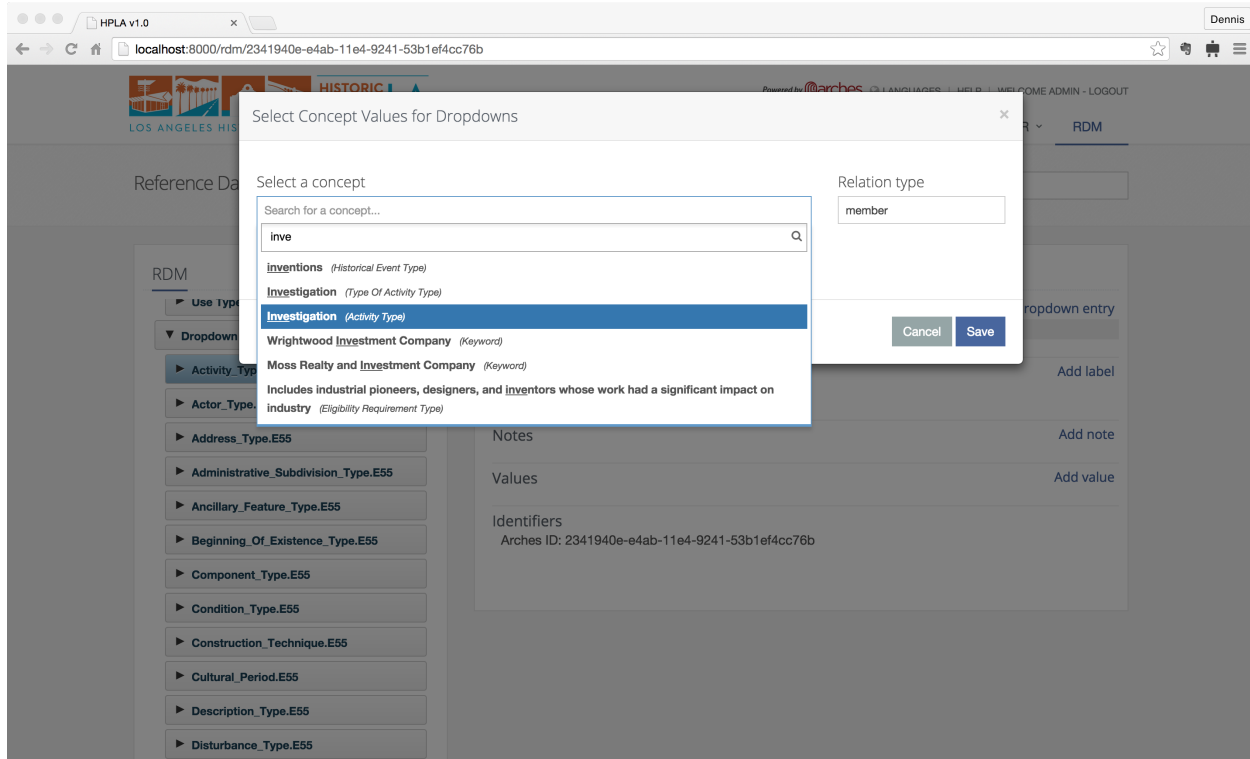
Notice that Arches includes many controlled vocabularies (see the list of vocabularies listed under the Arches button). You'll need to follow the same steps we just went through for each vocabulary in this list. If you don't, you won't be able to enter data into an Arches-HIP resource that requires a controlled vocabulary.

4.3 Importing Cultural Heritage Data into Arches-HIP

This section outlines how to import cultural heritage data into Arches-HIP. Arches-HIP allows you to import data from a `.arches` file or from a GIS (Geographic Information System) using a `.shp` file.

If you would like to use Arches-HIP's data entry forms to create your cultural heritage data, feel free to skip this section entirely. If you would like to load Arches-HIP sample data and not create your own cultural heritage data skip to 'Loading cultural heritage Data.'

If you would like to create and load your own Arches-HIP cultural heritage data this whole section is for you!

4.3.1 Importing / Creating Cultural Heritage Data Using the .ARCHES File

If you have existing cultural heritage data that you want to import into Arches, you'll need to prepare either an ".arches" text file or a GIS shapefile. Either file type will allow you automatically import thousands of cultural heritage data records into Arches-HIP.

Unsure of which format to use? The .arches file defines a simple format and allows you to import very sophisticated datasets (e.g.: heritage data with complex attributes such as multiple geometry types, names, typology). Use a shapefile to import your existing data if you use GIS to create data or if your cultural heritage data is "flat" (e.g.: one value per attribute, such as a single name for each heritage resource).

Creating a ''.arches'' file

For general directions on how to create a `.arches` file you'll want to refer to the 'Loading Business Data' section of the Arches Documentation. This section will review rules specific to creating a `.arches` file based on HIP graphs.

1. In this example we'll begin by creating a text file called `example.arches` and adding the following text to the first line:

```
RESOURCEID|RESOURCETYPE|ATTRIBUTENAME|ATTRIBUTEVALUE|GROUPID
```

2. On the second line enter your unique resourceid (e.g.: a unique identifier for the resource record) followed by a `|`:

```
RESOURCEID|RESOURCETYPE|ATTRIBUTENAME|ATTRIBUTEVALUE|GROUPID
unique_resource_id|
```

3. Now add a valid Arches-HIP resource type followed by a `|`:

```
RESOURCEID|RESOURCETYPE|ATTRIBUTENAME|ATTRIBUTEVALUE|GROUPID
unique_resource_id|ACTIVITY.E7|
```

Note: Valid Arches-HIP resource types include:

- Activity.E7

- Actor.E39

- Heritage Resource.E18

- Heritage Resource Group.E27

- Historical Event.E5

- Information Resource.E73

---

4. Next, declare the attribute that you want to load into HIP. You'll need to declare a valid Arches-HIP AT-TRIBUTENAME. Remember, `ATTRIBUTENAME` relates directly to an entity type from a resource graph. In fact the only valid values for `ATTRIBUTENAME` in the `.arches` file are entity types from the resource graph nodes file indicated in the `RESOURCETYPE` column. More specifically, only valid entity types are allowed to store data and business values. Furthermore, the validation process stringently restricts all files from storing data in any entity types that disallow that.

   Look here for a list of entity types for each Arches-HIP resource.

   Add the entity type name from the graph to the `ATTRIBUTENAME` column in your CSV followed by a `|`:

```
RESOURCEID|RESOURCETYPE|ATTRIBUTENAME|ATTRIBUTEVALUE|GROUPID
unique_resource_id|ACTIVITY.E7|START_DATE_OF_EXISTENCE.E49|
```

5. Next in the `ATTRIBUTEVALUE` 'column' input your specific information for that attribute ensuring that it is in the correct format for that data type (i.e. is formatted correctly for the attributes business table). For example, if you want to load a date field, make sure that you enter an value that is formatted as a date. Make sure your value is followed by a `|`:

```
RESOURCEID|RESOURCETYPE|ATTRIBUTENAME|ATTRIBUTEVALUE|GROUPID
unique_resource_id|ACTIVITY.E7|START_DATE_OF_EXISTENCE.E49|1945-01-12|
```

Note: Arches-HIP recognizes the following formats:

```
String: Words

Numbers: 32 or 3.1417

Dates: YYYY-MM-DD  like 2015-04-09 (which represents April 9, 2015)

Geometries: Well Known Text encoding of a geospatial geometry (see http://en.wikipedia.org/wiki/

Files: A path to a file that you want to upload to Arches-HIP.  See Arches documentation for mor

Controlled Vocabulary Terms: The unique identifier of the term you want to use.  If you loaded y
```

6. Finally, if the data item is to be grouped with another piece of data make sure to add a `GROUPID`. For example, if you want to add a start date and the type of start date (e.g.: July 10, 1934 is a BUILT DATE). If the data item is not intended to be grouped simply enter a generic value for `GROUPID`:

```
RESOURCEID|RESOURCETYPE|ATTRIBUTENAME|ATTRIBUTEVALUE|GROUPID
unique_resource_id|ACTIVITY.E7|START_DATE_OF_EXISTENCE.E49|1945-01-12|BEGINNING_OF_EXISTENCE.E63
unique_resource_id|ACTIVITY.E7|BEGINNING_OF_EXISTENCE_TYPE.E55|BEGIN_EXIST:1|BEGINNING_OF_EXISTE
```

Here the `START_DATE_OF_EXISTENCE.E49` attribute is grouped together with a `BEGINNING_OF_EXISTENCE_TYPE.E55` attribute.

Note: A few things to remember:

(a) All attributes of a resource must be on adjacent lines in your .arches file.

(b) Some attributes must be paired with another attribute to have meaning. For instance, a BEGINNING_OF_EXISTENCE_TYPE.E55 attribute would have no meaing if not paired with a START_DATE_OF_EXISTENCE.E49 attribute.

Note: Here is a list of attributes that must be paired in Arches-HIP:

- `BEGINNING_OF_EXISTENCE_TYPE.E55`, `START_DATE_OF_EXISTENCE.E49`

- `END_OF_EXISTENCE_TYPE.E55`, `END_DATE_OF_EXISTENCE.E49`

- `NAME.E41`, `NAME_TYPE.E55`

- `DESCRIPTION.E62`, `DESCRIPTION_TYPE.E55`

- `RIGHT.E30`, `RIGHT_TYPE.E55`

- `TITLE.E41`, `TITLE_TYPE.E55`

- `ACTOR_APPELLATION.E82`, `NAME_TYPE.E55`

---

7. You'll need to add a line for each attribute that you want to load for a given resource record. This means that an .Arches file can get very large. Obviously, you won't want to write out every line by hand. But now that you understand the format, you can use Excel, a database, or open source tools (such as Open Refine, http://openrefine.org/) to automate the creation of a .Arches file.

8. In some instances you may want to relate resources together (e.g. you may want to relate many resources to one resource district or an actor to a particular heritage resource). To do this you will want to create a .relations file to accompany your .arches file. The .relations file should have the same name as your .arches file and be stored in the same directory location. The .relations file contains six columns resourceid_from, resourceid_to, start_date, end_date, relation_type, notes. For more instructions on how to create a .relations file see the Arches Documentation.

> **Note:** In order to create a relationship between two resources both resourceids referenced in the .relations file must be present in your .arches file.

4.3.2 Importing Creating Cultural Heritage Data Using a Shape File

Arches makes it easy to import cultural heritage data from a GIS shapefile. You'll need to create a simple configuration file that tells Arches-HIP how to associate the information in your shapefile with a resource.

> **Note:** A few things to keep in mind before we begin
>
>   • Your shapefile can contain only one resource type per shapefile. For example, you would need one shapefile to load Heritage Resources and a second shapefile for Activities.
>
>   • Your shapefile projection must be defined as WGS84 Lat/Lon (EPSG:4326).

1. **Create a configuration file**

   You'll need to create configuration file to tell Arches how the fields in your shapefile correspond to nodes in a resource graph. The configuration file should have the same base name as your shapefile with the extension `.config`. For example, if your shapefile is called `heritage_resources.shp`, you will will need a `heritage_resources.config` file along with your `heritage_resources.dbf`, etc.

   The `.config` file is formatted in JSON as you can see in this example:

   ```
   {
           "RESOURCE_TYPE": "HERITAGE_RESOURCE.E18,
           "GEOM_TYPE": "SPATIAL_COORDINATES_GEOMETRY.E47,
           "FIELD_MAP": [
                   ["status","STATUS.E55],
                   ["start","START_DATE_OF_EXISTENCE.E49],
                   ["name","NAME.E41],
                   ["place_addr","PLACE_ADDRESS.E45],
                   ["descrip","DESCRIPTION.E62],
                   ["heri_type","HERITAGE_RESOURCE_TYPE.E55]
           ]
   }
   ```

   The JSON should contain the following properties:

   • `RESOURCE_TYPE`: the resource type of your shapefile.

   • `GEOM_TYPE`: the name of the entity type that Arches HIP uses to manage geometry.

   • `FIELD_MAP`: a list of shapefile fields and their corresponding entity types. Each field must be enclosed in brackets, with the shapefile column name in the first position and the `entitytypeid` in the second.

   > **Note:** The curly braces at the beginning and end of the file are important!

2. **Confirm Your Shapefile Data**

   There are just a couple of details to keep in mind when creating your shapefile.

   Remember that the projection of your data must be EPSG:4326 (WGS84 Lat/Lon).

   The data type of your shapefile columns need to match the datatype of the nodes (entitytypes) to which they are mapped. For example, if you want to load a "name" column in your shapefile to "NAME.E41" (which the HIP defines as a "string" data type), you must ensure that your shapefile treats "name" as a string.

If you want to load a field in your shapefile to an Arches controlled vocabulary, you'll need to make sure that the values in the shapefile match a preferred label in Arches' Reference Data Manager (RDM). For example, if you want to load values from your shapefile into the `HERITAGE_RESOURCE_TYPE.E55` node, the values in your shapefile column must match a preferred label in the RDM scheme (by default Arches uses the `HERITAGE_RESOURCE_TYPE_AUTHORITY_DOCUMENT.csv` scheme)

4.4 Load a .ARCHES or Shapefile into Arches-HIP

To load cultural heritage data, navigate to `/my_hip_app` and run the following command:

```
$ python manage.py packages -o load_resources -s 'path to file'
```

Note: This command works for both `.arches` and `.shp` files.

4.4.1 Arches-HIP Data Validation

In addition to the validation that is performed on every `.arches` file from Arches, there is some specific HIP validation that a `.arches` file must pass as well. These validations are listed below:

1. **Check Duplicates in Group**

   Some entity type groups should only ever contain zero or one of a certain entity type. For instance, every `NAME.E41` should be associated to one and only one `NAME_TYPE.E55`. You may have multiple names per resource or the same name repeated multiple times but each instance of `NAME.E41` should be associated with one and only one `NAME_TYPE.E55`. Here's a list of entity types that must appear only zero or one times in a given group.

   - `ACTOR_APPELLATION.E82`
   - `ADDRESS_TYPE.E55`
   - `COMPONENT_TYPE.E55`
   - `CULTURAL_PERIOD.E55`
   - `DESCRIPTION.E62`
   - `DESCRIPTION_TYPE.E55`
   - `EVALUATION_CRITERIA_TYPE.E55`
   - `FILE_PATH.E62`
   - `FROM_DATE.E49`
   - `HERITAGE_RESOURCE_TYPE.E55`
   - `HERITAGE_RESOURCE_USE_TYPE.E55`
   - `NAME.E41`
   - `NAME_TYPE.E55`
   - `PLACE_ADDRESS.E45`
   - `RIGHT.E30`
   - `RIGHT_TYPE.E55`
   - `THUMBNAIL.E62`

- `TITLE.E41`

- `TITLE_TYPE.E55`

- `TO_DATE.E49`

Remember there can be multiple instances of these entity types per resource just not per group within resource.

2. Check Paired Attributes**

Some attributes should always be paired with another attribute. For example, for each NAME_TYPE.E55 in your .arches file there should be a corresponding NAME.E41, otherwise the NAME_TYPE.E55 does not describe anything. Arches-HIP checks to see that the following attriubtes are paired in your .arches file.

- `BEGINNING_OF_EXISTENCE_TYPE.E55`, `START_DATE_OF_EXISTENCE.E49`

- `END_OF_EXISTENCE_TYPE.E55`, `END_DATE_OF_EXISTENCE.E49`

- `NAME.E41`, `NAME_TYPE.E55`

- `DESCRIPTION.E62`, `DESCRIPTION_TYPE.E55`

- `RIGHT.E30`, `RIGHT_TYPE.E55`

- `TITLE.E41`, `TITLE_TYPE.E55`

- `ACTOR_APPELLATION.E82`, `NAME_TYPE.E55`

Note: Any validation errors that occur during a data load can be viewed in the following file: `my_hip_app/my_hip_app/logs/validation_errors.txt`

4.5 Unloading Cultural Heritage Data

If you managed to load data that you did not intend to load, or if for some reason you changed your mind about loading your data, don't worry. You can easily unload a batch of data from Arches.

To do this you will need the Load ID that was created when you loaded your data. Arches writes the Load ID to the terminal you used to start the import, and the Load ID is also located in the notes column of your `data.edit_log` table.

The Load ID will look something like this: "LOADID:2014-12-2-7-17-23884" (e.g.: the timestamp by year-month-day-hour-minute-microsecond) of the start of your resource load, so that may help you find the correct Load ID if you are unsure.

Once you have the correct Load ID, you can delete all of the resources created during a load with the following command:

```
$ python manage.py packages -o remove_resources -load_id YOUR_LOADID
```

In order to use HIP three types of data must be loaded to the database: resource graphs, authority documents and cultural heritage data. Below we outline what data comes included with HIP, simple instructions on how this data can be modified and how to create your own cultural heritage data.

4.6 Optional GIS Layers for Administrative Areas

Arches-HIP allows you to load GIS data as well as cultural heritage data. This will allow Arches-HIP to compare the spatial coordinates of a cultural heritage resource with GIS data describing administrative areas and cadastres to automatically determine where your heritage resources overlap administrative and ownership polygons.

The parcels and overlays tables within the "aux" schema of Arches-HIP's Postgres database supports using GIS layers to automate the entry of spatially derivable attributes. In HIP, Arches automates the entry of ADMINISTRATIVE_SUBDIVISION and ADMINISTRATIVE_SUBDIVISION_TYPE (for both Heritage Resource and Heritage Group) by comparing the geometry of the resource against the geometry of features in the aux.overlays table. Those overlay features that the heritage resource spatially interacts with are assumed to define the correct administrative area(s) and are associated with the resource.

This "automated" data entry is a nice-to-have feature. That is, if you do not have GIS layers that define your administrative subdivisions, then no problem. The service that queries the GIS layers for information will not fire, and your users can manually key in this information just like any other attribute captured in the HIP data entry forms. Conversely, if you do use this feature, you can still override spatially captured attributes by writing over them manually within the Administrative Subdivision form.

If you wish to take advantage of this feature, it will be necessary to import your GIS data into the overlays table with valid data. The overlays table contains the following columns:

> **overlayty (string)** - Identifies the type of administrative subdivision that the geometry represents. For the spatial capture functionality to work, the value stored here must be consistent with the prefLabel of a concept from the Administrative Subdivision authority document. This value will populate the ADMINISTRATIVE_SUBDIVISION_TYPE node whenever the service is executed.

> **overlayval (string)** - This is the value that will populate the ADMINISTRATIVE_SUBDIVISION node whenever the service is executed. For reference, ADMINISTRATIVE_SUBDIVISION is a non-constrained text field in Arches.

> **geometry (geometry, 4326)** - We expect that overlay records will be polygons and they must be stored in EPSG 4326

> **overlayid (uuid NOT NULL)** - This is the unique ID for the record within Arches. Arches will populate this automatically unless you pass in your own UUID values.

The overlays table is intended to contain data for all administrative subdivisions. As such, if you have more than one overlay type (as indicated in the overlayty column), then it is likely that you will have overlapping polygons in your overlays data.

To populate the overlays table, use the following steps:

1> Prepare a polygon shapefile that mimics the structure of the overlays table as defined above. Remember, the shapefile need not contain an overlayid column, the geometry must be set of EPSG 4326, and the geometry data will not be explicitly called out in the attributes of the shapefile.

Tip: You can access an empty template shapefile that will easily populate into the overlays table from the repo here: `...HIP\docs\overlays template shapefile.zip` or click Shapefile.zip to download.

2> Use the PostGIS Shapefile Import/Export Manager to append the shapefile records to the aux.overlays table. Configuration of the import is pictured below:
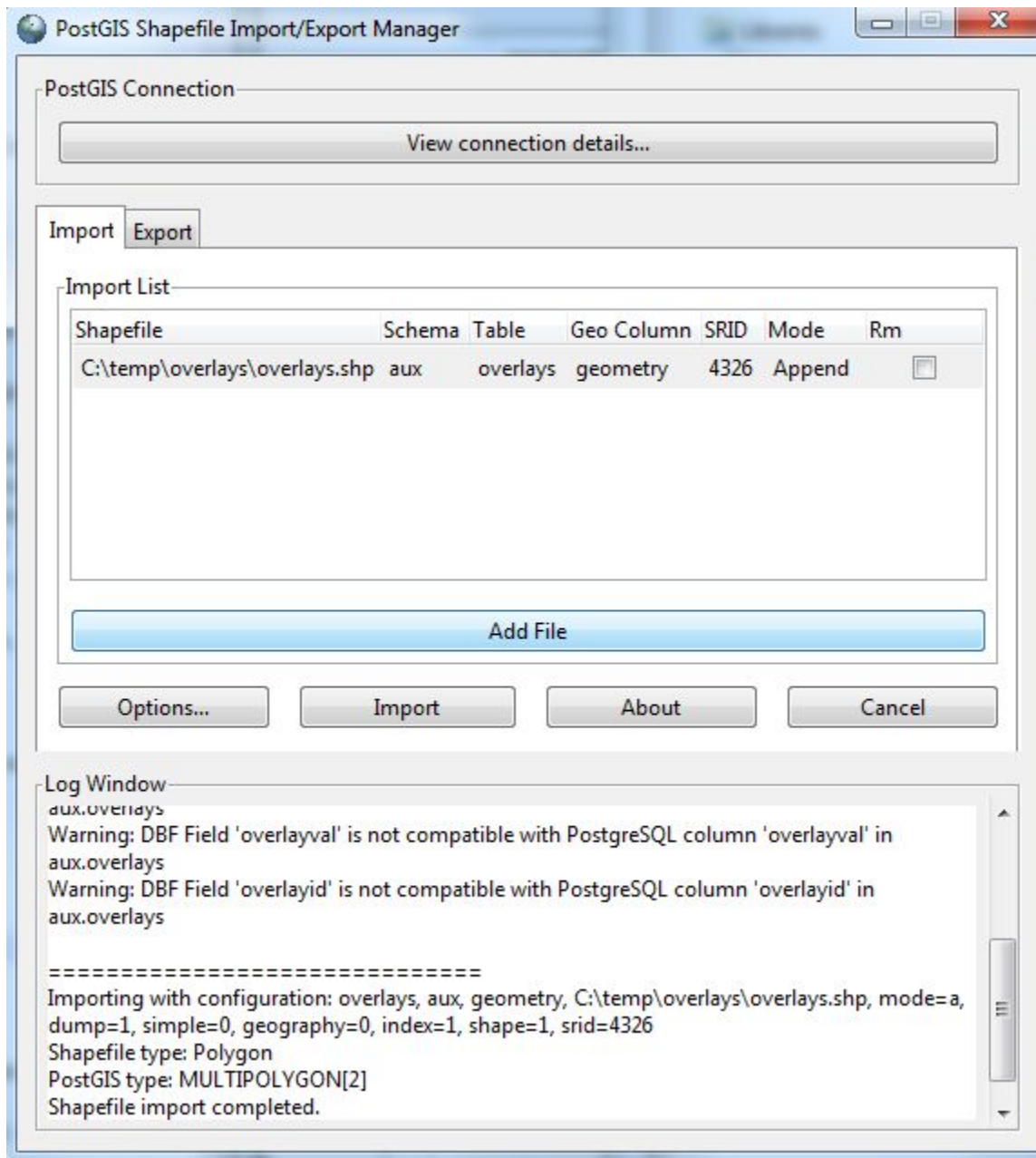
Please take special note of the following:

- Mode is set to append

- SRID is set to 4326

- Schema is set to "aux"

If you made a mistake in loading the data, you can delete the records with this sql command: "delete from aux.overlays;" and then re-run the PostGIS Shapefile Import/Export Manager tool.

Using Arches-HIP

User Documentation for the Arches web application is coming soon!

System Administration

6.1 Administration App

Arches uses the very powerful Django administration application to drive system administration. This interface is available at: `{ your application url }/admin`

From here, administrator users can manage users, groups and permissions. Arches-HIP will automatically create permissions for each entity type tracked, so that administrators can manage access to Arches data on a per entity basis. You will find these permissions available to be assigned to users and groups in the Django administration application.

For more on using the Django administration interface, please see the Django documentation: https://docs.djangoproject.com/en/1.6/ref/contrib/admin/

6.2 Reference Data Manager (RDM)

The HIP Reference Data Manager allows users to create, modify and delete reference data used by HIP. This data is used to populate labels in search and dropdown lists. A more complete discussion of using the RDM can be found in the Arches-HIP user documentation (coming soon!).

6.3 Exporting Data

By default, Arches will export the id, primary name and resource type to KML, shapefile and CSV formats. However, if you need to export more information for a resource, or customize the column names of your export file, an export mapping file is required.

6.3.1 Resource export mappings

The resource export mappings is a JSON file that maps resources in your search results, to the schema of an export format; KML, CSV or SHP. If you need to create this file, its path needs to be assigned to the `EXPORT_CONFIGS` variable in your applications `settings.py` file. For example:

```
EXPORT_CONFIG = os.path.normpath(os.path.join(PACKAGE_ROOT, 'source_data', 'business_data', 'resource
```

The top level members of the JSON object are the format extensions (e.g. "csv", "shp", "kml"). Each format extension has the following properties:

- `NAME`: base name of the export file

- SCHEMA: array of objects defining each field to be added to your exported file. Each object should have the following properties:

  - `field_name`: name of the exported field (required)

  - `source`: The location of the data in the search results. If the value is "field_map" the application will lookup the value using the field map for the resource.

  - `alternatename` : every name for the resource that is not primary will be concatenated into a list of alternate names. A value of "resource_name" will lookup the resource name from the `RESOURCE_TYPE_CONFIGS` in the application's `settings.py` file.

  - `data_type`: "str", "datetype", or "float" (required only for shapefiles)

  - `data_length`: an integer in quotes (e.g. "128" required only for shapefiles)

- RESOURCE_TYPES: an object containing an object for each resource type that will be exported. The key for each resource type object is the resource's `entitytypeid`. The value for each type contains the following properties:

  - `FIELD_MAP`: a list of objects each containing information needed to map child entity data to an export file field. Properties for the field map include:

    * `field_name`: The field_name value - match a field in the `SCHEMA` (required)

    * `entitytypeid`: The `entitytypeid` of the child entity (required)

    * `value_type`: A `conceptid` of the value used to define the value's type. For example, an address may have different types such as primary or postal. If you want to export only the primary address for this column, you can add the concept type for 'primary address' here. Concept ids can be found in your application's Reference Data Manager (RDM). (optional)

    * `alternate_entitytypeid`: Alternate entity type to use if no value is available for the `entitytypeid` (optional)

6.3.2 Example Resource Export Mappings

```
{
    "csv": {
        "NAME": "HistoricPlacesLA_Search_Results_Export",
        "SCHEMA": [
            {"field_name": "PRIMARY NAME","source": "primaryname"},
            {"field_name": "OTHER NAMES","source": "alternatename"},
            {"field_name": "ARCHES ID","source": "entityid"},
            {"field_name": "ARCHES RESOURCE TYPE","source": "resource_name"},
            {"field_name": "TYPE","source": "field_map"},
            {"field_name": "PRIMARY ADDRESS/LOCATION DESCRIPTION","source": "field_map"},
            {"field_name": "DESIGNATIONS","source": "field_map"}
        ],
        "RESOURCE_TYPES": {
            "ACTOR.E39": {
                "FIELD_MAP": [
                    {
                        "field_name": "TYPE",
                        "entitytypeid": "ACTOR_TYPE.E55"
                    },
                    {
                        "field_name": "PRIMARY ADDRESS/LOCATION DESCRIPTION",
                        "entitytypeid": "PLACE_ADDRESS.E45",
                        "value_type": "e4f5bd2f-56b7-4b8d-ac48-7e6d90e530ae",
```

```
                    "alternate_entitytypeid": "DESCRIPTION_OF_LOCATION.E62"
                }
            ]
        },
        "HERITAGE_RESOURCE.E18": {
            "FIELD_MAP": [
                {
                    "field_name": "TYPE",
                    "entitytypeid": "HERITAGE_RESOURCE_TYPE.E55"
                },
                {
                    "field_name": "PRIMARY ADDRESS/LOCATION DESCRIPTION",
                    "entitytypeid": "PLACE_ADDRESS.E45",
                    "value_type": "e4f5bd2f-56b7-4b8d-ac48-7e6d90e530ae",
                    "alternate_entitytypeid": "DESCRIPTION_OF_LOCATION.E62"
                },
                {
                    "field_name": "DESIGNATIONS",
                    "entitytypeid": "TYPE_OF_DESIGNATION_OR_PROTECTION.E55"
                }
            ]
        }
    }
},
"kml": {
    "NAME": "HistoricPlacesLA_Search_Results_Export",
    "SCHEMA": [
        {"field_name": "primary_name","source": "primaryname"},
        {"field_name": "other_names","source": "alternatename"},
        {"field_name": "arches_id","source": "entityid"},
        {"field_name": "arches_resource_type","source": "resource_name"},
        {"field_name": "geometry","source": "geometries"},
        {"field_name": "type","source": "field_map"},
        {"field_name": "primary_address_or_description","source": "field_map"},
        {"field_name": "designations","source": "field_map"}
    ],
    "RESOURCE_TYPES": {
        "ACTOR.E39": {
            "FIELD_MAP": [
                {
                    "field_name": "type",
                    "entitytypeid": "ACTOR_TYPE.E55"
                },
                {
                    "field_name": "primary_address_or_description",
                    "entitytypeid": "PLACE_ADDRESS.E45",
                    "value_type": "e4f5bd2f-56b7-4b8d-ac48-7e6d90e530ae",
                    "alternate_entitytypeid": "DESCRIPTION_OF_LOCATION.E62"
                }
            ]
        },
        "HERITAGE_RESOURCE.E18": {
            "FIELD_MAP": [
                {
                    "field_name": "type",
                    "entitytypeid": "HERITAGE_RESOURCE_TYPE.E55"
                },
                {
```

```
                                "field_name": "primary_address_or_description",
                                "entitytypeid": "PLACE_ADDRESS.E45",
                                "value_type": "e4f5bd2f-56b7-4b8d-ac48-7e6d90e530ae",
                                "alternate_entitytypeid": "DESCRIPTION_OF_LOCATION.E62"
                            },
                            {
                                "field_name": "designations",
                                "entitytypeid": "TYPE_OF_DESIGNATION_OR_PROTECTION.E55"
                            }
                        ]
                    }
                }
            },
            "shp": {
                "NAME": "HistoricPlacesLA_Search_Results_Export",
                "SCHEMA": [
                    {"field_name": "prime_name","source": "primaryname","data_type": "str","data_length": "12
                    {"field_name": "arches_id","source": "entityid","data_type": "str","data_length": "128"},
                    {"field_name": "resource","source": "resource_name","data_type": "str","data_length": "12
                    {"field_name": "othernames","source": "alternatename","data_type": "str","data_length": "
                    {"field_name": "type","source": "field_map","data_type": "str","data_length": "128"},
                    {"field_name": "address","source": "field_map","data_type": "str","data_length": "128"},
                    {"field_name": "designatns","source": "field_map","data_type": "str","data_length": "128"
                ],
                "RESOURCE_TYPES": {
                    "ACTOR.E39": {
                        "FIELD_MAP": [
                            {
                                "field_name": "type",
                                "entitytypeid": "ACTOR_TYPE.E55"
                            },
                            {
                                "field_name": "address",
                                "entitytypeid": "PLACE_ADDRESS.E45",
                                "value_type": "e4f5bd2f-56b7-4b8d-ac48-7e6d90e530ae",
                                "alternate_entitytypeid": "DESCRIPTION_OF_LOCATION.E62"
                            }
                        ]
                    },
                    "HERITAGE_RESOURCE.E18": {
                        "FIELD_MAP": [
                            {
                                "field_name": "type",
                                "entitytypeid": "HERITAGE_RESOURCE_TYPE.E55"
                            },
                            {
                                "field_name": "address",
                                "entitytypeid": "PLACE_ADDRESS.E45",
                                "value_type": "e4f5bd2f-56b7-4b8d-ac48-7e6d90e530ae",
                                "alternate_entitytypeid": "DESCRIPTION_OF_LOCATION.E62"
                            },
                            {
                                "field_name": "designatns",
                                "entitytypeid": "TYPE_OF_DESIGNATION_OR_PROTECTION.E55"
                            }
                        ]
                    }
                }
```

```
        }
}
```

Settings and Customization

There are lots of ways that you can customize your Arches-HIP application to best suit your needs.

Most of these will require comfort with modifying Python, HTML, CSS and/or Javascript files but are designed to be simple changes that can be easily applied without diving deep into the rest of the Arches or Arches-HIP code.

Note: In a production environment (using Apache), you will generally need to run `collectstatic` (https://docs.djangoproject.com/en/1.6/ref/contrib/staticfiles/#collectstatic) and restart Apache to apply these changes.

No action should be required when using the development server (Django).

7.1 Home Page and Branding

Modifying the home page and branding of your Arches-HIP application involves editing some simple HTML templates to modify the look and feel. A basic familiarity with HTML and CSS is assumed here.

Your application starts with templates that you can use to start modifying the home page and branding of the web application. These files are contained in the following folder:

```
my_hip_app/my_hip_app/templates/
```

You can add your own CSS style rules (applied application wide) in the following file:

```
my_hip_app/my_hip_app/media/css/package.css
```

Home Page

index.htm contains the template for the main home page content. You can modify this page to change the image slide show on the home page or modify the saved searches (more on that below).

This is the content of the first page that your users will see, so you'll probably want to really make it your own. You should consider doing the following:

- Update the images and text in the layer slider
- Add you own text to the title block below the layer slider
- Update the saved searches to suit your data (more on that below)

For example you can add or remove images to the layer slider as blocks of HTML like this:

```
<div class="ls-layer" style="">

    <!-- Background -->
```

```
            <img src="{% static 'img/arches_splash/slide1.jpg' %}" class="ls-s1" style="top: -450px; lef

            <!-- Text -->
            <span class="ls-s-1" style="line-height: 42px; font-size:22px; font-weight: 200; color:#fff;
            {% trans 'Arches Includes' %}
            </span>

            <span class="ls-s-1" style="line-height: 42px; font-size:42px; font-weight: 200; color:#fff;
            {% trans 'Activities and' %} </br>
            {% trans 'Important Historic Events' %}
            </span>

            <!-- Photo Credit -->
            <span class="ls-s-1" style="line-height: 16px; font-size:12px; color:#fff; top:470px; left:
            {% trans 'Photo: Image Caption, ' %}<a href="javascript:void(0);">{% trans 'Photographer Nam
            </span>

        </div>
```

Each block will create an item in the slider. You can modify the `src` path in the `<img>` tag and update the text in the tags below it to make the item show an image and message of your choosing.

Header Branding

The templates that manage site-wide branding are `footer.htm`, `header.htm`, and `copyright.htm`. These templates are used by default for branding on all web application pages.

You should probably keep much of the structure of these templates, and simply input your own labels, icons and information as appropriate.

For example, the header logo on all pages can be altered by modifying the following block of HTML start at line 54:

```
<!-- Default Logo, Replace with your own -->
<a class="navbar-brand" href="{% url 'home' %}">
    <img id="logo-header" class="arches-logo" src="{{ STATIC_URL }}img/arches_logo.png" alt="Log
</a>
```

Copy your logo to the img/ folder and just replace 'arches_logo.png' with the file name of your logo. Open main.css in a text editor and search for 'arches-logo' (in the initial Arches release its found at line 2530). You can modify:

```
height: 28px;
margin-top: 13px;
```

to set the height and position of you logo.

Footer Branding

The footer template sets your contact information (address, telephone number, email) as well as general information including a short description of your Arches implementation and recent activities.

You can modify the default "About Arches" message by opening the footer.htm file in a text editor and finding the following block of html (around line 10):

```
<!-- Default About Us.  Edit the <p> text below for your own message -->
<div class="col-md-3 sm-margin-bottom-40">
    <div class="heading-footer">
        <h2>About Arches</h2>
    </div>
    <p>Arches is built to create and manage rich inventories of culturally significant resources
```

```
    </div>
    <!-- End About Us -->
```

Edit the "About Arches" text to change the title of this section, and the text between the <p> and </p> tags to change the summary.

Your contact information is found here (around line 51):

```
<!-- Contacts -->
<div class="col-md-3">
    <div class="heading-footer"><h2>Contacts</h2></div>
    <ul class="list-unstyled contacts">
        <li>
            <i class="radius-3x fa fa-map-marker"></i>
            601 Montgomery St, Suite 1095,<br>
            San Francisco, CA 94111
        </li>
        <li>
            <i class="radius-3x fa fa-phone"></i>
            (+123) 456 7890 (main)<br>
            (+123) 456 7891 (mobile)
        </li>
        <li>
            <i class="radius-3x fa fa-globe"></i>
            <a href="#">info@arches.org</a><br>
            <a href="#">www.archesproject.org</a>
        </li>
    </ul>
</div>
<!-- End Contacts -->
```

Simply update the default address, phone numbers, and email/web address with your own values.

Copyright Branding

The bottom panel contains basic information about the Arches and Arches-HIP versions, as well as a link to the Arches project website. It also presents buttons that allow you to include your social media information. Open copyright.htm in a text editor to update these links (found near line 14):

```
<div class="col-md-4">
    <ul class="list-inline dark-social pull-right space-bottom-0">
        <li>
            <a data-placement="top" data-toggle="tooltip" class="tooltips" data-original-title="
                <i class="fa fa-facebook"></i>
            </a>
        </li>
        <li>
            <a data-placement="top" data-toggle="tooltip" class="tooltips" data-original-title="
                <i class="fa fa-twitter"></i>
            </a>
        </li>
        <li>
            <a data-placement="top" data-toggle="tooltip" class="tooltips" data-original-title="
                <i class="fa fa-google-plus"></i>
            </a>
        </li>
        <li>
            <a data-placement="top" data-toggle="tooltip" class="tooltips" data-original-title="
                <i class="fa fa-instagram"></i>
            </a>
```

```
        </li>
        <li>
            <a data-placement="top" data-toggle="tooltip" class="tooltips" data-original-title="
                <i class="fa fa-youtube"></i>
            </a>
        </li>
        <li>
            <a data-placement="top" data-toggle="tooltip" class="tooltips" data-original-title="
                <i class="fa fa-soundcloud"></i>
            </a>
        </li>
    </ul>
</div>
```

Find the "href="#" tag for each button and replace the "#" with the link to your site. Simply delete the buttons that you don't use. For example, if you want to delete the Soundcloud button, delete the following block:

```
<li>
    <a data-placement="top" data-toggle="tooltip" class="tooltips" data-original-title="Soundclo
        <i class="fa fa-soundcloud"></i>
    </a>
</li>
```

7.2 HIP Title Block

Arches-HIP places a bit of text beneath the images on the home page as a way to brief describe your use of Arches-HIP. You can edit this text by opening the index.htm page in a text editor and finding this section of html (found near line 135):

```
<!--=== HIP Title Block ===-->
<div class="purchase">
    <div class="container">
        <div class="row">
            <div class="col-md-9 animated fadeInLeft">
                <span>{% trans 'Survey Los Angeles' %}</span>
                <p>{% trans "Los Angeles is taking a significant step to protect its rich heritage by
            </div>
            <div class="col-md-3 btn-buy animated fadeInRight">
                <a href="#" class="btn-u btn-u-lg"><i class="fa fa-plus-square"></i> {% trans 'Learn
            </div>
        </div>
    </div>
</div><!--/row-->
<!-- End HIP Title Block -->
```

To change the title of this section, simply replace 'Survey Los Angeles' with your own text. Similarly, replace the text following the <p> tag with your own summary. The "Learn More" button can also be modified. Change the button text if you wish, and replace the "#" with the web address of the page you want your audience to visit.

Of course, you can simply delete this entire block of html if you prefer.

7.3 Saved Searches

Saved searches are the links shown on the home and search pages by default to help users find data of particular interest. You will need to update these to be useful for your application.

To do this, start by using the Arches-HIP web application search page to execute the search that you want to save.

For example, you could do a search for the word 'modern'. You should see the page's URL update. Copy everything after the ? in the URL; you should get something like this:

```
page=1&termFilter=%5B%7B%22inverted%22%3Afalse%2C%22type%22%3A%22string%22%2C%22context%22%3A%22%2
```

There are two html templates in your project that contain saved searches, these files are:

```
my_hip_app/my_hip_app/templates/index.htm
my_hip_app/my_hip_app/templates/views/saved-searches.htm
```

index.htm contains the saved searches for the home page and saved-searches.htm contains the saved searches for the search page.

In both cases, you will find saved searches in blocks that look like this:

```
<div class="col-sm-4">
    <div class="servive-block servive-block-default">
        <i class="icon-custom rounded-x icon-bg-dark fa fa-trophy"></i>
        <a class= "" href="{% url 'search_home' %}?"><h2 class="txt-color-blue">{% trans 'Saved Searc
        <p>{% trans 'Some description here...' %}</p>
    </div>
</div>
```

To update that block to use the example search above for the word modern, I can simply update the label and description text and paste what I copied above after `{% url 'search_home' %}?` in the `<a>` tag's `href` attribute, for example:

```
<div class="col-sm-4">
    <div class="servive-block servive-block-default">
        <i class="icon-custom rounded-x icon-bg-dark fa fa-trophy"></i>
        <a class= "" href="{% url 'search_home' %}?page=1&termFilter=%5B%7B%22inverted%22%3Afalse%2C%
        <p>{% trans 'Find things that are modern.' %}</p>
    </div>
</div>
```

7.4 Additional Content

Feel free to add additional content to the index.htm page. Simply add your content in the area provided in index.htm:

```
<!--=== Put Your Addition Content Here
    Use Arches' exist CSS classes or add your own with package.css ===-->
    <div class="">




    </div>
    <!-- End Additional Content -->
```

7.5 Google Analytics

Google Analytics is a web analytics service offered by Google that tracks and reports website traffic. It is freely available to anyone with a Google account.

To enable Google Analytics on your web application, you will need to update the `GOOGLE_ANALYTICS_TRACKING_ID` setting in your application's `settings.py` file.

Uncomment this line (remove the starting #) if it is commented, and paste in your own Google Analytics tracking id between the single quotes at the end of the line, eg:

```
GOOGLE_ANALYTICS_TRACKING_ID = 'my-tracking-id'
```

Enabling this setting will automatically configure the web application to include the Google Analytics scripts.

7.6 Map Settings

Arches-HIP needs to be configured to properly zoom and constrain map interfaces by default. This can be done by simply modifying some settings found in your application's `settings.py` file:

```
#DEFAULT_MAP_X = 0
#DEFAULT_MAP_Y = 0
#DEFAULT_MAP_ZOOM = 0
#MAP_MIN_ZOOM = 0
#MAP_MAX_ZOOM = 19
#MAP_EXTENT = '-13228037.69691764,3981296.0184014924,-13123624.71628009,4080358.407059081'
```

To use these settings, you will need to 'uncomment' them (remove the # at the start of each line), and then change their values to meet your needs.

`DEFAULT_MAP_X` and `DEFAULT_MAP_Y` are the x and y (respectively) coordinate values which all maps will be centered on by default. `MAP_EXTENT` is a string 'bounding box' [min x, min y, max x, max y] representing the extent to which all maps will be constrained.

All of these coordinates are expressed in the web standard Spherical Mercator projection (http://spatialreference.org/ref/sr-org/6864/). A simple converter tool to go between Lat/Long and Spherical Mercator coordinates can be found here: http://mal2.ch/stuff/latlontomercator.html

`DEFAULT_MAP_ZOOM` is the default zoom level that all maps will start on. `MAP_MIN_ZOOM` and `MAP_MAX_ZOOM` are the minimum and maximum zoom levels that all maps will allow a user to access. More on zoom levels can be found here: http://openlayers.org/en/v3.3.0/doc/tutorials/concepts.html

By default, Arches-HIP will use Bing services for basemaps and geocoding; if you want to use these, you will need to get a Bing Maps Key (https://msdn.microsoft.com/en-us/library/ff428642.aspx) and add it to your `settings.py` file like so (by default at line 21):

```
BING_KEY = 'my-bing-key'
```

7.7 Basemaps

By default, Arches-HIP will supply the application with Bing layers as basemaps, but it is easy to add more basemap layers to Arches-HIP using OpenLayers.

The basemap layers are added in the following file:

```
my_hip_app/my_hip_app/media/js/map/base-layers.js
```

You will notice that at the second to last line of this file, an array called `baseLayers` is returned:

```
return baseLayers;
```

You can add more baseLayers by adding an object containing an OpenLayers 3 layer instance to this array before the return statement.

The objects you add should contain the following keys:

- `id`: String; a unique id for this layer

- `layer`: OpenLayers3 Layer; any valid OpenLayers 3 Layer instance, for more on OpenLayers 3 and layers see: http://openlayers.org/en/v3.3.0/doc/quickstart.html

- `icon`: String; a valid path to an image used as an icon in the basemap list

- `name`: String; a display name for this layer used in the basemap list

For example, I can add a tile layer as a basemap layer like so:

```
baseLayers.push({
    id: 'mybasemap',
    name: 'My Tile Basemap',
    icon: arches.urls.media + 'img/map/my_basemap_icon.png',
    layer: new ol.layer.Tile({
        visible: false,
        source: new ol.source.XYZ({
            url: 'http://api.tiles.mapbox.com/v4/my-mapbox-map/{z}/{x}/{y}.png?access_token=my-access
        })
    })
});

return baseLayers;
```

7.8 Overlays

Additional overlays can also be added to the Arches-HIP map interface. This is quite similar to adding basemap layers except it is done in another file and requires a little extra metadata for each layer.

Overlays are added to the following file:

```
my_hip_app/my_hip_app/media/js/map/layers.js
```

You will find, again, at the bottom of this file an array being returned, this one called `layers`. Additional overlay layers can be added to the `layers` array prior to the line where it is returned.

However, in this case a little extra metadata is passed along with the layer instance in a Javascript object called a LayerModel. The LayerModel is instantiated with an object with the following keys:

- `layer`: OpenLayers3 Layer; any valid OpenLayers 3 Layer instance, for more on OpenLayers 3 and layers see: http://openlayers.org/en/v3.3.0/doc/quickstart.html

- `icon`: String; a css class used for displaying an icon in the overlay list. CSS rules can be added to `package.css` as described above

- `name`: String; a display name for this layer used in overlay list

- `description`: String; an optional description for the layer used in overlay list and for searching

- `categories`: Array; an optional list of categories (Strings) used as keywords for layer search

- `onMap`: Boolean; true to show this layer on the map by default

For example, you could add a tile layer overlay like so:

```
layers.push(new LayerModel({
    name: 'My Tile Layer',
    categories: ['cool', 'fun'],
    description: 'My tile layer is very cool',
    icon: 'tile-layer-icon',
    layer: new ol.layer.Tile({
        source: new ol.source.XYZ({
            url: 'http://api.tiles.mapbox.com/v4/my-mapbox-map/{z}/{x}/{y}.png?access_token=my-access
        })
    })
}));

return layers;
```

7.9 Resource Map Markers

Resource map markers are shown in the web application using icon fonts for scalability and flexibility.

By default, support is provided for Octicons (https://octicons.github.com/) and Font Awesome (http://fortawesome.github.io/Font-Awesome/).

Arches-HIP provides some simple settings for managing how resource markers are displayed on the map. These settings can be found in your `settings.py` file and should look something like this:

```
RESOURCE_MARKER_ICON_UNICODE = '\uf060'
RESOURCE_MARKER_ICON_FONT = 'octicons'
RESOURCE_MARKER_DEFAULT_COLOR = '#C4171D'
```

These settings manage markers as follows:

- `RESOURCE_MARKER_ICON_UNICODE`: the Unicode value of the icon you want to use

- `RESOURCE_MARKER_ICON_FONT`: the font name you want to use ('octicons' or 'FontAwesome')

- `RESOURCE_MARKER_DEFAULT_COLOR`: the hex value of the color you want to use by default

For example, if I wanted to symbolize my resource markers as pink Font Awesome house icons (http://fortawesome.github.io/Font-Awesome/icon/home/), I could adjust my settings like so:

```
RESOURCE_MARKER_ICON_UNICODE = '\uf015'
RESOURCE_MARKER_ICON_FONT = 'FontAwesome'
RESOURCE_MARKER_DEFAULT_COLOR = '#F423FF'
```

7.10 Geocoding

By default, as mentioned above, Arches-HIP uses the Bing geocoder to drive location search on the Map page. This can be customized by applying a new `GEOCODING_PROVIDER` in your application's `settings.py` file.

This setting should point to a python module with a simple interface:

- the module must have a method called `find_candidates`

- `find_candidates` should accept a single argument, which is the string search query
- `find_candidates` should return a list of candidate locations as dictionaries with the following keys:
  - `id`: a unique id for the location
  - `text`: a text description (i.e. name or address) of the location
  - `geometry`: a GeoJSON object (as a python dictionary) containing the location's geometry
  - `score`: an optional score value used for sorting (descending) in results drop down

7.11 Localization

Arches-HIP has put all of the hooks in place in it's code and UI templates to handle being localized via Django's localization API. For more information on how to localize your application using Django, see the Django documentation: https://docs.djangoproject.com/en/1.6/topics/i18n/translation/

Extra

8.1 Production Configuration

When it comes to setting up your app in a production environment, there are a lot of configuration modifications that can be made to optimize the way it is hosted and served. Here's some information on some things we've tested out...

8.1.1 Storing Uploaded Files on AWS Simple Storage Service (S3)

About S3

Amazon Simple Storage Service, or S3, is one of the many services available through Amazon Web Services. S3 can very basically be described as a limitless (for all practical purposes) external hard-drive in the cloud. It's a simple system that is secure and also cheap. To read more about S3, check out the official documentation.

To use S3, you will need an AWS account, which is just an extension of a normal Amazon account. Here's some information on how to get started with AWS.

Why Use S3 with Arches

By the time you are in a production environment, you will have configured Arches with a web server, such as Apache or nginx. While you need a web server to serve the app itself, there are two pieces of the app that can be separated from the web server and served independently. These are the 'static' files (the css, javascript, and logos that are used throughout the app) and the 'media' files (any user uploaded files, such as images or documents).

Many of the existing tutorials are concerned with serving both static and media files, because the more load you can take off of your web server the better. However, there are some advantages to S3 that do not have to do with giving your web server some relief:

- S3 is cheap: As per the price chart, it costs just $.03 per gb/month. So a database with 10gb of photos will have a media storage cost of $3/month, plus a small amount per transaction ($0.004 per 10,000 GET requests, e.g.).

- S3 is scalable: You only pay for the amount of data you have stored, and you have no real limit on how much you can store. This allows for an Arches deployment on a small server, either in-house or a small AWS EC2 or DigitalOcean instance to store hundreds of gigabytes of media–photos, audio, video, documents–without having to restructure to accommodate more data.

The main potential drawback is transfer speed to and from the S3 bucket, but that depends most on your what your default storage option is. If you look around, you'll find some comparison articles like this one that may be helpful.

> **Note:** You should be able to use S3 regardless of where your app is hosted, whether on an internal server, an AWS EC2 instance, a DigitalOcean droplet, etc.

> **Warning:** We've found that by following the steps below, deleting an Information Resource will *not* automatically remove the file from your S3 bucket. You can manually delete files from the bucket for now, or the intrepid developer may check out the answer to this question on the Arches forum.

Steps to Follow

Having worked through a number of existing tutorials (mostly here, here, here, and here), we've distilled these steps to show how you can use S3 in conjunction with your Arches app. Before beginning, you will need to have set up and logged into your AWS account.

#1. Create some new user credentials for your Arches app

The first step is to create some new AWS credentials that your Arches app will use to access the S3 bucket

1. Access the AWS Identity and Access Management (IAM) Console

2. Create a new user (named something like "arches_media"), and download the new credentials. This will be a small .csv file that includes an Access Key ID and a Secret Key.

3. Also, go to the new user's properties, and record the User ARN

#2. Create a new bucket on S3

Next, you'll need to create a new bucket and give it the appropriate settings.

1. Create a bucket, named something like "your-app-media"

2. In the new bucket properties, under Permissions, create a new bucket policy with the following text, inserting your own BUCKET-NAME, and the User ARN for your new user:

```
{
    "Statement": [
        {
          "Sid":"PublicReadForGetBucketObjects",
          "Effect":"Allow",
          "Principal": {
                "AWS": "*"
            },
          "Action":["s3:GetObject"],
          "Resource":["arn:aws:s3:::BUCKET-NAME/*"
            ]
        },
        {
          "Action": "s3:*",
          "Effect": "Allow",
          "Resource": [
                "arn:aws:s3:::BUCKET-NAME",
                "arn:aws:s3:::BUCKET-NAME/*"
            ],
          "Principal": {
                "AWS": [
                    "USER-ARN"
                ]
            }
        }
    ]
}
```

3. Also, make sure that the CORS configuration (click "Add CORS Configuration") looks like this:

```
<CORSConfiguration>
    <CORSRule>
```

```
            <AllowedOrigin>*</AllowedOrigin>
            <AllowedMethod>GET</AllowedMethod>
            <MaxAgeSeconds>3000</MaxAgeSeconds>
            <AllowedHeader>Authorization</AllowedHeader>
        </CORSRule>
    </CORSConfiguration>
```

#3. Update the Virtual Environment

> In order to configure Arches to use your new bucket, you need to install a couple of extra Django modules
> in your virtual environment. These will augment Django's flexibility in how it stores uploaded media.
>
>> 1. Once you have activated your virtual environment, use this command:

```
        (ENV) $: pip install boto django-storages
```

#4. Update settings.py

> Finally, you need to tell your app to use these new modules, give it the necessary credentials, and tell it
> where to store (and find) the uploaded media. Open the your settings.py file...
>
>> 1. Find the line that defines the settings "INSTALLED_APPS" and add 'storages' to it. It should look
>>   like this:

```
        INSTALLED_APPS = INSTALLED_APPS + (PACKAGE_NAME,'storages',)
```

>> 2. Next, add the following lines, replacing the AWS settings values with information from earlier steps
>>   (remember the "credentials.csv" file you downloaded?):

```
        DEFAULT_FILE_STORAGE = 'storages.backends.s3boto.S3BotoStorage'
        AWS_STORAGE_BUCKET_NAME = 'aws_bucket_name'
        AWS_ACCESS_KEY_ID = 'aws_access_key_id'
        AWS_SECRET_ACCESS_KEY = 'aws_secret_access_key'
        S3_URL = 'http://%s.s3.amazonaws.com/' % AWS_STORAGE_BUCKET_NAME
        MEDIA_URL = S3_URL
```

>> 3. Restart your web server.

You should be good to go! To test, create a new Information Resource in your installation and upload a file. Now
go back to check out your S3 bucket through the AWS console. Your file should show up in a new folder called files
within the bucket. If you are encountering issues, be sure to let us know on the forum.

8.2 Converting to `.arches` / `.relations`

8.2.1 From MS Excel Workbook

It's fairly common for legacy data to be stored in an Excel workbook. Or, perhaps it will be easier for you to aggregate
multiple existing data sources into a single workbook. Although MS Excel is not the friendliest program to work with,
we'd like to provide the following steps to assist with this specific conversion process. (If you are using an MS Access
or MySQL database, the query in the Using SQL section should be useful.)

Using SQL

One way to convert your data is to use tools already included in MS Excel. To show you the basic steps of this process,
we're going to take a simple spreadsheet–with columns for resource ID, resource name, and spatial coordinates–and
turn it into a `.arches` file. When you are ready to add attributes and make more complex resources, you should be

able to work from these building blocks. Be sure to read up on the `.arches` format in the Arches Documentation before continuing.

Note: Most of the action here takes place in MS Excel, which you already have, but the final step will be achieved with Notepad++–a lightweight but super powerful text editor. Download and install it now or whenever you get to the later steps. In fact, we highly recommend it for all of your other Arches work as well!

#1. Standardize Your Data

For this example, we're using a very basic spreadsheet that looks like this:

id	name	nametype	lat	long
1	119 Sidney	NAME_TYPE:1	31.75357	-93.08813
2	135 Sidney	NAME_TYPE:1	31.75359	-93.08825

Later steps depend on these specific column names, so if you are following along, make sure that yours match (case-sensitive). Also, make sure that your original lat/long coordinates use the WGS84 datum. If you need a refresher on what NAME_TYPE:1 means, be sure to go check out all of the available documentation on Controlled Vocabularies.

#2. Run a SQL Query in MS Excel

Once you have your data in columns and headers that match the example above, *save your workbook*. Now you're ready to to use a SQL query to convert it to the `.arches` format.

1. Open MS Excel (2010), click on the **Data** tab, click **From Other Sources**, and choose **From Microsoft Query**.

2. For Data Source, choose **Excel Files** and click **OK**.

3. Navigate to the workbook with your data (even if you already have it open), select it, and click **OK**.

4. Select the sheet that your table is in, and click **>** to add all of the columns from that sheet.

5. Click **Next** three times, until you are given the option to "View data or edit query in Microsoft Query". Select this option, and click **OK**. You should see, in the Microsoft Query window, a reproduction of your original table. Now let's modify it.

6. Click the ▣ **SQL** button, and you'll be presented with the simple SQL query that is being used to create the currently visible table. Don't remove it just yet. Paste the following text in below it:

```
SELECT
    id AS RESOURCEID,
    'HERITAGE_RESOURCE.E18' AS RESOURCETYPE,
    'NAME.E41' AS ATTRIBUTENAME,
    name AS ATTRIBUTEVALUE,
    'group1' AS GROUPID
FROM `your\path\here\workbook.xlsx`.`Sheet1$` `Sheet1$`
UNION ALL
SELECT
    id,
    'HERITAGE_RESOURCE.E18',
    'NAME_TYPE.E55',
    nametype,
    'group1'
FROM `your\path\here\workbook.xlsx`.`Sheet1$` `Sheet1$`
UNION ALL
SELECT
    id,
    'HERITAGE_RESOURCE.E18',
    'SPATIAL_GEOMETRY_COORDINATES.E47',
```

```
        'POINT (' & long & ' ' & lat &')',
        'group2'
FROM `your\path\here\workbook.xlsx`.`Sheet1$` `Sheet1$`
ORDER BY RESOURCEID, GROUPID;
```

7. As you can see, you need to enter the path to your own workbook and reference the correct sheet. To do this, copy the original `FROM` statement, and use it to replace the three new `FROM` statements. Now delete the entire original query, so that all you have is the modified code from above.

8. **Click OK. If you get errors, troubleshoot by double-checking the following:**

   - Are your field names exactly the same as in the example table above?

   - Did you update the path to your own workbook and reference the correct sheet?

   - Did you delete the entire original SQL query?

If you have followed all the steps correctly, you should be greeted by a table that looks like this:

RESOUR-CEID	RESOURCETYPE	ATTRIBUTENAME	ATTRIBUTEVALUE	GROUPID
1	HER-ITAGE_RESOURCE.E18	NAME.E41	119 Sidney	group1
1	HER-ITAGE_RESOURCE.E18	NAME_TYPE.E55	NAME_TYPE:1	group1
1	HER-ITAGE_RESOURCE.E18	SPA-TIAL_COORDINATES_GEOMETRY.E47	POINT (-93.08813 3E475357)	group2
2	HER-ITAGE_RESOURCE.E18	NAME.E41	135 Sidney	group1
2	HER-ITAGE_RESOURCE.E18	NAME_TYPE.E55	NAME_TYPE:1	group1
2	HER-ITAGE_RESOURCE.E18	SPA-TIAL_COORDINATES_GEOMETRY.E47	POINT (-93.08825 3E475359)	group2

Look familiar? That is your original data, reformatted to match the `.arches` schema. As you can see, three rows have been made from each row in our original table. In the SQL query, each new row is defined by a separate `SELECT` statement. You may notice that a little bit of processing was done in the third `SELECT` statement to rewrite your lat/long coordinates–they are now formatted as WKT, or Well-known text.

#3. Export and Touch-up

Now that we've reformatted the table, we're ready to export it and make the final changes.

1. Click the ![button] button (return data), and choose a blank sheet in your workbook for the output. (Your data may now be styled as a table, which is fine but not necessary.)

2. Go to **File > Save As...** and for "Save as type:" choose **Text (Tab delimited) (*.txt)**.

3. Click **Save**, and when you encounter warnings, choose **Yes**.

4. For the next steps, we're using Notepad++, as mentioned above. We need to convert the tabs to pipe characters (`|`). Open your newly created `.txt` file in Notepad++ and use **ctrl + F** to open the Find/Replace window.

5. Go the the Replace tab, and enter "Find what:" = `\t`, and "Replace with:" = `|`. Make sure that you have checked the box for **Extended** Search Mode, and click **Replace All**. If everything worked correctly, click **OK**.

6. For one final check, open the **Encoding** menu, and make sure that "Encode in UTF-8 without BOM" is selected (select it if it isn't).

7. Now go to **File > Save As...**, choose this type (at the top of the list): **All types (*.*)**, and name your file `<yourname>.arches`. Click **Save**.

#4. Make a `.relations` **File**

You must have a `.relations` file to accompany your `.arches` file when you load it into Arches. However, the file need not contain any more than this single line:

```
RESOURCEID_FROM|RESOURCEID_TO|START_DATE|END_DATE|RELATION_TYPE|NOTES
```

If you'd like to define some resource-to-resource relationships between the resources in your `.arches` file, you would do so in the `.relations` file. You can learn more about this in the Arches Documentation.

To make a `.relations` file with Notepad++, just open a new blank document, paste in the line shown above, and save it as `<yourname>.relations`. Your `.arches` and `.relations` files must have the same name (e.g., `buildings.arches` and `buildings.relations`), and should always be kept together.

Congratulations! You've successfully changed a sheet from a standard MS Excel workbook into a `.arches`/`.relations` file, and you're ready to upload it to Arches. You can find guidance on how to do so in the Loading Data section above.

Appendix A

Resource graphs and nodes.

9.1 Activity.E7

Label	mergenode	businesstable
ACTIVITY_TYPE_AUTHORITY_DOCUMENT.E32	ARCHES_EVENT_PHASE.E4	strings
ACTIVITY_TYPE.E55	ARCHES_EVENT_PHASE.E4	domains
ACTIVITY.E7	ACTIVITY.E7	
ADDRESS_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
ADDRESS_TYPE.E55	PLACE.E53	domains
ADMINISTRATIVE_SUBDIVISION_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
ADMINISTRATIVE_SUBDIVISION_TYPE.E55	PLACE.E53	domains
ADMINISTRATIVE_SUBDIVISION.E48	PLACE.E53	strings
ARCHES_ACTIVITY_PHASE.E4	ACTIVITY.E7	
BEGINNING_OF_EXISTENCE_TIME-SPAN.E52	ACTIVITY.E7	
BEGINNING_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	ACTIVITY.E7	strings
BEGINNING_OF_EXISTENCE_TYPE.E55	ACTIVITY.E7	domains
BEGINNING_OF_EXISTENCE.E63	ACTIVITY.E7	
DESCRIPTION_OF_LOCATION.E62	PLACE.E53	strings
DESCRIPTION_TYPE_AUTHORITY_DOCUMENT.E32	ACTIVITY.E7	strings
DESCRIPTION_TYPE.E55	ACTIVITY.E7	domains
DESCRIPTION.E62	ACTIVITY.E7	strings
END_DATE_OF_EXISTENCE.E49	ACTIVITY.E7	dates
END_OF_EXISTENCE_TIME-SPAN.E52	ACTIVITY.E7	
END_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	ACTIVITY.E7	strings
END_OF_EXISTENCE_TYPE.E55	ACTIVITY.E7	domains
END_OF_EXISTENCE.E64	ACTIVITY.E7	
EXTERNAL_RESOURCE.E1	ACTIVITY.E7	
EXTERNAL_XREF_TYPE_AUTHORITY_DOCUMENT.E32	ACTIVITY.E7	strings
EXTERNAL_XREF_TYPE.E55	ACTIVITY.E7	domains
EXTERNAL_XREF.E42	ACTIVITY.E7	strings
FROM_DATE.E49	ARCHES_EVENT_PHASE.E4	dates
GEOMETRY_QUALIFIER_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
GEOMETRY_QUALIFIER.E55	PLACE.E53	domains
KEYWORD_AUTHORITY_DOCUMENT.E32	ACTIVITY.E7	strings

Label	mergenode	businesstable
KEYWORD.E55	ACTIVITY.E7	domains
NAME_TYPE_AUTHORITY_DOCUMENT.E32	ACTIVITY.E7	strings
NAME_TYPE.E55	ACTIVITY.E7	domains
NAME.E41	ACTIVITY.E7	strings
PHASE_TYPE_ASSIGNMENT.E17	ARCHES_EVENT_PHASE.E4	
PLACE_ADDRESS.E45	PLACE.E53	strings
PLACE_SITE_LOCATION.E53	PLACE.E53	
PLACE.E53	ACTIVITY.E7	
SPATIAL_COORDINATES_GEOMETRY.E47	PLACE.E53	geometries
START_DATE_OF_EXISTENCE.E49	ACTIVITY.E7	dates
TIME-SPAN_PHASE.E52	ARCHES_EVENT_PHASE.E4	
TO_DATE.E49	ARCHES_EVENT_PHASE.E4	dates

9.2 Actor.E39

Label	mergenode	businesstable
ACTOR.E39	ACTOR.E39	
ACTOR_APPELLATION.E82	APPELATION.E41	strings
ACTOR_TYPE.E55	ARCHES_EVENT_PHASE.E4	domains
ACTOR_TYPE_AUTHORITY_DOCUMENT.E32	ARCHES_EVENT_PHASE.E4	strings
ADDRESS_TYPE.E55	PLACE.E53	domains
ADDRESS_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
APPELLATION.E41	ACTOR.E39	
ARCHES_EVENT_PHASE.E4	ACTOR.E39	
BEGINNING_OF_EXISTENCE.E63	ACTOR.E39	
BEGINNING_OF_EXISTENCE_TIME-SPAN.E52	ACTOR.E39	
BEGINNING_OF_EXISTENCE_TYPE.E55	ACTOR.E39	domains
BEGINNING_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	ACTOR.E39	strings
CULTURAL_PERIOD.E55	ARCHES_EVENT_PHASE.E4	domains
CULTURAL_PERIOD_AUTHORITY_DOCUMENT.E32	ARCHES_EVENT_PHASE.E4	strings
DESCRIPTION.E62	ACTOR.E39	strings
DESCRIPTION_OF_LOCATION.E62	PLACE.E53	strings
DESCRIPTION_TYPE.E55	ACTOR.E39	domains
DESCRIPTION_TYPE_AUTHORITY_DOCUMENT.E32	ACTOR.E39	strings
END_DATE_OF_EXISTENCE.E49	ACTOR.E39	dates
END_OF_EXISTENCE.E64	ACTOR.E39	
END_OF_EXISTENCE_TIME-SPAN.E52	ACTOR.E39	
END_OF_EXISTENCE_TYPE.E55	ACTOR.E39	domains
END_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	ACTOR.E39	strings
EPITHET.E82	APPELATION.E41	strings
EXTERNAL_RESOURCE.E1	ACTOR.E39	
EXTERNAL_XREF.E42	ACTOR.E39	strings
EXTERNAL_XREF_TYPE.E55	ACTOR.E39	domains
EXTERNAL_XREF_TYPE_AUTHORITY_DOCUMENT.E32	ACTOR.E39	strings
FROM_DATE.E49	ARCHES_EVENT_PHASE.E4	dates
KEYWORD.E55	ACTOR.E39	domains
KEYWORD_AUTHORITY_DOCUMENT.E32	ACTOR.E39	strings

Table 9.2 – continued from previous page

Label	mergenode	businesstable
NAME_TYPE.E55	APPELATION.E41	domains
NAME_TYPE_AUTHORITY_DOCUMENT.E32	APPELATION.E41	strings
PHASE_TYPE_ASSIGNMENT.E17	ARCHES_EVENT_PHASE.E4	
PLACE.E53	ACTOR.E39	
PLACE_ADDRESS.E45	PLACE.E53	strings
PLACE_SITE_LOCATION.E53	PLACE.E53	
START_DATE_OF_EXISTENCE.E49	ACTOR.E39	dates
TIME-SPAN_PHASE.E52	ARCHES_EVENT_PHASE.E4	
TO_DATE.E49	ARCHES_EVENT_PHASE.E4	dates

9.3 Heritage Resource.E18

Label	mergenode	busine
ADDRESS_TYPE.E55	PLACE.E53	domain
ADDRESS_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
ADMINISTRATIVE_SUBDIVISION.E48	PLACE.E53	strings
ADMINISTRATIVE_SUBDIVISION_TYPE.E55	PLACE.E53	domain
ADMINISTRATIVE_SUBDIVISION_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
ANCILLARY_FEATURE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
ANCILLARY_FEATURE_TYPE.E55	PRODUCTION.E12	domain
BEGINNING_OF_EXISTENCE.E63	HERITAGE_RESOURCE.E18	
BEGINNING_OF_EXISTENCE_TIME-SPAN.E52	HERITAGE_RESOURCE.E18	
BEGINNING_OF_EXISTENCE_TYPE.E55	HERITAGE_RESOURCE.E18	domain
BEGINNING_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
COMPONENT.E18	PRODUCTION.E12	
COMPONENT_PRODUCTION.E12	PRODUCTION.E12	
COMPONENT_TYPE.E55	PRODUCTION.E12	domain
COMPONENT_TYPE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
CONDITION_ASSESSMENT.E14	HERITAGE_RESOURCE.E18	
CONDITION_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	strings
CONDITION_DESCRIPTION.E62	CONDITION_ASSESSMENT.E14	strings
CONDITION_IMAGE.E73	CONDITION_ASSESSMENT.E14	
CONDITION_IMAGE_FILE_PATH.E62	CONDITION_ASSESSMENT.E14	files
CONDITION_IMAGE_THUMBNAIL.E62	CONDITION_ASSESSMENT.E14	files
CONDITION_STATE.E3	CONDITION_ASSESSMENT.E14	
CONDITION_TYPE.E55	CONDITION_ASSESSMENT.E14	domain
CONSTRUCTION_TECHNIQUE.E55	PRODUCTION.E12	domain
CONSTRUCTION_TECHNIQUE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
CULTURAL_PERIOD.E55	PRODUCTION.E12	domain
CULTURAL_PERIOD_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
DATE_CONDITION_ASSESSED.E49	CONDITION_ASSESSMENT.E14	dates
DATE_EVALUATED.E49	HERITAGE_RESOURCE.E18	dates
DESCRIPTION.E62	HERITAGE_RESOURCE.E18	strings
DESCRIPTION_OF_LOCATION.E62	PLACE.E53	strings
DESCRIPTION_TYPE.E55	HERITAGE_RESOURCE.E18	domain
DESCRIPTION_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
DESIGNATION_OR_PROTECTION_FROM_DATE.E49	RIGHT.E30	dates

Table 9.3 – continued from previous page

Label	mergenode	busine
DESIGNATION_OR_PROTECTION_TO_DATE.E49	RIGHT.E30	dates
DESIGN_OR_PROCEDURE.E29	PRODUCTION.E12	
DISTURBANCE_TYPE.E55	CONDITION_ASSESSMENT.E14	domair
DISTURBANCE_TYPE_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	strings
ELIGIBILITY_REQUIREMENT_TYPE.E55	HERITAGE_RESOURCE.E18	domair
ELIGIBILITY_REQUIREMENT_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
END_DATE_OF_EXISTENCE.E49	HERITAGE_RESOURCE.E18	dates
END_OF_EXISTENCE.E64	HERITAGE_RESOURCE.E18	
END_OF_EXISTENCE_TIME-SPAN.E52	HERITAGE_RESOURCE.E18	
END_OF_EXISTENCE_TYPE.E55	HERITAGE_RESOURCE.E18	domair
END_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
EVALUATION_CRITERIA_ASSIGNMENT.E13	HERITAGE_RESOURCE.E18	
EVALUATION_CRITERIA_TYPE.E55	HERITAGE_RESOURCE.E18	domair
EVALUATION_CRITERIA_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
EXTERNAL_RESOURCE.E1	HERITAGE_RESOURCE.E18	
EXTERNAL_XREF.E42	HERITAGE_RESOURCE.E18	strings
EXTERNAL_XREF_TYPE.E55	HERITAGE_RESOURCE.E18	domair
EXTERNAL_XREF_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
FROM_DATE.E49	PRODUCTION.E12	dates
GEOMETRY_QUALIFIER.E55	PLACE.E53	domair
GEOMETRY_QUALIFIER_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
HERITAGE_RESOURCE.E18	HERITAGE_RESOURCE.E18	
HERITAGE_RESOURCE_TYPE.E55	PRODUCTION.E12	domair
HERITAGE_RESOURCE_TYPE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
HERITAGE_RESOURCE_USE_TYPE.E55	PRODUCTION.E12	domair
INTEGRITY_TYPE.E55	HERITAGE_RESOURCE.E18	domair
INTEGRITY_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
KEYWORD.E55	HERITAGE_RESOURCE.E18	domair
KEYWORD_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
MANAGEMENT_RECOMMENDATION.E89	CONDITION_ASSESSMENT.E14	strings
MATERIAL.E57	PRODUCTION.E12	domair
MATERIAL_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
MEASUREMENT_TYPE.E55	HERITAGE_RESOURCE.E18	domair
MEASUREMENT_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
MODIFICATION_DESCRIPTION.E62	HERITAGE_RESOURCE.E18	strings
MODIFICATION_EVENT.E11	HERITAGE_RESOURCE.E18	
MODIFICATION_TYPE.E55	HERITAGE_RESOURCE.E18	domair
MODIFICATION_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
NAME.E41	HERITAGE_RESOURCE.E18	strings
NAME_TYPE.E55	HERITAGE_RESOURCE.E18	domair
NAME_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
PERIOD_OF_SIGNIFICANCE.E49	HERITAGE_RESOURCE.E18	
PERIOD_OF_SIGNIFICANCE_FROM_DATE.E49	HERITAGE_RESOURCE.E18	dates
PERIOD_OF_SIGNIFICANCE_TO_DATE.E49	HERITAGE_RESOURCE.E18	dates
PHASE_TYPE_ASSIGNMENT.E17	PRODUCTION.E12	
PLACE.E53	HERITAGE_RESOURCE.E18	
PLACE_ADDRESS.E45	PLACE.E53	strings
PLACE_APPELLATION_CADASTRAL_REFERENCE.E44	PLACE.E53	strings
PLACE_CADASTRAL_REFERENCE.E53	PLACE.E53	

Table 9.3 – continued from previous page

Label	mergenode	busine
PLACE_SITE_LOCATION.E53	PLACE.E53	
PLACE_THIS_LAND_PARCEL.E53	PLACE.E53	
PRODUCTION.E12	HERITAGE_RESOURCE.E18	
PROTECTION_EVENT.E65	RIGHT.E30	
REASONS.E62	HERITAGE_RESOURCE.E18	strings
RECOMMENDATION_TYPE.E55	CONDITION_ASSESSMENT.E14	domain
RECOMMENDATION_TYPE_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	strings
RESOURCE_CLASSIFICATION_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
RESOURCE_TYPE_CLASSIFICATION.E55	HERITAGE_RESOURCE.E18	domain
RIGHT.E30	HERITAGE_RESOURCE.E18	
SETTING_TYPE.E55	PLACE.E53	domain
SETTING_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
SPATIAL_COORDINATES_GEOMETRY.E47	PLACE.E53	geome
START_DATE_OF_EXISTENCE.E49	HERITAGE_RESOURCE.E18	dates
STATUS.E55	HERITAGE_RESOURCE.E18	domain
STATUS_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
STYLE.E55	PRODUCTION.E12	domain
STYLE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
THREAT_TYPE.E55	CONDITION_ASSESSMENT.E14	domain
THREAT_TYPE_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	strings
TIME-SPAN_PHASE.E52	PRODUCTION.E12	
TIME_SPAN_OF_DESIGNATION_OR_PROTECTION.E52	RIGHT.E30	
TO_DATE.E49	PRODUCTION.E12	dates
TYPE_OF_DESIGNATION_OR_PROTECTION.E55	RIGHT.E30	domain
TYPE_OF_DESIGNATION_OR_PROTECTION_AUTHORITY_DOCUMENT.E32	RIGHT.E30	strings
UNIT_OF_MEASUREMENT.E55	HERITAGE_RESOURCE.E18	domain
UNIT_OF_MEASUREMENT_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE.E18	strings
USE_TYPE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	strings
VALUE_OF_MEASUREMENT.E60	HERITAGE_RESOURCE.E18	strings

9.4 Heritage Resource Group.E27

Label	mergenode	bu
ADDRESS_TYPE.E55	PLACE.E53	do
ADDRESS_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	st
ADMINISTRATIVE_SUBDIVISION.E48	PLACE.E53	st
ADMINISTRATIVE_SUBDIVISION_TYPE.E55	PLACE.E53	do
ADMINISTRATIVE_SUBDIVISION_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	st
ANCILLARY_FEATURE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	st
ANCILLARY_FEATURE_TYPE.E55	PRODUCTION.E12	do
BEGINNING_OF_EXISTENCE.E63	HERITAGE_RESOURCE_GROUP.E27	
BEGINNING_OF_EXISTENCE_TIME-SPAN.E52	HERITAGE_RESOURCE_GROUP.E27	
BEGINNING_OF_EXISTENCE_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
BEGINNING_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
CONDITION_ASSESSMENT.E14	HERITAGE_RESOURCE_GROUP.E27	
CONDITION_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	st
CONDITION_DESCRIPTION.E62	CONDITION_ASSESSMENT.E14	st

Continued o

Table 9.4 – continued from previous page

Label	mergenode	bu
CONDITION_IMAGE.E73	CONDITION_ASSESSMENT.E14	
CONDITION_IMAGE_FILE_PATH.E62	CONDITION_ASSESSMENT.E14	fil
CONDITION_IMAGE_THUMBNAIL.E62	CONDITION_ASSESSMENT.E14	fil
CONDITION_STATE.E3	CONDITION_ASSESSMENT.E14	
CONDITION_TYPE.E55	CONDITION_ASSESSMENT.E14	do
CULTURAL_PERIOD.E55	PRODUCTION.E12	do
CULTURAL_PERIOD_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	st
DATE_CONDITION_ASSESSED.E49	CONDITION_ASSESSMENT.E14	da
DATE_EVALUATED.E49	HERITAGE_RESOURCE_GROUP.E27	da
DESCRIPTION.E62	HERITAGE_RESOURCE_GROUP.E27	st
DESCRIPTION_OF_LOCATION.E62	PLACE.E53	st
DESCRIPTION_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
DESCRIPTION_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
DESIGNATION_OR_PROTECTION_FROM_DATE.E49	RIGHT.E30	da
DESIGNATION_OR_PROTECTION_TO_DATE.E49	RIGHT.E30	da
DISTURBANCE_TYPE.E55	CONDITION_ASSESSMENT.E14	do
DISTURBANCE_TYPE_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	st
ELIGIBILITY_REQUIREMENT_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
ELIGIBILITY_REQUIREMENT_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
END_DATE_OF_EXISTENCE.E49	HERITAGE_RESOURCE_GROUP.E27	st
END_OF_EXISTENCE.E64	HERITAGE_RESOURCE_GROUP.E27	
END_OF_EXISTENCE_TIME-SPAN.E52	HERITAGE_RESOURCE_GROUP.E27	
END_OF_EXISTENCE_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
END_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
EVALUATION_CRITERIA_ASSIGNMENT.E13	HERITAGE_RESOURCE_GROUP.E27	
EVALUATION_CRITERIA_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
EVALUATION_CRITERIA_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
EXTERNAL_RESOURCE.E1	HERITAGE_RESOURCE_GROUP.E27	
EXTERNAL_XREF.E42	HERITAGE_RESOURCE_GROUP.E27	st
EXTERNAL_XREF_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
EXTERNAL_XREF_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
FROM_DATE.E49	PRODUCTION.E12	da
GEOMETRY_QUALIFIER.E55	PLACE.E53	do
GEOMETRY_QUALIFIER_AUTHORITY_DOCUMENT.E32	PLACE.E53	st
HERITAGE_RESOURCE_GROUP.E27	HERITAGE_RESOURCE_GROUP.E27	
HERITAGE_RESOURCE_GROUP_TYPE.E55	PRODUCTION.E12	do
HERITAGE_RESOURCE_GROUP_TYPE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	st
HERITAGE_RESOURCE_GROUP_USE_TYPE.E55	PRODUCTION.E12	do
INTEGRITY_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
INTEGRITY_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
KEYWORD.E55	HERITAGE_RESOURCE_GROUP.E27	do
KEYWORD_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
MANAGEMENT_RECOMMENDATION.E89	CONDITION_ASSESSMENT.E14	st
MEASUREMENT_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
MEASUREMENT_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
NAME.E41	HERITAGE_RESOURCE_GROUP.E27	st
NAME_TYPE.E55	HERITAGE_RESOURCE_GROUP.E27	do
NAME_TYPE_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
PERIOD_OF_SIGNIFICANCE.E52	HERITAGE_RESOURCE_GROUP.E27	

Continued o

Table 9.4 – continued from previous page

Label	mergenode	bu
PERIOD_OF_SIGNIFICANCE_FROM_DATE.E49	HERITAGE_RESOURCE_GROUP.E27	da
PERIOD_OF_SIGNIFICANCE_TO_DATE.E49	HERITAGE_RESOURCE_GROUP.E27	da
PHASE_TYPE_ASSIGNMENT.E17	PRODUCTION.E12	
PLACE.E53	HERITAGE_RESOURCE_GROUP.E27	
PLACE_ADDRESS.E45	PLACE.E53	st
PLACE_APPELLATION_CADASTRAL_REFERENCE.E44	PLACE.E53	st
PLACE_CADASTRAL_REFERENCE.E53	PLACE.E53	
PLACE_SITE_LOCATION.E53	PLACE.E53	
PLACE_THIS_LAND_PARCEL.E53	PLACE.E53	
PRODUCTION.E12	HERITAGE_RESOURCE_GROUP.E27	
PROTECTION_EVENT.E65	RIGHT.E30	
REASONS.E62	HERITAGE_RESOURCE_GROUP.E27	st
RECOMMENDATION_TYPE.E55	CONDITION_ASSESSMENT.E14	do
RECOMMENDATION_TYPE_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	st
RESOURCE_CLASSIFICATION_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
RESOURCE_TYPE_CLASSIFICATION.E55	HERITAGE_RESOURCE_GROUP.E27	do
RIGHT.E30	HERITAGE_RESOURCE_GROUP.E27	
SPATIAL_COORDINATES_GEOMETRY.E47	PLACE.E53	ge
START_DATE_OF_EXISTENCE.E49	HERITAGE_RESOURCE_GROUP.E27	st
STATUS.E55	HERITAGE_RESOURCE_GROUP.E27	do
STATUS_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
THREAT_TYPE.E55	CONDITION_ASSESSMENT.E14	do
THREAT_TYPE_AUTHORITY_DOCUMENT.E32	CONDITION_ASSESSMENT.E14	st
TIME-SPAN_PHASE.E52	PRODUCTION.E12	
TIME_SPAN_OF_DESIGNATION_OR_PROTECTION.E52	RIGHT.E30	
TO_DATE.E49	PRODUCTION.E12	da
TYPE_OF_DESIGNATION_OR_PROTECTION.E55	RIGHT.E30	do
TYPE_OF_DESIGNATION_OR_PROTECTION_AUTHORITY_DOCUMENT.E32	RIGHT.E30	st
UNIT_OF_MEASUREMENT.E55	HERITAGE_RESOURCE_GROUP.E27	do
UNIT_OF_MEASUREMENT_AUTHORITY_DOCUMENT.E32	HERITAGE_RESOURCE_GROUP.E27	st
USE_TYPE_AUTHORITY_DOCUMENT.E32	PRODUCTION.E12	st
VALUE_OF_MEASUREMENT.E60	HERITAGE_RESOURCE_GROUP.E27	st

9.5 Historical Event.E5

Label	mergenode	businesstable
ADDRESS_TYPE.E55	PLACE.E53	domains
ADDRESS_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
ADMINISTRATIVE_SUBDIVISION.E48	PLACE.E53	strings
ADMINISTRATIVE_SUBDIVISION_TYPE.E55	PLACE.E53	domains
ADMINISTRATIVE_SUBDIVISION_TYPE_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
ARCHES_EVENT_PHASE.E4	HISTORICAL_EVENT.E5	
BEGINNING_OF_EXISTENCE.E63	HISTORICAL_EVENT.E5	
BEGINNING_OF_EXISTENCE_TIME-SPAN.E52	HISTORICAL_EVENT.E5	
BEGINNING_OF_EXISTENCE_TYPE.E55	HISTORICAL_EVENT.E5	domains
BEGINNING_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	HISTORICAL_EVENT.E5	strings
CULTURAL_PERIOD.E55	ARCHES_EVENT_PHASE.E4	domains

Table 9.5 – continued from previous page

Label	mergenode	businesstable
CULTURAL_PERIOD_AUTHORITY_DOCUMENT.E32	ARCHES_EVENT_PHASE.E4	strings
DESCRIPTION.E62	HISTORICAL_EVENT.E5	strings
DESCRIPTION_OF_LOCATION.E62	PLACE.E53	strings
DESCRIPTION_TYPE.E55	HISTORICAL_EVENT.E5	domains
DESCRIPTION_TYPE_AUTHORITY_DOCUMENT.E32	HISTORICAL_EVENT.E5	strings
END_DATE_OF_EXISTENCE.E49	HISTORICAL_EVENT.E5	dates
END_OF_EXISTENCE.E64	HISTORICAL_EVENT.E5	
END_OF_EXISTENCE_TIME-SPAN.E52	HISTORICAL_EVENT.E5	
END_OF_EXISTENCE_TYPE.E55	HISTORICAL_EVENT.E5	domains
END_OF_EXISTENCE_TYPE_AUTHORITY_DOCUMENT.E32	HISTORICAL_EVENT.E5	strings
EXTERNAL_RESOURCE.E1	HISTORICAL_EVENT.E5	
EXTERNAL_XREF.E42	HISTORICAL_EVENT.E5	strings
EXTERNAL_XREF_TYPE.E55	HISTORICAL_EVENT.E5	domains
EXTERNAL_XREF_TYPE_AUTHORITY_DOCUMENT.E32	HISTORICAL_EVENT.E5	strings
FROM_DATE.E49	ARCHES_EVENT_PHASE.E4	dates
GEOMETRY_QUALIFIER.E55	PLACE.E53	domains
GEOMETRY_QUALIFIER_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
HISTORICAL_EVENT.E5	HISTORICAL_EVENT.E5	
HISTORICAL_EVENT_TYPE.E55	ARCHES_EVENT_PHASE.E4	domains
HISTORICAL_EVENT_TYPE_AUTHORITY_DOCUMENT.E32	ARCHES_EVENT_PHASE.E4	strings
KEYWORD.E55	HISTORICAL_EVENT.E5	domains
KEYWORD_AUTHORITY_DOCUMENT.E32	HISTORICAL_EVENT.E5	strings
NAME.E41	HISTORICAL_EVENT.E5	strings
NAME_TYPE.E55	HISTORICAL_EVENT.E5	domains
NAME_TYPE_AUTHORITY_DOCUMENT.E32	HISTORICAL_EVENT.E5	strings
PHASE_TYPE_ASSIGNMENT.E17	ARCHES_EVENT_PHASE.E4	
PLACE.E53	HISTORICAL_EVENT.E5	
PLACE_ADDRESS.E45	PLACE.E53	strings
PLACE_SITE_LOCATION.E53	PLACE.E53	
SPATIAL_COORDINATES_GEOMETRY.E47	PLACE.E53	geometries
START_DATE_OF_EXISTENCE.E49	HISTORICAL_EVENT.E5	dates
TIME-SPAN_PHASE.E52	ARCHES_EVENT_PHASE.E4	
TO_DATE.E49	ARCHES_EVENT_PHASE.E4	dates

9.6 Information Resource.E73

Label	mergenode	businessta
CONTRIBUTOR.E39	RESOURCE_CREATION_EVENT.E65	
CONTRIBUTOR_APPELLATION.E82	RESOURCE_CREATION_EVENT.E65	strings
CREATOR.E39	RESOURCE_CREATION_EVENT.E65	
CREATOR_APPELLATION.E82	RESOURCE_CREATION_EVENT.E65	strings
DATE_OF_CREATION.E50	RESOURCE_CREATION_EVENT.E65	dates
DATE_OF_PUBLICATION.E50	INFORMATION_RESOURCE.E73	dates
DATE_OF_UPDATE.E50	RESOURCE_CREATION_EVENT.E65	dates
DESCRIPTION.E62	INFORMATION_RESOURCE.E73	strings
DESCRIPTION_OF_LOCATION.E62	PLACE.E53	strings
DESCRIPTION_TYPE.E55	INFORMATION_RESOURCE.E73	domains

Table 9.6 – continued from previous page

Label	mergenode	businessta
DESCRIPTION_TYPE_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings
EXTERNAL_RESOURCE.E1	INFORMATION_RESOURCE.E73	
EXTERNAL_XREF.E42	INFORMATION_RESOURCE.E73	strings
EXTERNAL_XREF_TYPE.E55	INFORMATION_RESOURCE.E73	domains
EXTERNAL_XREF_TYPE_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings
FILE_PATH.E62	INFORMATION_RESOURCE.E73	files
GEOMETRY_QUALIFIER.E55	PLACE.E53	domains
GEOMETRY_QUALIFIER_AUTHORITY_DOCUMENT.E32	PLACE.E53	strings
IDENTIFIER.E42	INFORMATION_RESOURCE.E73	
IDENTIFIER_TYPE.E55	INFORMATION_RESOURCE.E73	domains
IDENTIFIER_TYPE_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings
INFORMATION_CARRIER.E84	INFORMATION_RESOURCE.E73	
INFORMATION_CARRIER_FORMAT_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings
INFORMATION_CARRIER_FORMAT_TYPE.E55	INFORMATION_RESOURCE.E73	domains
INFORMATION_RESOURCE.E73	INFORMATION_RESOURCE.E73	
INFORMATION_RESOURCE_TYPE.E55	RESOURCE_CREATION_EVENT.E65	domains
INFORMATION_RESOURCE_TYPE_ASSIGNMENT.E17	RESOURCE_CREATION_EVENT.E65	
INFORMATION_RESOURCE_TYPE_AUTHORITY_DOCUMENT.E32	RESOURCE_CREATION_EVENT.E65	strings
KEYWORD.E55	INFORMATION_RESOURCE.E73	domains
KEYWORD_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings
LANGUAGE.E55	INFORMATION_RESOURCE.E73	domains
LANGUAGE_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings
PLACE.E53	INFORMATION_RESOURCE.E73	
PUBLICATION_EVENT.E12	INFORMATION_RESOURCE.E73	
PUBLISHER.E39	INFORMATION_RESOURCE.E73	
PUBLISHER_APPELLATION.E82	INFORMATION_RESOURCE.E73	strings
RESOURCE_CREATION_EVENT.E65	INFORMATION_RESOURCE.E73	
RESOURCE_UPDATE_EVENT.E65	RESOURCE_CREATION_EVENT.E65	
RIGHT_DESCRIPTION.E30	INFORMATION_RESOURCE.E73	strings
RIGHT_TYPE.E55	INFORMATION_RESOURCE.E73	domains
RIGHT_TYPE_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings
SPATIAL_COORDINATES_GEOMETRY.E47	PLACE.E53	geometries
SPATIAL_COVERAGE.E53	PLACE.E53	
TEMPORAL_COVERAGE.E4	INFORMATION_RESOURCE.E73	
TEMPORAL_COVERAGE_TIME-SPAN.E52	INFORMATION_RESOURCE.E73	strings
THUMBNAIL.E62	INFORMATION_RESOURCE.E73	files
TIME-SPAN_PUBLICATION_EVENT.E52	INFORMATION_RESOURCE.E73	
TIME-SPAN_RESOURCE_CREATION_EVENT.E52	RESOURCE_CREATION_EVENT.E65	
TITLE.E41	INFORMATION_RESOURCE.E73	strings
TITLE_TYPE.E55	INFORMATION_RESOURCE.E73	domains
TITLE_TYPE_AUTHORITY_DOCUMENT.E32	INFORMATION_RESOURCE.E73	strings