DOIng more with Hyrax

•••

Chris Colvard
Ubiquity Press

Samvera Connect 2020
October 28th, 2020

repositories

What is a DOI?

doi "Digital Object Identifier"   Example DOI: 10.18130/v3-xt15-3w34

- Resolvable via https://doi.org
- Unique prefix 10.xxxx assigned and managed by registration agency
- Many registration agencies for different knowledge domains
- Register a DOI with the object's url and metadata
- Metadata schemas vary among registration agencies
- Some DOIs have a status (draft vs. published)
- Noteable registration agencies:

Crossref

DataCite
FIND, ACCESS, AND REUSE DATA

Previous work on DOIs

Samvera EZID to DataCite DOI WG

Hydra-remote_identifier

UCrate (Hyrax-based app that uses a branch of the above)

Mahonia (Hyrax-based app)

Libra2 (Sufia-based app)

And many more

Introducing Hyrax-doi

History

- Feature spec and workflows originally developed by Ubiquity Press for British Library's Hyku application (Shared Research Repository)
- Extracted as part of Advancing Hyku alignment work

Use cases

- Create DOI for a work
- Mint a Draft DOI before work has been deposited
- Update DOI metadata when work is updated
- Autofill the deposit form with metadata from a DOI

New Hyrax Infrastructure

Hyrax infrastructure for external identifiers ported from Mahonia (available in Hyrax 2.9+)

- Dispatcher - assigns registered identifier to a given object
- Registrar - handles communication with external identifier service
- Builder - constructs identifier to submit to external identifier service
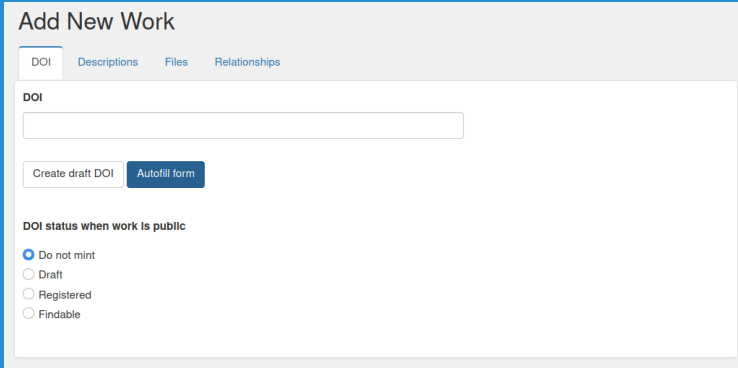
Configure in Hyrax initializer:

```
config.identifier_registrars = { datacite: Hyrax::DOI::DataCiteRegistrar }
```

Then mint and assign an identifier to a work:

```
Hyrax::Identifier::Dispatcher.for(:datacite).assign_for!(object: work)
```

Hyrax-doi Features

- Enabled per work-type
- DOI tab on deposit form
- DOIs created or updated on work save (using a background job)
- Autofill form using [bolognese gem](#) which can read from many registrars
- Work-type metadata crosswalking to DOI metadata also through bolognese

DataCite specific features (when enabled):

- DOI statuses including draft
- Form validation (warning if not all required DataCite properties present)

Feedback and Future

Would like adopters and feedback on:

- Default metadata crosswalking
- UI/UX on deposit form
- More registrar implementations
- Code abstractions and choices

Next steps:

- Hyku integration (included in [Hyku roadmap](#))
- Admin settings for enabling/disabling and registrar configuration

GitHub

[Hyrax-doi on GitHub](#)

[Shared Research Repository](#)

Advancing Hyku Project

[Advancing Hyku Project](#)