



Tips for Developing and Maintaining Plugins in ArchivesSpace

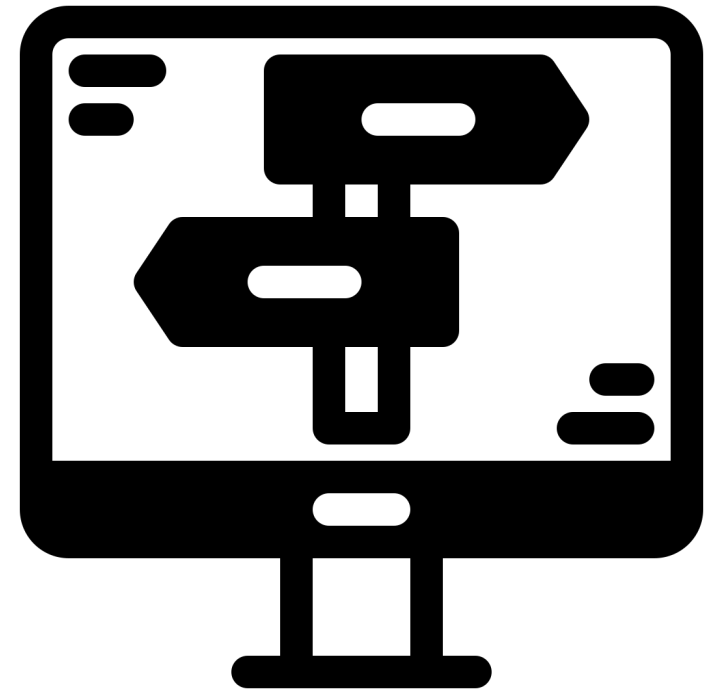
Joshua Shaw

Digital Library Technologies Group

Dartmouth Library, Dartmouth College

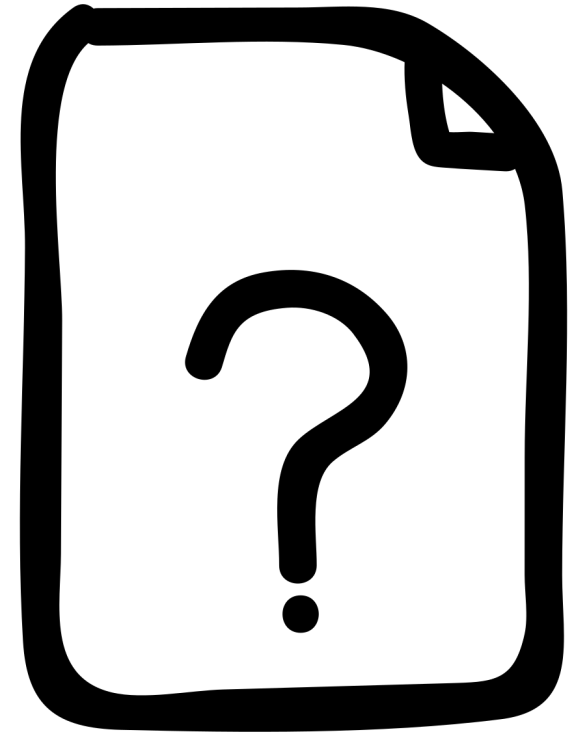
Getting Started – What to think about

- Are you doing something simple that only involves some minor UI changes?
- Are you creating a new subrecord for an existing object type?
- Do you need to create schemas, migrations, models, and controllers or extend or modify existing pieces?
- Do you need to add to the indexer for faceting or other purposes?
- Are you modifying the PUI beyond simple UI changes?



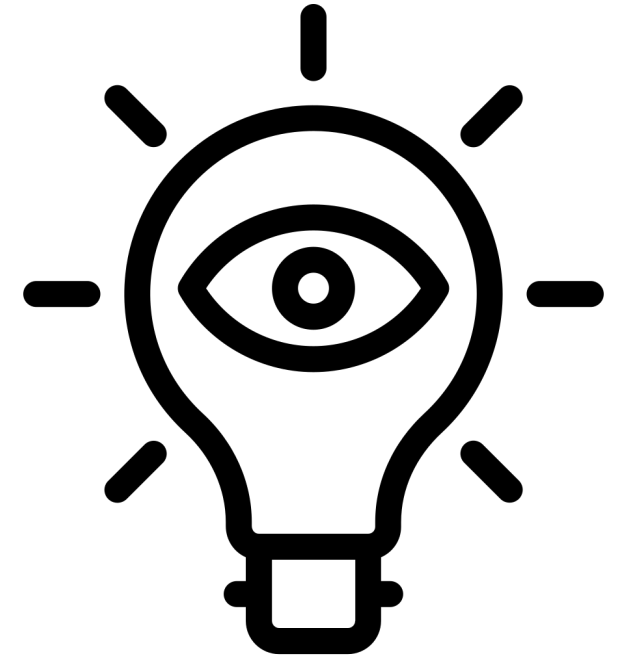
What sort of Plugin is it - examples

- Adding a new subrecord to an accession is an example of a plugin that **ADDS** functionality.
- Changing the EAD export is an example of a plugin that **OVERRIDES** functionality.
- Adding a subrecord to a resource that **ALSO** displays in the PUI would be a plugin that **ADDS** and **OVERRIDES** functionality.



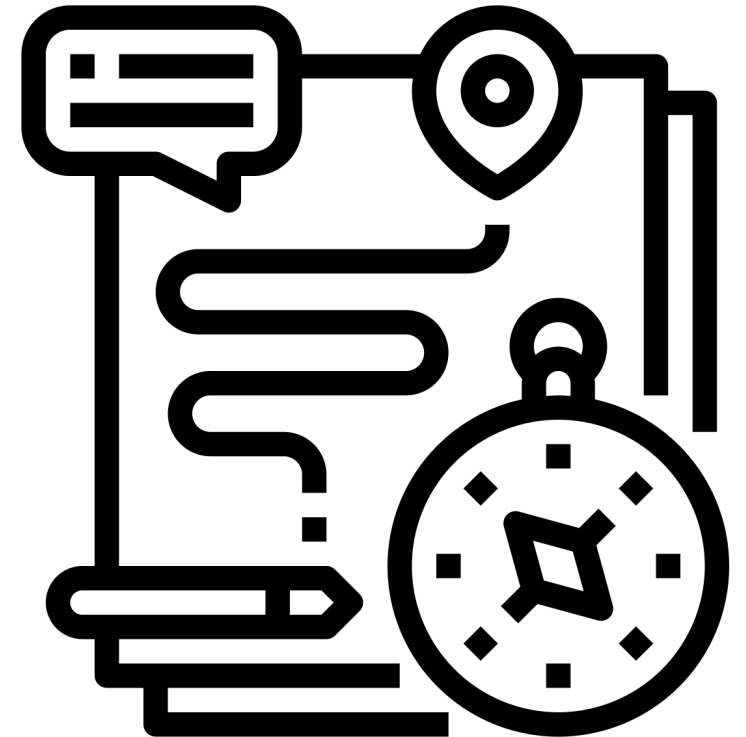
Getting Started - Resources

- I tend to look for examples in existing plugins that do something like what I'm trying to do, but sometimes you've just got to read the core code itself to figure out what to do.
- Tech Docs: <https://archivesspace.github.io/tech-docs/customization/plugins.html>
- Awesome ArchivesSpace: <https://github.com/archivesspace/awesome-archivesspace>
- Hudson-Molonglo: <https://github.com/orgs/hudmol/repositories>
- Queensland State Archives – features a bunch of work from HM: <https://gitlab.gaiaresources.com.au/qsas>
- /plugins in the core repo: <https://github.com/archivesspace/archivesspace/tree/master/plugins>
- Other peer institutions (Harvard, Yale, etc. I usually run a Google search)



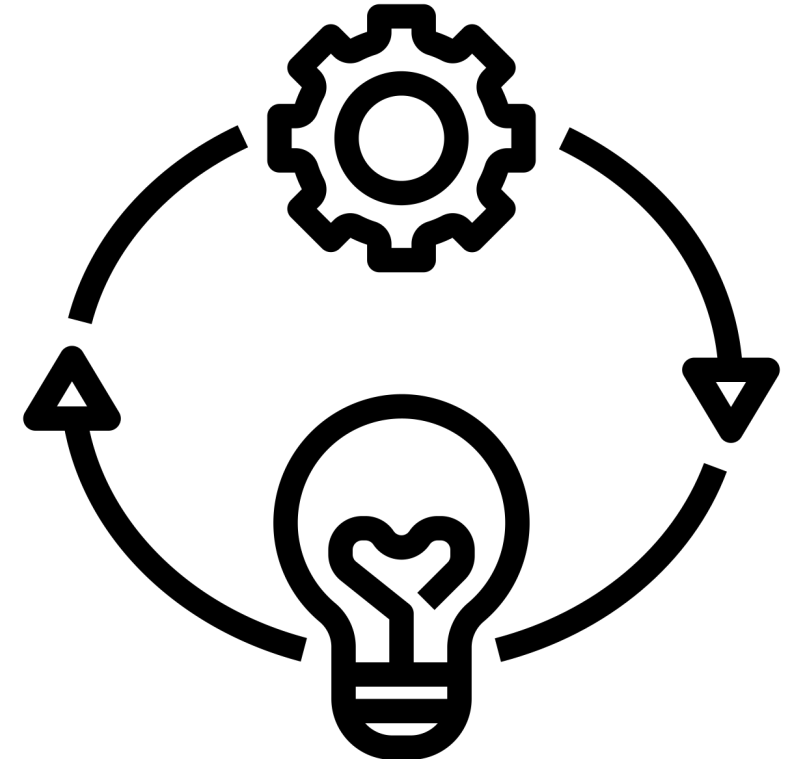
Getting Started - Planning

- Plan out the plugin workflow and figure out what additional data (if any) you will need
- If you need to add data to the database, you're going to need to add migration(s), model(s), and controller(s).
- You'll probably need to add view(s) as well.
- Indexer modifications are typically optional but can be useful for adding new facets or adding information you will need.
- Figure out what you will need to override or extend in the core code.



Getting Started - Implement

- This is where having a model/example plugin is very useful. You'll be able to figure out what pieces you need. It's hard to describe the process in detail, but in general terms.
- Start by adding the schemas, models, and migrations to get the database ready
- Add your controllers and views
- Test! Fix!
- Test! Fix!
- Test! Rinse & Repeat!



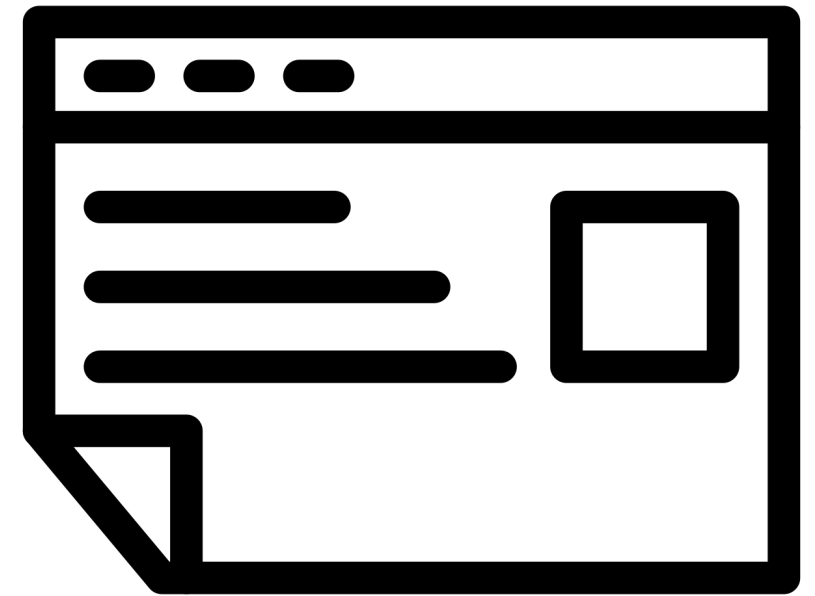
What to check when you upgrade ArchivesSpace

- Thinking about this is going to be very similar to your process when developing the plugin.
- Typically, any schemas, migrations, and models won't need fixing, though you should double check!
- Check the views. Are you overriding any core views? Have they changed?
- Check the `plugin_init`, controllers, and any helpers. Are you overriding any core methods?
- Has any underlying technology changed?



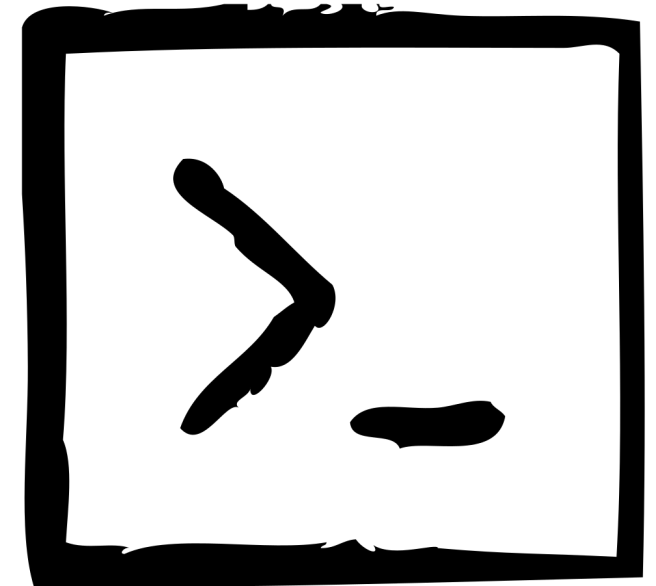
Specifics - Views

- If your plugin is adding subrecords to the staff side only, chance are good you won't need to update your views
- If you are overriding any staff views, make sure to check for any changes in the core code
- If you have added any functionality to the PUI, check the core code for those as well



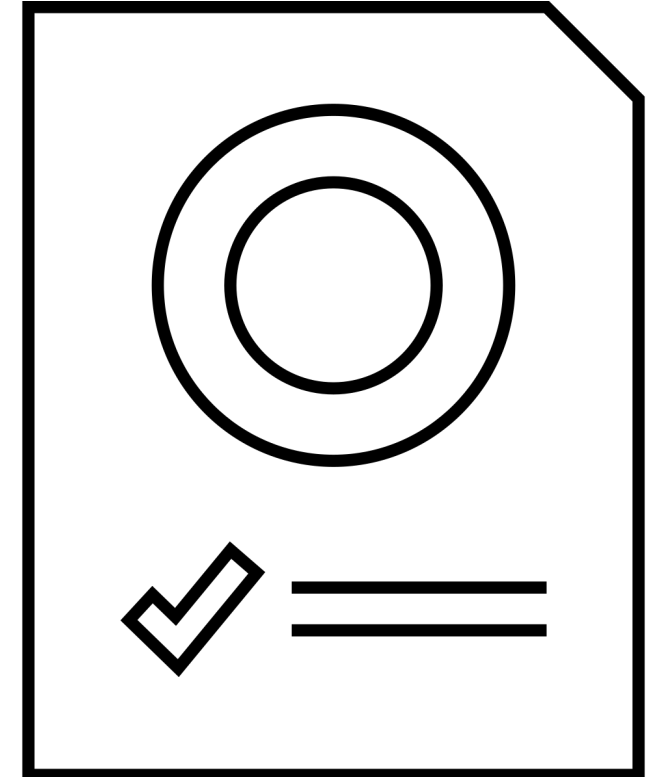
Specifics – Controllers, etc

- As with views, if you are only adding subrecords to the staff side, chances are good that your controllers and other classes, methods, etc won't need any updating.
- If you are overriding or extending any core code, make sure to check the core code for changes that you may need to incorporate or be aware of.



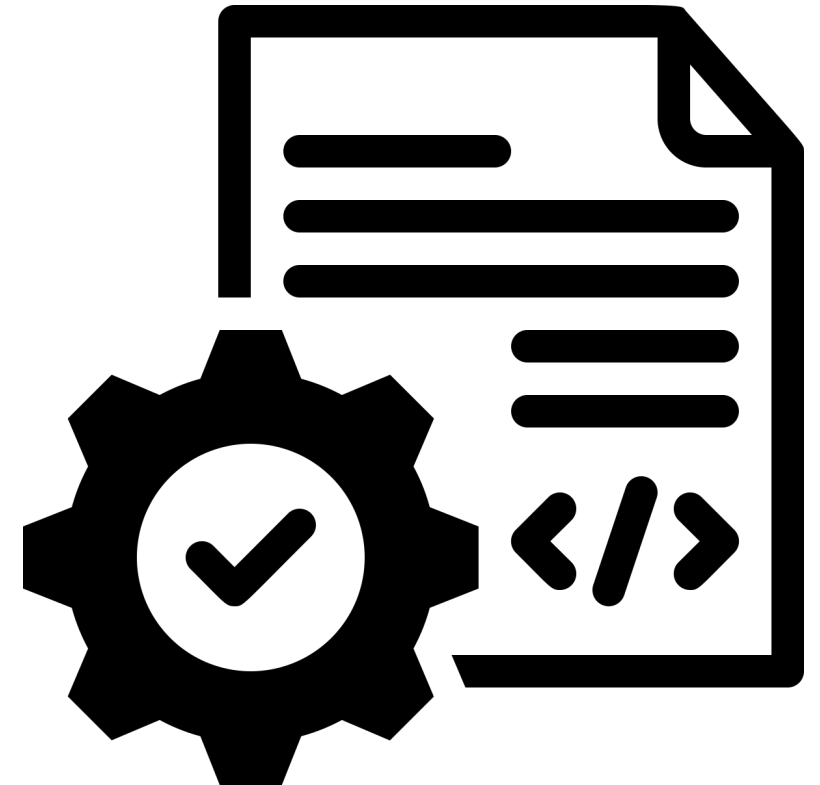
Specifics – PUI

- The PUI has a few extra things to check since you can also override the PDF exports. Make sure to check those core views if you have made any changes in your plugin.



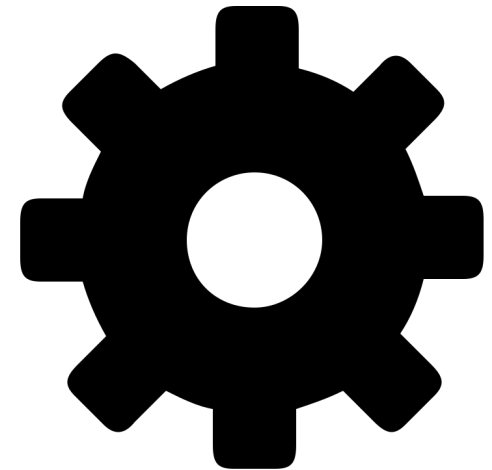
Specifics – Other pieces

- Check your models, migrations, and schemas.
- Its unlikely that these will need to be changed UNLESS you are overriding or extending some core functionality.
- You're most likely to need to check any code that deals with some of the exports (EAD, MARC, etc)



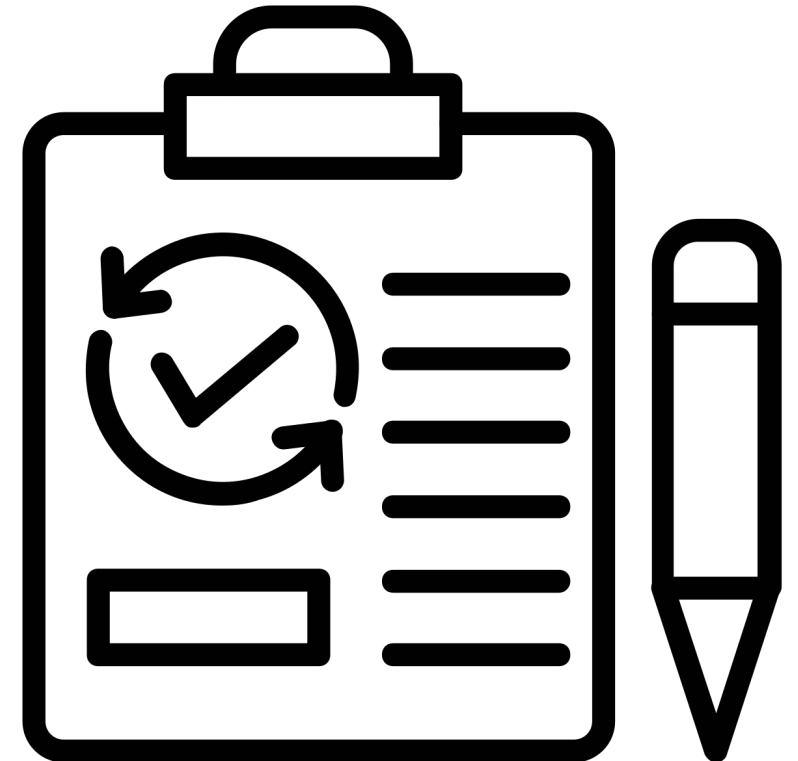
Specifics – Underlying Technology Changes

- Check to see if any core technology changes impact your plugin.
- Example: The recent(ish) update from v2.8.0 to v2.8.1 included a major Rails update. That update included a change to the way that params were treated.



Test it!

- Before releasing any updates, test your plugin in a **NON-PRODUCTION** environment.
- Best if you can personally test locally during development and again with additional users in a **NON-PRODUCTION** environment.





Thank you!

joshua.d.shaw@dartmouth.edu

All icons from the Noun Project.

