

# 2015-10-09 - Fedora API Extensions Meeting

## Dial In Details

Date: Friday September October 9, 2pm EDT (-4 UTC)

- ★ **NEW** ★ Dial-in Number: (712) 775-7035
  - Participant Code: 479307#
  - International numbers: [Conference Call Information](#)
  - You may also call in using the [VoIP dialer](#) from a web browser, or Android/iOS apps
- IRC:
  - [Join the #fcrepo chat room via Freenode Web IRC](#) (enter a unique nick)
  - Or point your IRC client to #fcrepo on [irc.freenode.net](#)

## Meeting Goals

Ratify the [list of things we want to distill from use cases](#)

Close the loop on [Stefano Cossu's content and structural validation use case](#)

Discuss the potential role of API-X for supporting transactions

Review the use case for composite (referred to as "chaining" in the previous meeting) services

## Attendees

- [Aaron Birkland](#)
- [Unknown User \(acoburn\)](#)
- [Stefano Cossu](#)
- [Ruth Duerr](#)
- [Bethany Seeger](#)
- [Joshua Westgard](#)

## Agenda

1. Ratify [initial list](#) of "what to distill from use cases". ([Aaron Birkland](#))
2. [Refined](#) validation use cases from [Stefano Cossu](#)
  - a. [Enforce validation across repository](#)
  - b. [Optional validation](#)
  - c. [Validation only for selected resources](#)
3. Role of API-X for supporting transactions (??)
  - a. [Actor models](#) - having the API-X (or an extension) be an actor
  - b. Transactions as a core feature of the Fedora 4 platform
4. Think about composition of extensions for next meeting?
5. Any Other Business
  - a. Elliot away: 9/28 - 10/28

## HydraConnect touch points

Taking the API-X effort coupled with F4's direction towards a continued shrinking of the core codebase and limited scope to the core services, many conversations around "repository capabilities" naturally migrate towards API-X solutions. That is very general, but if we establish a robust yet simple API-X framework, I anticipate it getting a lot of use. The specific, Hydra-related use cases that came up at HydraConnect are:

- batch ingest
- locking resources

## Related Resources

[Design Page \(with use cases outline\)](#)

[Use Cases Parent Page](#)

[Previous meeting agenda, including minutes](#)

## Minutes

- Consensus on use case evaluation - let's use the proposal as-is
  - Should we create sub-pages of each use case, or edit/modify pages in place?

- Editing in place is probably more clear
  - Decision: Edit in place
- Action item: Everybody evaluate/reformat at least one use case
  - [Unknown User \(acoburn\)](#) believes that most or all of Amherst use cases can be finished by next call
  - [Joshua Westgard](#) will adopt deposit/ingest related use case
  - [Aaron Birkland](#) will work with [Ruth Duerr](#) for ESIP use case.
  - [Stefano Cossu](#) may add additional AIC use case related to notifying clients when depositing duplicate binary resources
- Validation use cases - discussion around whether validation extension would be 'globally on' (with the means to configure it to disable validation where it's not wanted), or a hook to have API-X invoke validation extension or not
  - Decided to just go with 'globally on'
- Actor model seems well suited to API-X
  - Extension is autonomous 'actor' that does everything for you
  - Client interacts with it via message passing (e.g. POST a list of resources to be ingested)
  - Actor does all the heavy lifting (possibly employing transactions with Fedora) and responds
  - Client wouldn't be aware of transactions
  - For concurrency, actor could possibly be a single thread + queue, all operations serialized
  - API-X framework would largely be agnostic about what's going on in an extension, so we're taking about what occurs within an extension when it's handling a user request.
- Aaron to find out more about Hydra use cases
- Meeting on the 23rd may be problematic, will send out poll to see if another date better