

Time/Place

- Time: 11:00am Eastern Daylight Time US (UTC-4)
- Dial-in Number: (712) 775-7035
 - Participant Code: 479307#
 - International numbers: [Conference Call Information](#)
 - Web Access: <https://www.freeconferencecallhd.com/wp-content/themes/responsive/flashphone/flash-phone.php>
- 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

- A. Soroka
- Esme Cowles
- Jared Whiklo
- Bethany Seeger ★
- Benjamin Armitor
- Andrew Woods
- Allen Flynn
- Namita Bahulekar
- Unknown User (acoburn)
- Stefano Cossu

1. PUT with	
a.	Unable to locate Jira server for this macro. It may be due to Application Link configuration.
2. 4.5.1 Rel	
a. One blocker:	Unable to locate Jira server for this macro. It may be due to Application Link configuration.
b. Release Test	
3. Bugs are beginning to pile up	

4. Fedora Specification updates
 - a. [Messaging SPI](#)
 - b. [Atomic Batch Operations](#) - name? BatchOps? Bag-o'-Ops? OpSack? AtomicOp?
 - c. [CRUD](#)
 - d. [Resource Versioning](#)
 - e. [Binary Fixity Checking](#)
 - f. [Authorization](#)
5. Updates on release status of fcrepo-camel and toolbox?
6. ...
7. Status of "in-flight" tickets

key	summary	type	created	updated	due	assignee	reporter	priority	status	resolution
Unable to locate Jira server for this macro. It may be due to Application Link configuration.										

Ticket Summaries

1. Please squash a bug!

key	summary	type	created	updated	due	assignee	reporter	priority	status	resolution
	Unable to locate Jira server for this macro. It may be due to Application Link configuration.									

2. Tickets resolved this week:

key	summary	type	created	updated	due	assignee	reporter	priority	status	resolution
	Unable to locate Jira server for this macro. It may be due to Application Link configuration.									

3. Tickets created this week:

key	summary	type	created	updated	due	assignee	reporter	priority	status	resolution
	Unable to locate Jira server for this macro. It may be due to Application Link configuration.									

Minutes

1. PUT with server-managed triples... fail or ignore?
 - a. when you have a resource that you want to update with a PUT request (body being RDF, that doesn't not include server managed triples) the current logic tries to remove the triples that are not in the body of the RDF - throwing a 4XX error. Question - should we not require the server managed triples be included in the body of RDF request? Use case is to update the resource w/o having to know the server managed triples, since we can't change them anyways.
 - b. [Esmé Cowles](#) reminded us that there is a `Prefer` header `""handling=lenient;return="minimal""` that should do the trick.
 - c. Why is the default behavior taking server managed triples into consideration in the first place? Should it be the other way around? Lenient default - and have a strict option? Use case: one may want the full graph in a case where other's are editing at the same time - for conflicts.
 - d. [Esmé Cowles](#) has used this recently - would be surprised if not implemented, since it's worked for him.
 - e. [Esmé Cowles](#) will test this (the lenient and return=minimal)
 - i. Confirms that PUT with return=minimal does do the requested behavior - updates the user triples.
 - ii. [Esmé Cowles](#) to update the [documentation](#) for this
 - iii. Should work for Stefano's use case.
 - f. [Andrew Woods](#) - Is there any situation that should require putting the server managed triples back on the server? Would this capability help in some way? ETags should serve the same purpose.
 - g. Concern about what it means if someone tries to write server manage triples that are wrong? Persist anyways?
 - i. if you're trying to change the value on a server managed triple, you should get an exception. (including implicit removal for triple not specified in request).
 - ii. LDP spec says HTTP PATCH is for merge of data, and to use HTTP PUT for replacement.
 - iii. HTTP PUT for replacement - but with server managed triples it's more complicated.
 - iv. The delete and recreate use case is the one to figure out how to work with here.
 - v. Want to be careful of situations with server managed triples that change w/o user knowing (Indirect Containers, for example).
 - h. Two ways to do this stuff
 - i. PUT with return=minimal
 - ii. SPARQL update via PATCH
 - i. is it worth someone's time to investigate a different behavior here, or are things good as is?
 - j. handling=strict; return="minimal" seems t
 - k.

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

 - to add comments to

Unable to locate Jira server for this macro. It may be due to Application Link configuration.

2. 4.5.1 Release planning

- a. one blocker - Unable to locate Jira server for this macro. It may be due to Application Link configuration. - auth is a core feature and if
 - i. what do we think the expected behavior should be in this situation (a batch operation)?
 - 1. a link to an ACL that is prefixed with transaction context ?
 - 2. a link to an ACL that is not prefixed with transaction context?
 - 3. a link to an ACL that points to non-accessible uri?
 - 4. no ACL link header?
 - 5. behavior different depending on whether or not the ACL is created w/i the batch request?
- b. Any other tickets in before we put out release? None put forward
- c. Who can work on release?
 - i. Benjamin Armintor - Release Manager
 - ii. Esmé Cowles and Jared White - happy to help out

3. Bugs are

- a. Unable to locate Jira server for this macro. It may be due to Application Link configuration. - Unknown User (acoburn)
ent is emitted for that resource. However, with MOVE, you can make changes on many resources, so we need to be able to fire off an event for each resource that changes as a result of that action. Easiest way to do this may be to say that there's a delete event and a create event for each contained resource.
 - i. maybe emit a message ...
 - ii. delete events must recursively traverse tree.
 - iii. do we create the messages manually, or take advantage of what modeshape gives us? Later accomplished by not using modeshape move - but using modeshape creation and delete methods, resulting in messages. However could end up in inconsistent state.
 - iv. perform MOVE and then async send out delete and create messages
 - b. Please review bugs and help on some of the major ones.
4. Please continue working on the Fedora Specifications. If you're looking for feedback, please move the JIRA ticket related to your doc to "in review".