

Modelling Overlay Peer Review Processes with Linked Data Notifications

By Paul Walk, Martin Klein, Herbert Van de Sompel, and Kathleen Shearer

[Use cases contributors](#)

August 4, 2020

This draft is being shared for public comment until September 11, 2020

Introduction

In November 2017, the Confederation of Open Access Repositories (COAR) published a report outlining the technologies and behaviours of the Next Generation Repository (NGR). In the report, the NGR Working Group argues that repositories must take their place in a resource-centric network, where the individual resources (metadata and actual content) within the repositories are interconnected on the Web both with each other and, more importantly with resource-oriented networked services. These links between resources and overlay services can bring many new opportunities for broadening the scope of the services offered by repositories and 3rd party initiatives. The emphasis on moving to a fully resource-centric paradigm presented in the vision for the Next Generation Repository offers an opportunity to exploit what programmers call “pass by reference” - a notion which underlies the fundamental function of the Web.

One specific use case related to this vision is the linking of repository resources with services providing commentary, annotation and peer reviews; a use case that is currently being considered by several different initiatives in the scholarly communications landscape. The wide distribution of resources (typified by articles) in repositories, coupled with the growing interest in overlay journals, introduces the possibility of adopting an asynchronous notification paradigm to achieve interoperability between repositories and peer review systems.

In January 2020, COAR convened a meeting to investigate the potential for a common approach that would connect repositories with overlay peer-review services, and, therefore, primary scholarly resources with reviews. A number of different use cases were presented, each with their own unique attributes. However, it became clear that there are significant similarities in terms of workflows, functionalities, and objectives that a common approach can be developed.

The use cases involve interactions between an author of an article and peer reviewers (e.g., overlay journal). The typical peer review workflow a number of exchanges such as: submission of article, rejection or acceptance to undertake the review, request for revision of article, and acceptance or rejection of article for publication. While there are other potential nuances or alternatives to this workflow, these represent the majority of the communications described in each of the use cases. We envision that this approach could be adopted for a number of different communities and domains, with varying peer review practices, as well as can be expanded to a variety of content types beyond articles, such as research articles.

Rationale

The resource-oriented nature of the Web is well suited to an environment which places value in the fact that control of resources is distributed across a large number of repositories. In such an environment, it makes sense to take a pass-by-reference approach to interaction between different networked services, rather than relying on machine or human mediated processes to pass copies of resources around the network.

Resources in repositories have stable URIs that can be used for referencing. This means that a request for review can be sent as a standards-based notification that carries a resource's stable URI to the inbox of a review service. This also means that the review service can obtain the resource that is to be reviewed by visiting that stable URI. From there, the actual resource can be retrieved by following some simple standards-based navigational conventions (e.g., retrieve the full text of a preprint, automatically, from having accessed a landing page describing it). Generally, this means that it becomes possible to invoke and use remote services on the network, by passing instructions to them together with, crucially, URIs identifying particular resources.

Where there is a set of reasonably well understood functions provided by a potentially large range of services, it makes sense to try to encourage the use of standard ways of invoking those functions. Furthermore, where the number and range of services and consumers (repositories) is large, bespoke point-to-point service interactions are unfeasible.

Fortunately, W3C has already developed some very useful standards which may be used to pass notifications between systems such as repositories and other, related service, where those services provide useful functionalities (e.g., annotation, review...) related to the resources held in repositories:

- Linked Data Notifications¹, providing a standard for a system to send a notification (e.g., a request for some service), often bearing a URI (e.g., identifying a resource in a repository), to a remote system, in an asynchronous fashion
- Activity Stream², providing a machine-readable, semantically standardised way to describe “activities” especially related to interactions with resources on the Web

A prototype has already been developed and was presented by Herbert Van de Sompel and Martin Klein at the meeting convened by COAR in January 2020. Based on a review of the use cases also presented at that meeting, we confirmed that this prototype can generally support a variety of cases, but the key will be to agree on a standard set of vocabularies for the notifications.

This document presents some simple models and vocabularies for using standard notification protocols to achieve common interactions between repositories and overlay peer review services, based on the use cases provided.

Workflow components modelled with Linked Data Notifications (LDN)

The following models each describe a "notification" between a repository and a journal system. One component which is common to all is the URI called *URI-P*. This URI identifies a resource in the repository (typically a preprint). For the purpose of these models, it is assumed that both metadata and the actual manuscript may be retrieved by a remote system (e.g., the journal) by visiting URI-P and from there, for example, follow HTTP Link Headers that convey the necessary information, as outlined in the Signposting concept³. In the model presented below, the request for review is initiated by the author, through the repository. However, we also envision an alternative use case in which the request for review is initiated by a peer review service that has identified a preprint or article which would be of interest.

¹ <https://www.w3.org/TR/ldn/>

² <https://www.w3.org/TR/activitystreams-core/>

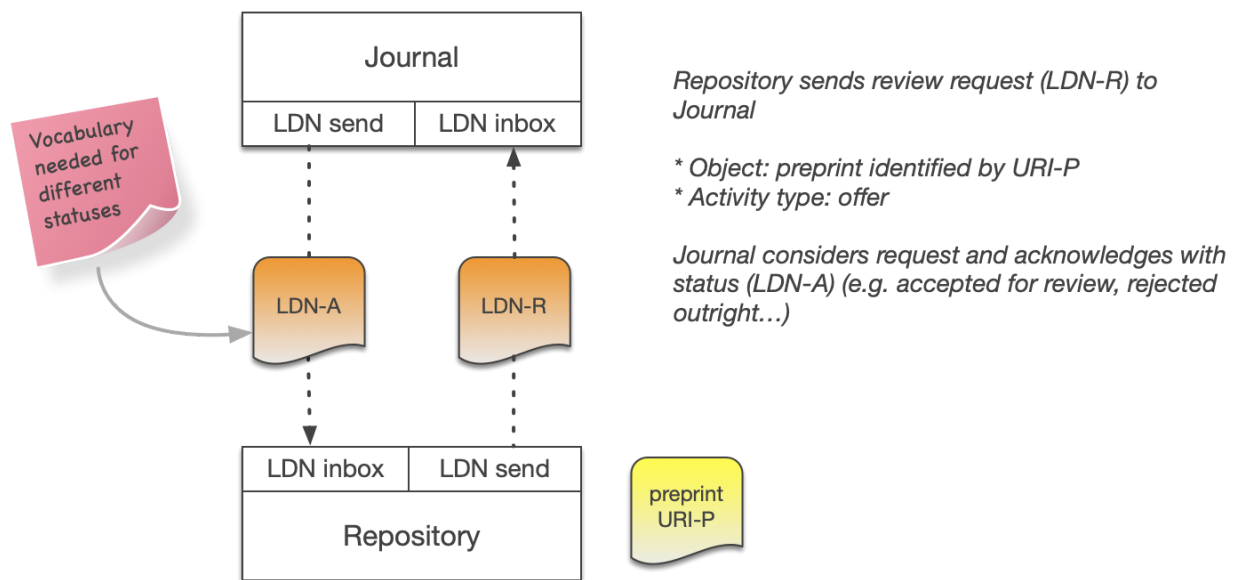
³ <http://signposting.org/>

Notification 1: Request for review initiated via repository

Pre-state

A preprint manuscript and associated metadata have been deposited in a repository by an author. The author has requested that the preprint be reviewed by a journal for possible publication there. Overlay peer review and other services available to authors and the repositories would be listed in a registry/directory that could be accessed in an automated way via the repository interface.

Model



Notes

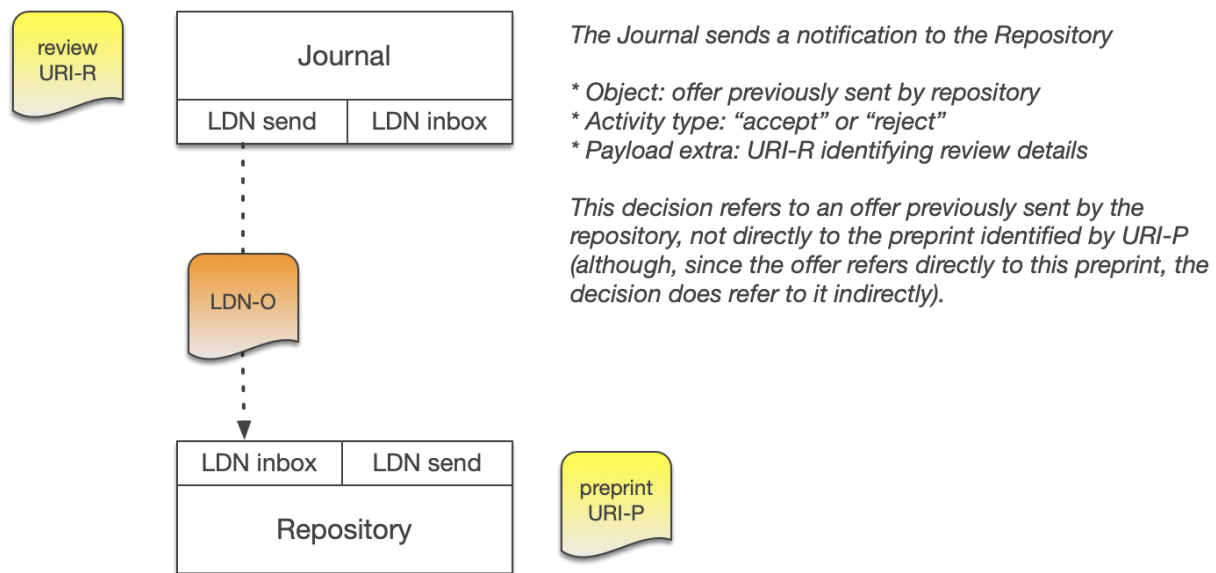
The acknowledgement notification (LDN-A), while not entirely necessary, can be a highly useful response to let the repository and author know that their request for review is being considered.

Notification 2: Outcome of review

Pre-state

The journal has reviewed a submission from the repository, and wishes to inform the submitting repository of the result.

Model



Notes

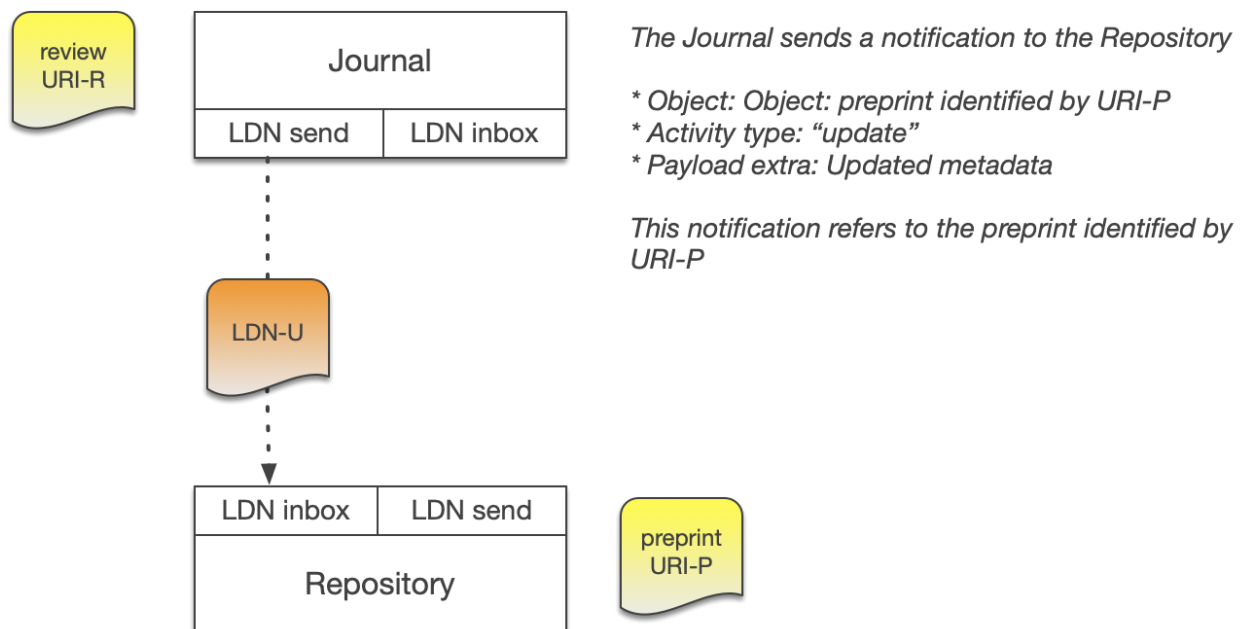
The reviews (metadata, comments, decision) are held by the journal (since this is where the review was conducted), but it is identified by URI-R which may be shared with the repository in the payload of the notification (LDN-O). Just the presence of URI-R may be enough to indicate that the journal has carried out a review of this manuscript and the URI-R can be recorded in the metadata associated with the manuscript in the repository so that the same manuscript is not re-submitted for review to the same journal. How much metadata might be made available by de-referencing the URI-R would depend on the journal's privacy policy for reviews. However the strength of this general approach is that the preprint and the review of that preprint are linked to each other - that is, it is possible to find the review from the preprint, and *vice versa*.

Notification 3: General update

Pre-state

The journal is already aware of a preprint identified by URI-P and held in the repository. The journal wishes to notify the repository that it has new information about the preprint, for example that the preprint has now been reviewed, will be published, and now has a PID for the version of record which may be added to the metadata record associated with URI-P, held in the repository.

Model



Opportunities for vocabulary development

For these interactions to scale beyond a single system to system interaction and be adopted across a distributed network of repositories and peer review services, there is a need to define and adopt a controlled vocabulary that will standardise the requests and responses between the repository and overlay service.

In the table below we provide some examples of notifications that will need standard vocabularies. These vocabularies will be defined in the next phase of development and implementation of this model.

Table: Examples of workflows and related notifications from use cases

Workflow(s)	Notification
Author deposits article and chooses an overlay review service from a pre-existing list	Request for review
The overlay service acknowledges that the request for review has been received	Acknowledge request
Overlay service decides to accept article for review (or not)	Response to request for peer review
Article reviewed by reviewers	(Article stays in repository)
Peer reviews are complete author is notified of result	Response to decision about whether to publish article
When revisions requested, then new version of article uploaded to the repository	Repository informs overlay service that a new version is available for review