What is Neatline?
  What makes it different?
Neatline Exhibits: An Overview
  At a glance
  Exhibits overview
  Map exhibits vs. image exhibits
Creating Exhibits
  At a glance
  Title
  URL Slug
  Narrative
  Widgets
  Base Layers
  Default Layer
  Public
Working with Exhibits
  At a glance
  [Exhibit Title]
  Public View
  Exhibit Settings
  Import Omeka Items
  Delete
Neatline Records: An Overview
  At a glance
  Everything is a record
  The record data model
  Neatline records vs. Omeka items
Creating Records
  At a glance
  Manually creating records inside of exhibits
  Connecting a Neatline record with an Omeka item
  Bulk-importing Omeka items
    Defining an item query
    What happens if you add new items?
Adding Text Content
  At a glance
  Slug (optional)
  Title (optional, recommended)
  Body (optional)
Drawing Geometry on the Map
  At a glance
  Navigate
  Draw Point
  Draw Line
  Draw Polygon
Neatline Manual

Draw Regular Polygon
Editing Vector Geometry
   At a glance
   Modify Shape
   Rotate Shape
   Resize Shape
   Drag Shape
   Delete Shape
   Clear all Geometry
Importing SVG Documents
   At a glance
   Draw SVG
      Generating the SVG document
      Importing the SVG into Neatline
Editing Record Groups
   At a glance
   Tags
      Tag requirements
   Presenter
Editing Record Colors
   At a glance
   Examples:
      Fill Color
      Fill Color (Selected)
      Stroke Color
      Stroke Color (Selected)
Editing Record Opacities
   At a glance
   Examples:
      Fill Opacity
      Fill Opacity (Selected)
      Stroke Opacity
      Stroke Opacity (Selected)
Editing Record Dimensions
   At a glance
   Stroke Width
   Point Radius
   Z-Index
      Order / Weight (Experimental, subject to change)
Editing Record Dates
   At a glance
   Examples:
      Start Date
      End Date
      After Date
      Before Date
Editing Record Imagery

taken from http://docs.neatline.org/
What is Neatline?

In the broadest sense, Neatline is an annotation framework that makes it possible to create rich, interactive editions of visual objects. In the past, the project has focused on maps, but Neatline can also be used to annotate anything that has some sort of visual instantiation - the same set of vector-drawing and content management tools can be used to create interpretive views of paintings, drawings, photographs, documents, diagrams, and anything else that can be captured as an image.

Here are some examples of the kinds of projects that could be built (or have been built!) with Neatline:

- Wordsworth in the Alps: In Book Sixth of The Prelude, "Cambridge and the Alps," Wordsworth describes his 1790 grand tour of Europe, culminating in the famous description of the crossing of Simplon Pass in the Alps. You want to create an interactive edition of the poem that traces Wordsworth's journey from Cambridge across the channel to Calais, south through France, into the Alps, and along the Stockalper trail over the pass. How does Wordsworth's description of the sequence of towns and landmarks map onto the actual geography of the area?

- The Declaration of Independence: The signatures at the bottom of the Declaration of Independence are iconic - we can all recognize John Hancock and Thomas Jefferson, but what about the rest? You want to create an interactive edition of the document that traces out visual annotations around each of the signatures that connects it to a short biography - who they were, where they came from, and how they ended up signing the
Neatline Manual

declaration. With the signatories in place, you then provide a transcription of the declaration itself, and link the individual sentences or words in the text to the corresponding locations on the high-resolution scan of the original document.

- Minard's Napoleon Infographic: Charles Minard's 1869 diagram showing the gradual depletion of the French army over the course of Napoleon's 1812 invasion of Russia is a seminal work in the history of information design. You want to take high-resolution scan of Minard's original graphic, overlay it on top of modern satellite geography, and layer on an interactive reimagination of the original material - trace out the components of Minard's flowchart, add background information about each of the cities and rivers marked along the route, and link individual objects on the map to paragraphs and sentences in a narrative that describes the history of the invasion.

- Whitman's "Salut au Monde": "Salut au Monde" is an index of Whitman's geographic imagination, an expansive catalog of cities, countries, regions, landmarks, oceans, rivers, and cultures. You want to create a rich interactive edition of the poem by layering the actual text on top of modern satellite imagery and connecting each of the geospatial references to hand-selected (and at times highly interpretive) focus locations on the map - the "Amazon" to the expansive, dark green rainforests in Brazil, the "northern blasts" to the glaciers in the Brooks Range, etc.

- Hotchkiss at Chancellorsville: Jedediah Hotchkiss, a military cartographer for the Confederacy during the Civil War, spent the rest of his life revisiting maps he made during the war, sketching in marginalia and marking off troop locations. In one instance, he printed a series of three identical engravings of the area around Chancellorsville and Fredericksburg and sketched in the positions of the Union and Confederate armies on each of the three days of the Battle of Chancellorsville with colored pencils. You want to create an exhibit that positions the three maps on top of modern satellite imagery of the battle field and layers on a second layer of visual annotation about the maps - the movement of the armies in the weeks leading up to the battle, Lee's risky decision to split his army and leave half of his force behind at Fredericksburg, and why Jackson decided to perform a risky flanking maneuver on the morning of the first day.

- The Chelyabinsk Meteor: On February 15, 2013, a 10,000-ton meteor streaked over the city of Chelyabinsk in the Ural Mountains, producing a massive airburst that shattered windows and collapsed buildings. The explosion occurred during the morning rush hour, and the event was captured by dozens of commuters with dashboard-mounted video cameras. You want to create an exhibit that traces out the trajectory of the meteor as it enters the atmosphere and traverses the airspace over the city, and plot out the exact position and orientation of each of the recordings relative to the location of the airburst.

What makes it different?

How does Neatline fit into the existing ecosystem of geospatial and annotation tools? Neatline sits in the space between consumer-grade mapping applications like the Google custom map-maker and heavyweight, analytical GIS tools like ArcGIS. It's designed to be simple enough that college undergraduates can easily use it for class assignments, but also sufficiently flexible, scalable, and feature-rich that it can be used for professional scholarship, journalism, and art.

taken from http://docs.neatline.org/
Neatline Manual

Neatline keeps things simple and provides sensible defaults.

Neatline Exhibits: An Overview

At a glance

- An exhibit is the "canvas" for a Neatline project, the environment in which the content is displayed.
- Exhibit-level settings define basic, high-level defaults for the project (title, base layers, prose narrative, etc).
- Each exhibit has its own editing environment, where you can create a collection of records (the contents of the exhibit).
- Likewise, each exhibit has its a separate page on the public-facing version of the website.
- Each installation of Neatline can contain an indefinite number of exhibits.

Exhibits overview

Each Neatline project has its own exhibit, which can be thought of as the "canvas" or "environment" for the project - the set of base maps or images on top of which the content is displayed, the text narrative that introduces or describes the project, and the combination of user-interface widgets and components that are enabled for the project (timelines, lists of waypoints, etc).

These high-level configuration options are defined using the "Exhibit Settings" form, which is used to both create new exhibits and edit existing exhibits. Once an exhibit has been created and configured, you can add content to the exhibit by opening the editing environment for the exhibit (generally just called "the editor"), a specialized, interactive map-making application that makes it possible to create and manipulate a collection of records, which make up the actual content of the exhibit.

Most of the work happens in the editor. You might spend just a couple minutes in the "Exhibit Settings" form, and a couple months building out the content in the editing environment.

Map exhibits vs. image exhibits

Exhibits can be grouped into two basic categories depending on the type of base layer used as the visual foundation for the project:

1. Map-based exhibits, which are built on top of geospatial layers - OpenStreetMap, the Google API layers, Stamen Design layers, or completely custom tile sets delivered as WMS layers, MBTiles, or any other layer format supported by the OpenLayers mapping library. Out of the box, Neatline comes with a core collection of general-purpose spatial layers, and makes it easy to add custom layers if you need something different.

taken from http://docs.neatline.org/
2. Image-based exhibits, which can be built on top of any static, web-accessible image (.jpg, .png, etc). This makes it possible to use Neatline to create interactive editions of paintings, drawings, photographs, documents, and anything else that has some kind of two-dimensional, visual instantiation.

Exhibits built using regular, static images are easy to set up and don't require any additional server infrastructure, but there's a fundamental limitation - since the entire image has to be loaded in bulk into the in-browser application (as compared to the spatial layers, which are loaded dynamically depending on the focus and zoom of the map), the performance of the exhibit will get worse as the image gets larger. Things work well if you're working with a more or less normally-sized image (up to around 2-3000 pixels in height/width), but after that things start to get unacceptably sluggish.

If you need to use a really high-resolution image, there's an effective workaround that involves essentially tricking Neatline into thinking the image is a spatial layer - you can "faux-georeference" the static image (just assign it random, meaningless spatial coordinates), load it into Geoserver as a WMS layer, and then import it into an exhibit as the sole base layer, with no map underneath. The image will be presented just as if you were using a regular static file, but you'll have all the scalability that comes with a dedicated tile server.

Creating Exhibits

At a glance

- Click "Create an Exhibit" to add a new exhibit.
- Use the form to define basic, high-level information about a project.
- You can always go back and edit any of the default settings.

To get started with a new project, click on the "Create an Exhibit" button at the top of the main "Browse Exhibits" page:

Title

A top-level identifier for the exhibit, used to label the project in listings of exhibits on the public site.

URL Slug

A string of letters, numbers, and hyphens used to form the last part of the
public-facing URL for the exhibit. For example, if your site is at http://www.neatline-site.org and the URL slug for an exhibit is wordsworth-in-the-alps, the URL for the public-facing version of the exhibit would be http://www.neatline-site.org/neatline/show/wordsworth-in-the-alps.

Tip: As a convenience, the URL slug field is actually linked to the "Title" field, since you'll often want the slug to "resemble" the title - as you type text into the "Title" input, a corresponding value will be automatically generated and inserted into the "Slug" field. For example, if you type "Wordsworth in the Alps" for the title, the slug will automatically read wordsworth-in-the-alps. If you want to change the automatic slug (for example, if the title for the exhibit is really long, and you want something shorter for the URL), just manually edit text in the slug field. Once you manually change the automatic value, Neatline assumes that you want to use the value that you entered, and the slug will stop auto-generating if you go back later and make a change to the "Title."

Narrative

A piece of text to introduce, narrate, accompany, or otherwise generally support the exhibit. This could be anything from a short blurb up to a long-format scholarly essay or a monograph. How you use this field will often depend on how the site theme lays out the public-facing exhibit views - if the narrative is positioned below the exhibit, you might just want a snippet of introductory content; if it's placed to the side of the exhibit and given more room, it might make sense to add more content. In practice, it often make sense to modify the theme according to the needs of the project - if you need a lot of supporting text, build the theme to accommodate it.

Tip: The "Narrative" field integrates tightly with the NeatlineNarrative widget, which makes it possible to connect individual paragraphs, sentences, and words in the "Narrative" content with records in the exhibit. For example, if "Narrative" contained markup like this:

... <span data-neatline-slug="paris">Paris</span> ...

And you had a record in your Neatline exhibit with a slug of paris, the word "Paris" narrative would be interactively linked with the location on the map, and vice versa.

Widgets

The combination of user-interface widgets that are "activated" for the exhibit. Since widgets are installed as separate "sub-plugins" in Omeka, there won't be any options available for selection by default. If you've installed the NeatlineWaypoints and NeatlineSimile sub-plugins, though, you'll see two options in the menu that appears when you click the field: "SIMILE Timeline" and "Waypoints." Just click
one or both of the options (or neither), and a block will appear in the box, indicating that the widget has been enabled for the exhibit.

**Base Layers**

The base layers that are available in the exhibit. All of these layers will be accessible by way of the "layer switcher" tool displayed in the top right corner of the map in the editing environment and public view. Like with the "Widgets" field, click on the input to display a list of layers, any of which can be clicked and added to the list of selections.

**Default Layer**

The default base layer that's displayed when an exhibit is first accessed. This is an important selection, since the default layer will often strongly influence the visual aesthetic of the exhibit.

**Public**

If left unchecked, the exhibit will be visible in the Omeka administrative interface, but not on the public site. Check the box to publish the exhibit to the web.

---

**Working with Exhibits**

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
Neatline Manual

At a glance

- In the "Browse Exhibits" view, click an exhibit's title to open the Neatline editor.
- Click Public View to open the public-facing version of the exhibit visible to anonymous users.
- Click Exhibit Settings to edit the default values entered when you created the exhibit.
- Click Import Omeka Items to bulk-import Omeka items into the exhibit.
- Click Delete to permanently remove the exhibit.

Once you've created one or more exhibits, the "Browse Exhibits" view makes it possible to find, filter, edit, and display them. A couple of options are available for each of the exhibit listings:

[Exhibit Title]

Click on the exhibit title to open up the editing environment for the exhibit, which makes it possible to create the collection of "records" that constitutes the exhibit's content (see the "Neatline Records: An Overview" guide for more information).

Public View

Click here to view the the exhibit as it appears on the public-facing version of the website.

Exhibit Settings

Click here to re-display the form for the basic configuration settings that were defined when the exhibit was created (eg, "Title," "URL Slug," "Base Layers"). These values can be editing at any point after the exhibit has been created.

Import Omeka Items

Click here to open the form that makes it possible to populate the exhibit with content by bulk-importing Omeka items. For more information, see the "Creating Records" guide.

Delete

Click here to permanently delete the exhibit. You'll get a pop-up message asking for confirmation. Important: When an exhibit is deleted, it is completely removed from the database - along with all the records that belonged to it - and cannot be recovered.

Neatline Records: An Overview

At a glance

- All pieces of content in a Neatline exhibit are records.
- Records can appear in multiple contexts (map, timeline, etc).
- Optionally, records can be associated with Omeka items.

Everything is a record

taken from http://docs.neatline.org/
Neatline Manual

Records are the fundamental unit of content in Neatline exhibits. In Neatline, everything is a record - vector annotations on the map, plottings on the timeline, listings in the waypoints tray, WMS overlay layers, text annotations in the exhibit narrative, and any other content displayed by sub-plugins. Depending on the type of information that's entered into the record metadata, the same record could be displayed more than once in the same exhibit. For example, if a record has both a geometric annotation value in the coverage field and a defined value for the start_date, it will be displayed both as a shape on the map and a point on the timeline. When this is the case, Neatline will automatically link all instantiations of the record - when the user clicks on the span on the timeline, the map will zoom to the corresponding location, and vice versa.

The record data model

All records share a common data model. Similar to elements on Omeka items, all fields are optional.

- **Title**: A top-level, human-readable identifier. Used as a label for the record. Eg: "War and Peace"
- **Slug**: A plain-text ID for the record, used to reference the record from TEI or HTML. Eg: war-and-peace
- **Body**: The main content of the record. Could be a short blurb, a long-format essay, a video, etc.
- **Coverage**: A WKT string that defines a geometric annotation on the map (point, line, polygon, etc).
- **Tags**: A string of comma-delimited tags used to slice and dice the collection into related subgroups.
- **Presenter**: The mechanism by which the record content is displayed (static bubble, floating bubble, etc).
- **Widgets**: Which of the "viewports" the record is visible in (timeline, waypoints browser, etc).
- **Fill Color**: The color of polygons on the map. Also used as the default color in other contexts.
- **Fill Color (Selected)**: The color of map polygons when the record is highlighted or selected.
- **Stroke Color**: The color of the lines running around the edges of polygons on the map.
- **Stroke Color (Selected)**: The color of the lines when the record is highlighted or selected.
- **Fill Opacity**: The opacity (translucency) of the polygons on the map.
- **Fill Opacity (Selected)**: The opacity of map polygons when the record is highlighted or selected.
- **Stroke Opacity**: The opacity of lines running around the edges of polygons on the map.
- **Stroke Opacity (Selected)**: The lines opacity when the record is highlighted or selected.
- **Stroke Width**: The thickness, in pixels, of the lines around polygons.
- **Point Radius**: The size of individual points on the map.
- **Z-Index**: The "stacking" order of records when displayed on the map.
- **Order / Weight**: Used to determine the display order of a record in relation to other records.
- **Start Date**: The beginning of the event that the record describes.
- **End Date**: The end of the event that the record describes.

taken from http://docs.neatline.org/
Neatline Manual

- After Date: The date after which the record should be displayed in the exhibit.
- Before Date: The date before which the record should be displayed in the exhibit.
- Point Image: A web-accessible image used to display individual points on the map.
- WMS Address: The location of a WMS server (e.g., an installation of Geoserver).
- WMS Layers: A comma-delimited collection of layers (hosted on the WMS server) to be overlayed on the map.
- Min Zoom: The map zoom level above which the record is visible (where zooming "in" is "higher").
- Max Zoom: The map zoom level below which the record is visible.
- Default Focus: The lat/lon coordinates that the map zooms to when the record is selected.
- Default Zoom: The zoom level that the map zooms to when the record is selected.

Neatline records vs. Omeka items

Even though all records share the same data model, they can be divided into two basic categories - exhibit-specific records that exist just inside of a single exhibit, and item-backed records that link back to items in the Omeka collection. The two types behave exactly the same, with one exception - if a Neatline record is associated with an Omeka item, the record's "Title" and "Body" fields will be automatically compiled from the content in the Omeka item. The title will be filled in with the item's Dublin Core "Title" field, and the "Body" will be populated with the compiled metadata output of the entire item. Once the association is established, the content in the Neatline record will be automatically updated whenever the Omeka record is changed.

Why does Neatline make this distinction? Why aren't records just the same thing as items? In the early stages of development, there actually was a one-to-one correspondence between Omeka items and Neatline records - records were just the direct instantiations of Omeka items in a specific Neatline exhibit. The problem, though, is that Neatline exhibits often require a body of "annotative" or "supporting" information that doesn't really fit well in the context of an archival collection. For example, imagine you're working with a collection of correspondence - each letter in the collection is represented by a canonical Dublin Core record in Omeka. Then, when you pull the letters into a Neatline exhibit, you start to sketch in arrows, brackets, flowcharts, and other little bits of presentational information to support the core archival objects.

This supporting information is essential in the context of the exhibit, but almost meaningless in isolation, and it often doesn't make sense to formalize all of these elements as first-class archival entities - it's not the best practice to have items like "Blue Arrow 4" or "Outline of New York" mixed heterogeneously into the the collection of letters. To get around this problem, Neatline makes it possible to create these exhibit-specific, "unaffiliated" records can be used to formalize this type of annotative information, while also making it possible to create records that link back to the durable archival resources in the Omeka collection. This way, we get the best of both worlds - we can integrate seamlessly with the content in Omeka, and also choose not to integrate with it when doing so would degrade the integrity of the collection.

Creating Records

taken from http://docs.neatline.org/
Neatline Manual

At a glance

- Records can be created directly inside of a Neatline exhibit.
- Existing Neatline records can be individually linked to (and unlinked from) Omeka items at any point.
- Omeka items can be bulk-imported into Neatline exhibits.

Manually creating records inside of exhibits

The simplest way to add a new record is just to create one directly inside of the Neatline editor.

1. Open the editor for an exhibit by clicking on the exhibit's title in the main browse view.
2. Click the "New Record" link just below the search bar.

This will open up an empty editing form for the new record. Important: Keep in mind that the record won't actually be saved to the database until you click the "Save" button at the bottom of the form.

Connecting a Neatline record with an Omeka item

Once a record has been created in an exhibit, it can be "linked" back to an item in the Omeka collection:

1. Open the editing form for the record that you want to tie to an Omeka item (or click "New Record" if you want to start from scratch).

2. In the "Omeka ID" field, start typing the title (or any text that appears in any of the other elements) of the Omeka item that you want to associate the record with. As you type, a list of matching search results will autocomplete below the input.

3. Click on the name of the item that you want to link the record with. As soon as you click on the listing, the "Omeka ID" field will be populated with the items' id, and the title will be inserted into the "Title" field.

4. Click the "Save" button at the bottom of the form. The "Body" field of the record will be populated by the compiled metadata output of the parent item (similar to the content that's displayed on the default item "show" pages in Omeka).

Once the association is established, the Neatline record will update automatically whenever the Omeka item is changed. For example, if you changed the title of the record, or added a new file upload, the "Title" and "Body" fields would automatically reflect the new information when you view the Neatline exhibit.

Important: Once you've associated a record to an Omeka item, you can't make custom changes to the "Title" or "Body" fields. For example, if you change the title to something else, and then saved the record, the new value would be overriden by the Dublin Core "Title" field on the item. This is a deliberate restriction - if you find yourself wanting to tweak the content of the record -

taken from http://docs.neatline.org/
but don't want to change the elements directly on the parent item - it may be an indication that the record shouldn't be associated with an item, and should just be an unaffiliated record in the exhibit.

That said, it's quite possible that you might want to change the structure or layout of the compiled metadata output in the "Body" field. For example, imagine that you're working with items that represent photographs - in the Neatline exhibit, you might want to hide some of the more nitty-gritty element texts on the items (e.g., "Date" or "Published"), and just show the title, thumbnail, and description. To address this, Neatline makes it possible to completely customize the structure of the compiled "Body" fields at the level of the theme in the same way that you create a custom show.php template for the regular item pages. See "Theming Exhibits" guide for detailed instructions about how to do this.

**Bulk-importing Omeka items**

So far, we've manually created individual Neatline records and associated them with items in the Omeka collection. But what if you already have an existing collection of hundreds or thousands of items? Instead of manually creating Neatline records for each of the items individually, you can just bulk-import part or all of the collection into a Neatline exhibit:

1. Go to the main exhibits browse page and find the listing for the exhibit.
2. Click the "Import Omeka Items" link under the exhibit title.

**Defining an item query**

This takes you to the item import form. Think of this as a "search" form - you can use any of the available inputs to define a "query" on the Omeka archive that determines which items will be imported into the exhibit. For example, if you have a large archive with multiple collections, you might just be interested in working with one individual collection, and don't want to clutter up the Neatline exhibit with all of the other unrelated items. The fields here work just like the corresponding options in the Omeka advanced search form:

- Use Search by a Range of ID#s to specify an individual ID (not so useful) or a range of id's (more useful), and all items with ID's that fall within the specified range will be imported. This is a good way to import the entire collection all at once - just enter in an indiscriminate query like "1-1000," which, as long as you have fewer than 1,000 items in your collection, will import all the items on the site. This is fine for experimentation, but in the long run, we generally recommend slicing and dicing the items in to groups with collections or tags, which often prevents content management problems down the road.
- Use Search By Collection to import items in a given collection.
- Use Search By Type to import items of a given type
- Use Search By Tags to constrain the import to items that are tagged with all of the listed tags.

(Keep in mind that the fields are AND'ed together, not OR'ed - so, if you select a collection and enter a tag, the import will only match items that are both in the collection and have the tag.)

Once you've defined a search query, click "Import Items" to kick off the import. You'll be taken
Neatline Manual

back to the exhibits browse page, and you'll see a success notification saying that "The item import was successfully started!"

Important: Behind the scenes, this actually kicks off a "background process" that does the heavy lifting of importing the items. This is necessary because it can sometimes take up to 30-40 seconds to import really large collections of items (many thousands), and the process can fail if the web request times out (smaller imports, up to about 1,000 items, will generally finish in just a couple of seconds). When you're first redirected to the exhibits browse view, though, the "# Items" counter for the exhibit will probably still be the same as it was before, since the import was started at the same moment that you were redirected. Refresh the page, though, and you'll see the effect of the background process as it fills in the items.

What happens if you add new items?

When the import is finished, open the editor for the exhibit. You'll see new listings for all of the records that were matched by the import. When you open the edit form for one of the records, you'll see that the "Omeka ID" field is populated with the ID of a corresponding Omeka item and that the "Title" and "Body" fields are populated with the item title and metadata output. Just as if the records had been manually linked to their parent items, any change to the items will be propagated to the imported records.

But what if you then continue to add new Omeka items to the collection that would have been matched by the original import, and want to synchronize the Neatline exhibit with the new collection of items? Neatline expects this, and makes it easy - if you go back to the "Import Omeka Items" form for the exhibit, you'll see that the original query has been saved, making it possible to re-run the identical query over and over again. When you run the query, Neatline will never duplicate an existing item-backed record in an exhibit, meaning that only the newly-added items will be imported.

For example, imagine you import a collection that has 100 items. Later, you add another 20 items to the collection. If you want to vacuum up those new items into the Neatline exhibit, you can just re-run the same import query, and Neatline will only import the 20 new records and ignore the other 100 items that already have corresponding records in the collection.

Adding Text Content

At a glance

- Text content for records is entered in the "Title" and "Body" fields in the "Text" tab.
- The title should be short, and generally shouldn't contain complex HTML or formatting.
- The body can be anything from a short blurb up to a long-format narrative essay.
- Any kind of HTML-embeddable media can be put in the body - audio, video, etc.
- Use the "Edit HTML" links to open a fullscreen WYSWYG editor for either of the fields.
- All text fields are optional, but it's recommended to at least enter a title.

Slug (optional)

The slug is a plain-text identifier for the record

taken from http://docs.neatline.org/
Neatline Manual

- think of it as a human-readable ID that can be used to reference the record from without. Specifically, the Neatline Narrative plugin - which makes it possible to connect text documents to Neatline exhibits - makes it possible to connect individual paragraphs, sentences, or words in a text with individual Neatline records. For example, in the exhibit "Narrative" field, you might have something like this:

... <span data-neatline-slug="paris">Paris</span> ...

If you had a record in your Neatline exhibit with a slug of paris, the word "Paris" document would be interactively linked with the location on the map. The slug isn't actually displayed anywhere in the public-facing version of the exhibit - if you're not using the Narratives plugin, it fine to just leave it blank.

**Title (optional, recommended)**

The title is the top-level snippet of text used to label the record in pop-up bubbles, on timelines, and in vertical listings of waypoints. Try to keep it short and simple, and avoid using complex formatting. A title isn't strictly required, but if you leave it blank you'll likely end up with some odd visual artifacts in your exhibit (empty preview bubbles, unlabelled points and spans on the timeline, etc).

If you want to add special formatting to the title (eg, make it bold, increase the size), click the "Edit HTML" link next to the field label. This will open a full-screen WYSWYG editor with a full load-out of formatting options. When you're done with the editor, click the Minimize button to return to the regular editing interface.

**Body (optional)**

The body is a catch-all field designed to house the main textual content for the record. Depending on what kind of content you're working with and how you're structuring the user-interface for the exhibit, it might make sense to enter anything from a short little blurb all the way up to a 10,000-word scholarly essay.

Like with the title, click on the "Edit HTML" link to open up a full-screen WYSWYG editor for the field. Although it's generally a good idea to keep the formatting pretty minimal for the title, it fine to use much more elaborate and structural styling in the body - bullets and numbered lists, images, tables, etc.

taken from http://docs.neatline.org/
Neatline Manual

Also, you can use the body as a space to include any kind of embeddable third-party multimedia content - videos from YouTube or Vimeo, audio from SoundCloud, etc. Just click on the "Source" button at the top left of the editor to activate view-HTML mode, and paste in the embed code.

**NOTE:** If you use the “Edit HTML” WYSIWYG, you do **not** save & close out by clicking the icon that looks like a floppy disk. You **do** save & close out by clicking the icon on the 3rd row which looks like a “fullscreen” button (with 4 arrows pointing at the corners)

---

The Rotunda is a building located on The Lawn in the original grounds of the University of Virginia. It was designed by Thomas Jefferson to represent the “authority of the Pantheon in Rome. Construction began in 1822 and was completed in 1826, after his death. The grounds of the new university were unique in that they surrounded common at other universities in the English-speaking world. The Rotunda is seen as a lasting symbol of Jefferson’s belief in the separation of church and education, and architecture.

The collegiate structure, the immediate area around it, and Jefferson’s nearby home at Monticello combine to form one of only three modern man-made sites in the United States to be a UNESCO World Heritage Site (the other two are the Statue of Liberty and Independence Hall).

The original construction cost of the Rotunda was $67,775 ($998,792 in 2006 dollars). The building stands 77 feet (23.5 m) in both height and diameter.

(source http://en.wikipedia.org/wiki/The_Rotunda_(University_of_Virginia))

---

### Drawing Geometry on the Map

**At a glance**

- The drawing controls under the "Map" tab make it possible to add vector annotations to the map.
- There are three basic types of geometry - points, lines, and polygons.
- The built-in tools are best suited for simple shapes that don't require large numbers of points.
- Activate the different drawing modes by selecting the corresponding radio buttons.
- When you're done drawing, switch back to the "Navigate" mode to avoid accidentally drawing more shapes.

**Navigate**

Select the "Navigate" radio button to enable the default panning interaction with the map - click and drag to move the map in one direction or another, and use the scroll wheel on your mouse to zoom in and out.

Tip: Think of Navigate as your "home base" - after sketching in new geometry with one of the drawing tools, it's usually easiest to reactivate "Navigate" mode, which has the effect of turning off any of the drawing or modification tools. This means that you don't accidentally add unwanted...
Draw Point

Use this control to lay down individual points on the map.

1. Select the "Draw Point" radio button, move the cursor to the location on the map where you want to place the point, and click down once to drop the point.

2. To add multiple points, just click multiple times in different places.

Important: Points are displayed as circles on the map, but, in the underlying data, they're actually represented as dimensionless pairs of lat/lon coordinates. Since they're infinitely small and have no measurable "size" that can be correlated to the scale of the map, the points will always stay the same size, regardless of the zoom level of the map (the size of the circles can be controlled by the "Point Radius" field in the "Style" tab).

Depending on the situation, this can be good or bad. For example, if you're using points in conjunction with the "Point Image" feature to plot thumbnails of photographs onto the map, you might want the points to stay the same size at different zoom levels so that the preview images never become too big or too small. If you want a real geometric circle that changes size with the zoom level, use the "Draw Regular Polygon" tool.

Draw Line

Just like the "Draw Point" radio button, except that the points are connected by a line. To draw a line:

1. Select the "Draw Line" tool, move your cursor to the place on the map where you want the line to start, and click down once. Now, when you move the mouse, a line will connect the cursor to the
Neatline Manual

location of the first point.

2. Click again to lay a second point. Keep on clicking to add multiple points. You can also draw a smooth line that automatically follows the movement of the cursor. Hold down the Shift key and then click and drag on the map. This lays down a new point for each individual pixel movement registered by the browser.

Tip: This feature has the effect of creating extremely dense geometry, which is very difficult to maintain (e.g., if you decide later on that you want to change the shape of the line, you would have to manually move hundreds or thousands of individual points). If you need smooth, curvy shapes, always see if you can use the "Draw SVG" feature, which makes it possible to import easily-maintained vector graphics created in programs like Adobe Illustrator and Inkscape.

1. When you're ready to lay down the final point, move your cursor to the place where you want the line to end and double click. This "completes" the line.

Draw Polygon

Just like "Draw Line," except that the shape defined by the points is "closed" into a polygon as soon as you put down more than two points. To draw a polygon:

1. Select the "Draw Polygon" radio button, move your cursor to the place on the map where you want to place the starting "corner" of the polygon, and click down once.

2. Click again to lay a second point. Keep on clicking until to add multiple points. Like with the point tool, you can draw smooth shapes by holding down Shift, clicking down on the map, and dragging the cursor.

3. When you're ready to lay down the final point, move your cursor to the place where you want the last corner of the shape and double click. This "closes" the polygon.

Draw Regular Polygon

"Draw Regular Polygon" creates closed shapes just like the regular "Draw Polygon" tools, but it doesn't let you manually position the individual points - instead, it automatically creates a polygon with a given number of sides. This is useful if you want to create consistently-shaped geometric primitives (triangles, squares, circles, etc.), and it can also be a good way to sketch in basic components of more complex shapes:

1. Select the "Draw Polygon" radio button and move your cursor to the place on the map where you want to position the center of the polygon.

2. Click down and, while still holding down on the mouse button, move the cursor in any
direction away from the center point. As you drag the mouse, a polygon will be dynamically rendered on the map with a radius equal to the distance between the cursor and the center point. Rotate the cursor around the center point to change the orientation of the polygon.

3. When the polygon is positioned correctly, release the mouse button to lock the shape in place.

There are a couple of configuration options for the regular polygon tool:

- **Sides**: The number of sides on the polygon. For example, 3 will draw triangles, 4 squares, etc. As this number gets higher, the resulting shape will smooth out into a circle/ellipse. This is a good way to create circular shapes with "real" diameters that will expand and shrink with the zoom level of the map (as opposed to the circles used to represent points, which are "dimensionless" and always stay the same size - see the "Draw Point" section). Don't go overboard with the number of sides - in most cases, 30-50 sides is more than enough to create a smooth shape.

- **Snap Angle**: As you drag the cursor away from the center point, the rotation of the polygon will "snap" into a series of fixed angles. This makes it possible to create a series of shapes with the exact same orientation on the map (eg, a series of rectangles with sides that are all perfectly aligned with the cardinal directions). This field controls the size of the angle between each of the consecutive rotations.

For example, if you set a snap angle of 30 degrees, it will only be possible to orient the shape in a couple of different ways; as you decrease the angle, the rotation becomes and more granular, and at 0 the snapping is completely disabled.

- **Irregular**: By default, all of the sides of the polygon are equal in length (eg, a 3-sided shape is always an equilateral triangle, a 4-sided shape is always a square, etc). Check this box to make it possible to create "skewed" shapes with sides of different lengths - rectangles instead of squares, ellipses.
Neatline Manual

instead of circles.

Tip: This setting can have strange interactions with certain "Snap Angle" values - if you need to create an irregular shape, it's generally easiest to set the snap angle to 0, drag out the shape, and then rotate it as needed with the "Rotate Shape" tool.

Editing Vector Geometry

At a glance

- The editing controls under the "Map" tab make it possible to modify existing geometries on the map.
- Annotations can be reshaped, rotated, resized, dragged, and deleted.
- Activate the different editing modes by selecting the corresponding radio buttons.
- When you're done editing, switch back to the "Navigate" mode to avoid accidentally editing other shapes.
- Use the "Clear all Geometry" button to delete all of the geometries for the record.

Modify Shape

Use this mode to change the shape of lines or polygons created with any of the drawing modes. It can also be used to change the position of stand-alone points.

1. Select the "Modify Shape" radio button and click on the shape that you want to modify. Once the shape is selected, you should see a set of circular drag handles appear on top of each of the vertices in the shape.

2. To change the position of an existing vertex, just click down on the point's drag handle, move the cursor to a different location, and release.
Neatline Manual

the mouse button to lock in the new location.

3. To add new vertices to the shape, click and drag on any of the translucent points that are displayed in the middle of each of the sides of the shape. This has the effect of bisecting the vertex - as soon as you release the mouse button, the drag handle will be converted into a regular point, and you can then continue to progressively add detail to the shape by re-bisecting the new edges.

Rotate Shape

Rotate an existing line or polygon.

1. Select the "Rotate Shape" radio button and click on the shape that you want to resize. Once the shape is selected, you should see a new "control point" appear to the bottom right of the shape.

2. Click and drag on the control point to rotate the shape.

Resize Shape

Expand or shrink an existing line or polygon.

1. Select the "Resize Shape" radio button and click on the shape that you want to resize. Once the shape is selected, you should see a new "control point" appear to the bottom right of the shape.

2. Click and drag on the control point to scale the shape.

Tip: If you want to change the size of the circle used to represent an individual point, head over to the "Style" tab and change the "Point
Neatline Manual

Drag Shape

Change the location of an existing point, line, or polygon.

1. Select the "Drag Shape" radio button and click on the shape that you want to drag. You should see a new "control point" appear right in the middle of the shape. If you're dragging an individual point, the point will just change color to indicate that it has been highlighted.

2. Click and drag on the control point to change the center position of the shape.

Delete Shape

Delete an existing point, line, or polygon.

1. Select the "Delete Shape" radio button.

2. To delete a shape, just click once on the shape, and it will immediately be removed from the map.

Clear all Geometry

This button is similar to "Delete Shape," but it delete all vector annotations for the record. This can be useful when you've made a mistake of some sort (for example, when experimenting with different ways of positioning an imported SVG document) and want to just completely clear out your work.

Importing SVG Documents

At a glance

- The basic drawing controls (points, lines, polygons) aren't suitable for really complex illustrations.
- Instead, it's better to create the graphics in specialized vector editors and import them into Neatline.
- Neatline can import SVG, a common format that can be generated by programs like Adobe Illustrator.
- Once a SVG document has been imported, the geometry has all the same characteristics as hand-drawn shapes.

Draw SVG

The standard "Draw Point/Line/Polygon" tools are good for simple, diagrammatic shapes - dots,
rectangles, circles, simple arrows, etc. The problem, though, is that they don't really scale well - what if you want to create a really smooth, intricate shape? Neatline exhibits are always exercises in information design, and there's a class of "illustrative" annotation that doesn't fit well into the blocky aesthetic supported by the standard-issue GIS tools. You could painstakingly sketch out complex illustrations using the line and polygon tools, but the resulting geometry is brittle and difficult to maintain. What if you decide later on that you want to change the drawing? You'd have to manually reposition hundreds or thousands of points.

To address this problem, Neatline 2.0 introduced the "Draw SVG" tool, which makes it possible to create high-fidelity, easily-maintained illustrations in specialized vector editing programs like Adobe Illustrator or Inkscape (and even in-browser tools like SVG-Edit) and import them directly into the Neatline editor. This way, you can do the difficult work of vector illustration in tools that are specifically designed for that task, and then drop the final products into Neatline exhibits. Later in, if you want to change something about the drawing, you can just go back to the original source file, make the change, and then update the Neatline exhibit with the new content.

Neatline reads a commonly-used serialization format called SVG (Scalable Vector Graphics), a form of XML that encodes geometric information. To start, you'll need to save off your vector file as an SVG document.

Generating the SVG document

Using Adobe Illustrator:

1. Click File > Save As.
2. In the "Format" drop-down menu, select "SVG (svg)." The "SVG Options" modal should be displayed.
3. In the "Fonts" fieldset, set "Type" to Convert to outline. This ensures that means text elements in the document will be saved as raw geometric paths, not abstract <text> elements that can't be parsed by Neatline.
4. Click OK to save.

Using Inkscape:

1. Before saving, if you have any text elements in the document, first convert them to raw path elements by selecting them and entering Ctrl+Shift+C (see the documentation for more information.)
2. Click File > Save As.
3. In the dropdown menu for the format, select Plain SVG.
4. Click Save.

taken from http://docs.neatline.org/
Importing the SVG into Neatline

Once the SVG document is saved off, we can import it into Neatline:

1. Find the .svg file that was created in the previous step. Open it with a plain text editor or a code editor by right clicking on the file and choosing a program from the list of options in "Open With." It doesn't matter what you use - as long as you can get at the raw SVG output.

2. You should see a soup of dense XML markup with lots of numbers. Copy the content of the entire document to the clipboard.

3. Go back to the Neatline editor and click on the Enter Markup link next to the Draw SVG option.

4. Paste the SVG markup into the "SVG" box.

5. Optionally, set a custom "Density" value. This field controls the number of points that Neatline will generate when converting SVG<path> elements to hard-coded spatial coordinates - low numbers produce "blocky" shapes, high numbers produce "smooth" shapes.

6. Click Parse. Behind the scenes, Neatline converts the SVG markup into geospatial coordinates.

7. Now, you can drag the SVG document onto the map in the same way that you would drag out a regular polygon shape. Move the cursor to the location on the map where you want the bottom left corner of the illustration to be location, click down, and move the cursor in any direction. The illustration will scale and rotate according to the position of the mouse.

8. When the illustration is positioned correctly, release the mouse button to lock the geometry into place.

Tip: It's easy to make mistakes when doing this, and it often takes a couple tries to get things right. If you mess up, it's easiest just to totally wipe out the previous attempt by clicking the "Clear all Geometry" button at the bottom of the form.

taken from http://docs.neatline.org/
Neatline Manual

Editing Record Groups

At a glance

- Use tags to slice and dice large collections into subsets of related records.
- Tags can be used as the "selectors" for Neatline's stylesheet system, which uses a simplified dialect of CSS.
- Each record is assigned to a "Presenter," which controls how the record's content is displayed.

Tags

The tags field takes a comma-delimited list of one or more tags. For example:

- virginia
- precinct, democrat, 2004
- precinct, republican, 2012
- confederate, may_3
- union, may_4

Neatline's tagging system provides a powerful and flexible way to slice and dice records into groups of related subsets. This can be useful for a number of reasons:

- Tags provide the "selectors" that makes it possible for the Neatline styling system to apply bulk updates to large collections of records. For example, you could do something like:


```.
.virginia {
    fill-color: #08c;
}
```

This automatically updates the "Fill Color" of all records in the exhibit tagged with virginia, even if there are hundreds of thousands of them. For more information about this, head over to the "Styling Exhibits with Neatline CSS" guide.

- You can filter the list of records in the content management pane in the editor with a query syntax. For example, you could search for:

```
tags: precinct, democrat
```

And just show the records that are tagged with both precinct and democrat.

- If you're developing a custom theme or sub-plugin for an exhibit, you can use tags to toggle on and off different portions of the exhibit. For example, imagine you're mapping presidential election results, and you want to add a little widget that makes it possible for the user to check on or off different batches of data from 2000, 2004, etc. If all the data points are tagged to one of the election cycles, you could just run simple API queries like:

```
Neatline.execute('MAP:load', { tags: ['2004', 'democrat'] });
```
Neatline Manual

Tag requirements

In order for the tags to work correctly, there are a couple important requirements:

1. Tags can contain letters, numbers, and underscores (_), but no spaces or hyphens (-). Technically, it's actually possible to use spaces and hyphens in tags, but it's strongly discouraged, because it effectively results in multiple taggings. For example, if you had a tag called democrat-2012, Neatline would consider the record to be tagged with democrat, 2012, and democrat-2012. So, if you were to start using a more generalized tag called just democrat, queries for democrat would match the record, even though it's tagged with the more specific democrat-2013. This is confusing, but, unfortunately, the issue lies at the level of the fulltext searching system in the underlying MySQL database that we use to implement the feature. We're working on a fix for this - for now, if you need "spaces" in the tags, just use underscores (e.g., democrat_2012).

2. Tags must be at least 4 characters in length. Again, this is actually a requirement that's built in to the full text searching system in MySQL. Really, though, this is a healthy constraint - tag names should be semantic and descriptive, so most tags should be longer than three letters anyway.

3. Tags must be separated by commas, or else Neatline won't know where one tag stops and another starts.

Presenter

Each record is assigned to a "presenter," which determines the mechanism by which the record's content is displayed (the "Title" and "Body" fields). Out of the box, Neatline comes with two simple presenters - the "Static Bubble," which displays the pop-up bubbles that appear when the cursor hovers or clicks on a record, and the "None" presenter, which, as you might guess, does nothing (this is useful for purely-visual annotations that don't need any kind of text content - arrows, brackets, etc).

The Neatline plugin API makes it easy for developers to implement completely new presenters, which would then be available for selection in this field. For example, it would be easy to write a presenter that would add a "floating" bubble that would track the location of the cursor on the page, or a presenter that would open up a full-screen modal view. See the developer documentation for more information about creating new presenters.

Editing Record Colors

At a glance

- Color values are stored as hexadecimal values.
- To change a color field, click anywhere on the input and use the interactive color picker widget.
- As you change the color, new value will be automatically previewed on the map.

taken from http://docs.neatline.org/
If you want to use a specific value, just paste the value into the text input below the color picker.

**Examples:**

- #ffffff
- #000000
- #c4d243
- #08c

**Fill Color**

The hexadecimal color value of the "body" or "area" of points and polygons on the map (the area inside the lines, not including the lines themselves). Also used as a default color for the record in other contexts. For example, if the record has date information, the fill color is used to render the plotting that represents the record on the timeline.

**Fill Color (Selected)**

The hexadecimal color value of the "body" or "area" of points and polygons on the map when the record is highlighted or selected (eg, when the cursor hovers or clicks on the shape).

**Stroke Color**

The hexadecimal color value of the lines that run around the edges of a shape on the map.

---

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
Stroke Color (Selected)

The hexadecimal color value of the lines that run around the edges of a shape on the map when the record is highlighted or selected (eg, when the cursor hovers or clicks on the shape).

taken from http://docs.neatline.org/
## Editing Record Opacities

### At a glance

- Opacities are stored as 3-digit decimal values between 0.00 (completely translucent) and 1.00 (completely opaque).
- To change an opacity value, you can type directly into the field.
- Or you can click on the field and drag up or down on the page, causing the value to change smoothly.
- As an opacity is changed, the new value will be automatically previewed on the map.

#### Examples:

- 0.00
- 1.00
- 0.56

### Fill Opacity

The opacity of the "body" or "area" of points and polygons on the map (the area inside the lines,

[http://docs.neatline.org/](http://docs.neatline.org/)
Neatline Manual

not including the lines themselves).

<table>
<thead>
<tr>
<th>Opacities</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Fill Opacity</td>
<td>0.48</td>
</tr>
<tr>
<td>Fill Opacity (Selected)</td>
<td>0.4</td>
</tr>
<tr>
<td>Stroke Opacity</td>
<td>0.9</td>
</tr>
<tr>
<td>Stroke Opacity (Selected)</td>
<td>1</td>
</tr>
</tbody>
</table>

**Fill Opacity (Selected)**

The opacity of the "body" or "area" of points and polygons on the map when the record is highlighted or selected (eg, when the cursor hovers or clicks on the shape).

<table>
<thead>
<tr>
<th>Opacities</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Fill Opacity</td>
<td>0.48</td>
</tr>
<tr>
<td>Fill Opacity (Selected)</td>
<td>0.82</td>
</tr>
<tr>
<td>Stroke Opacity</td>
<td>0.9</td>
</tr>
<tr>
<td>Stroke Opacity (Selected)</td>
<td>1</td>
</tr>
</tbody>
</table>

**Stroke Opacity**

The opacity of the lines that run around the edges of a shape on the map.

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
Stroke Opacity (Selected)

The opacity of the lines that run around the edges of a shape on the map when the record is highlighted or selected (e.g., when the cursor hovers or clicks on the shape).

<table>
<thead>
<tr>
<th>Opacities</th>
<th>Stroke Opacity (Selected)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Fill Opacity</td>
<td>0.47</td>
</tr>
<tr>
<td>Fill Opacity (Selected)</td>
<td>0.62</td>
</tr>
<tr>
<td>Stroke Opacity</td>
<td>0.66</td>
</tr>
<tr>
<td>Stroke Opacity (Selected)</td>
<td>0.74</td>
</tr>
</tbody>
</table>

Editing Record Dimensions

taken from http://docs.neatline.org/
Neatline Manual

At a glance

- These fields control the size of different aspects of the presentation of the record.
- And (a) the sequence or ordering of a record in relation to other records in the exhibit.
- Dimension values are stored as natural numbers (integers greater than 0).
- To change a dimension, you can type directly into the field.
- Or you can click on the field and drag up or down on the page, causing the value to change smoothly.
- When the "Stroke Width" and "Point Radius" fields are changed, the new value will automatically preview on the map.

Stroke Width

The width, in pixels, of the lines that run around the edges of a shape on the map. Generally, it makes sense to keep this value in the range of 0-10 - anything more and the lines will start to "clump" together and blot out the shape of the polygon, especially when the map is zoomed back away from the vectors.

In some cases, it can be aesthetically pleasing to completely hide the lines, and just display the "area" of points and polygons on the map. To do this, just set the stroke width to 0.

Point Radius

The radius of individual points on the map. You'll almost always want this value to be greater than 0 - otherwise, points will be invisible, in which case they should probably just be deleted. If you're making use of the "Point Image" field to change the visual representation of points (see

taken from http://docs.neatline.org/
Neatline Manual

the "Editing Record Imagery" guide), this field will also determine the size of the image graphics overlayed on the map.

Important: The point radius is a static value, meaning that the size of the points on the map won't change with the zoom level of the map, as a polygon or line would. This is because points are actually just dimensionless pairs of lat/lon coordinates. Since they're infinitely small and have no measurable "size" that can be correlated to the scale of the map, the points will always stay the same size, regardless of the zoom level of the map.

**Dimensions**

**Stroke Width**

3

**Point Radius**

13

**Z-Index**


**Order / Weight**


**Dates**

**Start Date**


**Z-Index**

The "stacking" order of vector annotations and WMS layers on the map. For example, imagine you have one record that includes a polygon that traces the shape of Manhattan, and another record that traces the shape of Central Park. Since Central Park is contained inside of Manhattan, you'll want the Central Park polygon to be displayed "on top of" the Manhattan polygon - otherwise, it would be impossible for the user to click on Central Park.

Z-indexes are just relative numbers that control the display order, with higher numbers stacking above lower numbers. So, in this case, you could set the Z-Index on Manhattan to 1, and the Z-Index on Central Park to 2, and Central Park would always be displayed on top of Manhattan.

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
Order / Weight (Experimental, subject to change)

The display order of a record relative to other records. This is similar to the "Z-Index," but designed to be used in situations where records need to be put into a specific, unambiguous sequence (i.e., where no two records can be in the same position in the sequence, as opposed to stacking order, in which multiple records can exist on the same "layer" or "level"). For example, this field is automatically populated by the Neatline Waypoints plugin to store the display order of records in the waypoints panel, a list of clickable record titles displayed on the map.

Important: For now, this is an abstract field that has no direct effect on out-of-the-box Neatline exhibits. It's included in the current version to make it possible for sub-plugins to store ordering information about records. Unless you're using a sub-plugin that makes use of the field, it usually makes sense just to leave it blank.
Editing Record Dates

At a glance

- If a record represents an event, these fields store when the event occurred and how long it lasted.
- They can also be used to control when a record is visible in the exhibit.
- Important: All dates must be entered in a portable, standards-compliant format called ISO 8601.
- To display temporal information, you'll need to install a widget like NeatlineSimile.

Examples:

CE Dates

- 1564 - The year 1564.
- 1564-04 - April, 1564.
- 1564-04-23 - April 23, 1564.
- 1564-04-23T08:30 - 8:30 AM on April 23, 1564.
- 1564-04-23T08:30:15 - 15 seconds
Neatline Manual

after 8:30 AM on April 23, 1564.

BCE Dates

- -001563 - The year 1564 BCE.
- -000563 - The year 564 BCE.
- -000063 - The year 64 BCE.
- -000003 - The year 4 BCE.
- -001563-04 - April, 1564 BCE.
- -001563-04-23 - April 23, 1564 BCE.

Start Date

The "first" or "beginning" date for an event. If the event is an instant (it happened exactly once, and did not occupy any kind of duration or interval of time), enter the date of the instant in this field.

If you're working with Neatline Simile: If you just enter a "Start Date" and leave "End Date" blank, the record will be displayed as point on the timeline.

End Date

The "last" or "finishing" date for an event.

If you're working with Neatline Simile: If you enter a "Start Date" and an "End Date" date, the record will be displayed as a span (line) on the timeline.

If you enter the same date for the "Start Date" and "End Date," the span will have width = 0 and will, therefore, be invisible on the timeline.

After Date

The date after which the record should be displayed in the exhibit.

If you're working with Neatline Simile: If an "After Date" is defined, then the record will only be displayed on the map when the timeline is centered on a date that falls after this date. For example, imagine you're working with a record that's plotted as a point on the map. If you set the
taken from http://docs.neatline.org/
"After Date" to 2000, then the point will disappear when you scroll the timeline back to 1999, and reappear when you scroll forward to 2001.

Before Date

The date before which the record should be displayed in the exhibit.

If you're working with Neatline Simile: If a "Before Date" is defined, then the record will only be displayed on the map when the timeline is centered on a date that falls before this date. Again, imagine you're working with a record that's plotted as a point on the map. If you set "Before Date" to 2000, then the point will disappear when you scroll the timeline forward to 2001, and reappear when you scroll back to 1999.

If both a "After Date" and a "Before Date" are defined, then the record will only be visible when the timeline is within the duration between the two dates. For example, if "After Date" is 2000, and "Before Date" is 2010, the record would be visible between those two dates, but not before 2000 or after 2010. By stringing together collections of records with different after- and before-dates, it's possible to create complex time-series animations.

Editing Record Imagery

At a glance

- These fields make it possible to layer images on top of the map base layer - thumbnails, historical maps, etc.
- The "Point Image" field replaces the default circular graphic for points with any web-accessible image.
- The "WMS Address/Layers" fields make it possible to overlay a WMS layer hosted by software like Geoserver.

<table>
<thead>
<tr>
<th>Imagery</th>
</tr>
</thead>
<tbody>
<tr>
<td>Point Image</td>
</tr>
<tr>
<td>WMS Address</td>
</tr>
<tr>
<td>WMS Layers</td>
</tr>
</tbody>
</table>

Point Image

By default, points on the map are represented with circle graphics, the size of which can be configured by the "Point Radius" field in the "Dimensions" field set. This field makes it possible to replace this default representation of the points with any random, web-accessible image. This makes it possible to place effectively any kind of graphic on the map - pins, icons, image thumbnails, etc.

Once a "Point Image" is provided, the points continue to behave just like regular points in all other ways. For example, you can change the size of the point-images by changing the "Point Radius" field, and you can configure the opacities by changing the "Fill Opacity" and "Fill Opacity (Selected)" fields.

Point images can be the web-accessible location of any image. For example, to use the Neatline
Important: Since we're really just changing the cosmetic appearance of the points, there are some restrictions:

- Like regular points, point-images will always stay the same size (the value of "Point Radius"), no matter what the zoom level of the map. If you need an image to scale depending on the zoom level, you'll need to use a WMS overlay hosted by Geoserver (see below).

- Point-images can't be "rotated" - they will always have the exact orientation of the source image. Again, if you find yourself wanting to pivot the image to make it line up with something on the underlying base layer, you're probably looking for the type of functionality provided by a WMS overlay.

Tip: When preparing images for use as point-images, try to use image types like *.png that support transparency. Otherwise, you'll have unsightly white or black boxes filling in the negative space surrounding the graphics.

**WMS Address**

The location of a web-accessible Web Map Service server like Geoserver. This field has to link directly to the WMS API endpoint on the server, such as:

http://libsvr35.lib.virginia.edu/geoserver/hotchkiss/wms

**WMS Layers**

A comma-delimited list of layers on the served defined by "WMS Layers," such as:

hotchkiss:fredericksburg,hotchkiss:chancellorsville

Once a WMS address and layers, Neatline will automatically overlay the requested layers on top of the base layer. WMS layers react to many of the same styling, visibility, ordering settings that can be used to configure the behavior of regular vector annotations:

- Use the "Fill Opacity" field to control the opacity of the WMS overlay.
- If an "After Date" or "Before Date" is defined for the record, the WMS layers will appear and disappear as the timeline is dragged in and out of the visibility interval defined by the date fields. For example, if the "After Date" is set to 2000, the WMS layer will be invisible when the timeline is focused at 1999.
- If a "Min Zoom" or "Max Zoom" is defined for the record, the WMS layer will appear and disappear as the map is zoomed in and out of the visibility interval defined by the zoom fields. For example, if "Min Zoom" is set to 10, the WMS layer will be invisible when the map is at zoom level 9.

**Editing Record Visibility Settings**

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
Neatline Manual

At a glance

- Use "Min Zoom" and "Max Zoom" to hide or display the record depending on the zoom level of the map.
- Use the "Use Current" buttons to insert the current zoom level as the value for "Min/Max Zoom."
- Use "Default Focus/Zoom" to set the location and zoom level that the map "snaps" to when the record is selected.
- Use the "Use Current Viewport as Default" to use the current focus/zoom as the record default.

Visibility

Min Zoom (Use Current)

Max Zoom (Use Current)

Default Focus

Default Zoom

Use Current Viewport as Default

Min Zoom

The zoom level "above" which the record will be visible, with zooming "in" being "higher" (eg, focusing on Spain is "higher" than focusing on Europe). For example, if the record has a point on the map and "Min Zoom" is 10, then the point will be invisible when the map at zoom level 9, and will become visible as soon as the map zooms to 10.

To set the value, just zoom the map the the level that you want to use for the value, and click the "Use Current" button next to the field title to automatically insert the current zoom offset into the input.

Max Zoom

The zoom level "below" which the record will be visible, with zooming "out" being "lower" (eg, focusing on Europe is "lower" than focusing on Spain). For example, if the record has a point on

taken from http://docs.neatline.org/
the map and "Max Zoom" is 10, then the point will be invisible when the map at zoom level 11, and will become visible as soon as the map zooms to 10.

Like with "Min Zoom," click the "Use Current" button next to the field title to automatically insert the current zoom level into the input.

**Default Focus and Default Zoom**

These two fields define the focus location and zoom level that the map "snaps" to when the record is selected. For example, if the record is represented by a vector annotation on the map, and the user clicks on the title of the record in the Waypoints panel, the map will move to the location and zoom level defined by these fields. Think of these fields as a related pair that work together to set the record's "framing" or "home base" in the exhibit.

To set a default focus and zoom, just move the map to the exact location and zoom level that you want to use, and click the "Use Current Viewport as Default" to insert the current values into the inputs.

If no values are provided for these fields, Neatline will fall back on an automatically-computed focus location by centering the map around the geometric extent of the record's vector annotations on the map. This can often work well without any modification, but there are a couple of cases when you might want to explicitly set a custom focus:

- If the record is represented with a single point, Neatline will always zoom in to the highest possible zoom level, since points are effectively dimensionless, infinitely-small dots. This is often undesirable. To fix, just zoom the map back to a reasonable level and click "Use Current Viewport as Default."

- Even when the record is represented by lines or polygons that result in a fairly reasonable automatic focus, there are times when you want to tweak the default viewport, perhaps to situate the record in the context of some other nearby feature on the map. For example, imagine that the record represents the location of a battle that was affected by the presence hill to the north of the battle site. You might want the default focus of the record to be zoomed back a bit and moved up to the north so that the hill appears in the default viewport of the record for the battle.

**Searching for Records**

**At a glance**

- Use the forward and backward arrows to page through the records.
- Enter plain-text queries to search in the "Title," "Body," and "Slug" fields.
- Type tags: <tag1>, <tag2>, ... to search by tags.
- Type map: to make the record list mirror the currently-visible records on the map.
- Clear the search box to show all

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
When there are just a handful of records in an exhibit, it's easy to keep track of everything. But what if there are 1,000 records, or 100,000? In Neatline 2.0, it's possible to work with really large collections of records - as many as about 1,000,000 in a single exhibit. If the exhibit can display that quantity of information, though, there need to be tools to effectively manage information at that scale. To help with this, Neatline provides some useful searching and filtering tools:

Tip: As you use the pagination and searching tools, you'll notice that the URL is dynamically updated to represent the current "query" on the collection. For example, you'll see URL's like:

- /neatline/editor/1#records/search/start=100
- /neatline/editor/1#records/search/query=keywords
- /neatline/editor/1#records/search/query=keywords/start=100

These are all "durable" links, meaning that they'll point back to the same set of results if you copied and pasted into a different browser window.

**Pagination buttons**

To make it easier to browse through the content of an exhibit, Neatline splits apart large collections of records into a series of "pages," each containing a fixed number of records (by default, 50). When you first start building an exhibit, the pagination buttons will be hidden, since there aren't enough records yet to spill over onto a second page. As soon as you go over the paging limit, though, you'll see the forward and backwards buttons to page back and forth through the content.

**Search for keywords**

To find records by regular, plain-text search queries, just enter search terms into the "Search" box at the top of the content management pane in the editor. As you type, Neatline will immediately query the underlying collection and update the list with the set of matching records. Fulltext search looks for the terms in each of the three text-based fields - the "Title," "Body," and "Slug."

When you're finished searching, just empty the search box to go back to complete, unfiltered list of records.

**Search for tags**

If you've added tags to your records (for more information about Neatline's tagging system, see the "Working with Tags" guide), there's a special search syntax you can use to search for records that have a given tag. For example, if you want to find all records that are tagged with tag1, enter:

tags: tag1

To search for records with multiple tags, just enter a comma-delimited list:

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
Search by location

Sometimes, the map itself is the best mechanism for finding records - if you know where the records are that you're interested in, you can enter a special query that will cause the records in the left panel to automatically synchronize with the collection of records that's currently visible on the map. Just enter:

map:

As soon as you enter this query, the previous list of records will be replaced by the exact set of records that are currently visible on the map. When you pan or zoom the viewport on the map and a fresh batch of records is loaded, the list of records in the editing pane will automatically update to display the new collection from the map.

To disable the search-by-location mode, just clear out the search box.

Styling Exhibits with Neatline-Flavored CSS

At a glance

- Neatline 2.0 makes it possible to use a dialect of CSS to perform bulk updates on large groups of records.
- Neatline-infected CSS uses a custom set of rules that correspond to the properties on the record data model.
- Groups of records are selected using the CSS class syntax, with record tags providing the class names.

Quick reference (for the impatient)

Neatline 2.0 makes it possible to use a simplified, Neatline-inflected dialect of CSS - integrated directly into the editing environment - to perform bulk updates on large groups of related records clustered together by tags (see the "Editing Record Groups" guide for more information about tags).

All exhibits have a built-in stylesheet:

1. Open the editing environment for an exhibit.

taken from http://docs.neatline.org/
Neatline Manual

2. Click on the Styles tab. Right under the tab, you’ll see a code editor labelled "Stylesheet.

Neatline CSS is syntactically identical to regular CSS, but semantically a bit different:

- Instead of using DOM selectors, Neatline CSS just treats tags on records as "classes." So, if you have a group of records that are tagged with democrat, you can select them as if "democrat" were a class on an HTML element:

  ```
  .democrat {
    fill-color: #08c;
    stroke-width: 5;
  }
  ```

- To match all records in the exhibit, use the special .all tag:

  ```
  .all {
    fill-color: #08c;
    stroke-width: 5;
  }
  ```

- You can’t match multiple tags with the same selector (we’re hoping to add this soon). For now, this won’t work:

  ```
  .democrat, .republican {
    fill-color: #08c;
    stroke-width: 5;
  }
  ```

- Instead of using regular CSS rules, use a special set of rules that map onto properties in the Neatline data model (and, by extension, to the fields in the "Style" tab in the record edit form):

  - widgets
  - presenter
  - fill-color
  - fill-color-select
  - stroke-color
  - stroke-color-select
  - fill-opacity
  - fill-opacity-select
  - stroke-opacity
  - stroke-opacity-select
  - stroke-width
  - point-radius
  - zindex
Neatline 2.0 makes it possible to work with really large collections of records - as many as about 1,000,000 in a single exhibit. That level of scalability is liberating, but it also introduces some interesting new content management challenges. If the map can display that many records, there also need to be new tools that make it possible to effectively update and maintain content at that scale.

For example, imagine you're plotting returns from the last four presidential election. You import a big collection of about 800,000 records, 200,000 for each of the four elections. Each record represents an individual precinct somewhere in the country with a dot on the map, scaled according to how many votes were cast at the location and shared red or blue depending on which party won more votes. Once the data is loaded into Neatline, you realize that you want to change the shade of blue used to represent the democratic precincts. How do you do that, short of manually making the change on all ~400,000 democratic records?

This is obviously a problem with really massive data sets, but, as you work with Neatline, you'll find that this type of problem rears its head surprisingly quickly, even with quite small exhibits in the 50- to 100-record range. The essence of the problem is this - records are almost never "unique snowflakes" in the exhibit. They almost always exist as part of some kind of general taxonomy or grouping in the exhibit - democratic, 2012,northeast, etc. And, in almost every case, those groupings should share some common attributes. All democratic records should be the same shade of blue; all precincts from 2004 should be visible on the map during the same range of dates; all precinct records should have the same basic opacity settings; ad infinitum.

But, as you can see, they shouldn't share all of their attributes - all 2004 precincts should share the same range of dates (the "After Date" and "Before Date" fields), but they most definitely should not share the "Fill Color" field, since that would clobber out the all-important distinction between the blue democratic records and the red republican precincts. In other words, different groups of records need to share different sets of attributes.

We can map this out schematically. Here's a list of all the "categories" in the exhibit, and the fields that should be shared for each of the categories:

- All Precincts: Fill Opacity, Stroke Width

Rationale: Why use stylesheets?

Neatline Manual
How do we keep the groups synchronized? Really, this is a familiar problem - it's almost exactly the same task as styling web pages with **CSS** (Cascading Style Sheets), a simple styling language that makes it possible to "select" specific parts of a page and apply a set of rules to those elements. In Neatline 2.0, it's possible to use a simplified dialect of CSS - integrated directly into the editing environment - to model these kinds of relationships among records and keep them in sync.

In Neatline, the stylesheet system is closely tied to the tagging system. Tags are just comma-delimited lists of identifiers that can be added to each individual record under the "Style" tab in the record edit form. In this example, individual records in the collection might have tag strings that look like this:

- precinct, democrat, 2000
- precinct, republican, 2000
- precinct, democrat, 2012
- state, republican, 2008

**Getting started with stylesheets**

Let's start by defining some basic, generic styles for the each of the precincts, which are all represented as dots on the map:

```css
.precinct {
    fill-opacity: 0.5;
    stroke-width: 0;
}
```

Now, when I click "Save" at the bottom of the form, Neatline will update the "Fill Opacity" and "Stroke Width" of all 800,000 records tagged as precincts with 0.5 and 0. Next, let's add the date visibility settings for each of the three election-season tags:

```css
.2000 {
    after-date: 2000;
    before-date: 2004;
}

.2004 {
    after-date: 2004;
    before-date: 2008;
}

.2008 {
    after-date: 2008;
    before-date: 2012;
}

.2012 {
    after-date: 2012;
    before-date: 2016;
}
```

taken from [http://docs.neatline.org/](http://docs.neatline.org/)
Likewise, when I click "Save," Neatline will update the "After Date" and "Before Date" fields on each record depending on which of the election-season is assigned to it.

**Auto-updating stylesheet values**

This is all fine and well, but what if we don't actually know what value we want to use? In each of these cases, we're working with fields that have fairly "semantic" values that we can reason about in the abstract (e.g., 2004 just means what it means). This isn't always true, though, notably in the case of colors, where it's impossible to reason in the abstract about which specific hexadecimal value you want to use. For example, I know that I want the democratic precincts to be "blue" and the republican precincts to be "red," but I don't know that I want to use the #206bbf and #9d0000.

You could always just open one of the individual record forms, use the built-in color pickers to find a color that works well, and copy and paste it back into the stylesheet. This is sort of awkward, though. To fix this, Neatline makes it possible to just "activate" a set of styles for a tag in the stylesheet without providing a concrete value, and then set the value for the entire group of tagged records by making a change to an individual record.

We do this with the special auto value:

```css
.democrat {
  fill-color: auto;
}

.republican {
  fill-color: auto;
}
```

Once this is in place, I can just open up any of the individual republican precinct records and pick a shade of red for that specific record. When I click "Save," Neatline recognizes that the "Fill Color" style has been enabled for the republican tag, and that the record being saved is tagged as republican. When this happens, Neatline does two things. First, it update the stylesheet with the new value: