Review of http://tools.oasis-open.org/version-control/browse/wsvn/oslc-core/specs/attachments-v3.html

Review by ian green (oslc core tc). SVN rev 82.

I point out in the "Details" section below several places where I think the spec is being overly constraining or prescriptive. Perhaps this is to meet some requirement I don't know about? If not, I suggest that the normative statements be really the essence of what is needed - to my mind that would be along the lines of

- Terms to describe AttachmentDescriptor resources
- Terms to link a resource to attachment containers
- Attachment container == LDPC
- Use of Slug to seed name of the attachment, and the requirement that server should not fail a request because of Slug
- What do do when Content-Type is missing
- That when attachment is updated, so must its descriptor be updated (if there is one)
- When attachment is created, and there is a descriptor it MUST be linked

The other statements are either HTTP or LDP behaviours or non-normative. I think this would make the spec easier to read and understand from a compliance perspective.

Details

1. Question on scope: Is it the case that any LDPR can be an attachment? 7.3.1 currently states that an attachment MUST be an LDP-NR, and my understanding of LDP is that this means that an attachment would not be allowed if it were, say, application/rdf+xml.
2. Suggestion: in section 5 which is declared non-normative, there are what look like normative statements. I suggest rewording to remove any normative statements and try to bring out the role of an attachment descriptor and its connection with an attachment - the idea is that an attachment descriptor describes the size, media type and title of the attachment, and that there is a way of taking the POST and building such a descriptor from that.
3. oslc:AttachmentContainer doesn't seem to be defined as a vocabulary term (its in the turtle, but not in section 9). I wasn't sure why.
4. I think the examples should come after the normative content which defines the terms appearing in those examples.
5. 7.1.1 "at least" seems redundant and/or I don't think defined anywhere.
6. 7.1.1. what is an "Attachment server"?

7. 7.2.1 “holds attachments for only that resource” - why is this necessary? In the case there is more than one attachment container on the resource, this spec is silent on how a client would work out which to use (that’s ok, but what is the use case for more than one?)

8. 7.2.2 Why is http://open-services.net/ns/core#AttachmentContainer used here when that term is also an rdf:type? Shouldn’t this be a distinct term, say http://open-services.net/ns/core#attachmentContai

9. 7.2.3 Why is this in the spec? We might want to additionally constrain the representations of those LDP-RS resources which have attachment containers so as to ensure that the HTTP header and and triples in the representation are in agreement (same containers), but I don’t see the need to have this clause otherwise.

10. 7.2.3 How would this be done? I was expecting a property (not a class) http://open-services.net/ns/core#attachmentContainer. This term would need to be included in the vocabulary.

11. 7.2.2 suggest rewording so make clear there MUST be a Link for each and every AttachmentContainer associated with the resource (as permitted in 7.2.1) . Not sure if need to say anything about ordering of those Link headers.

12. 7.3.1 See question above on scope. I think this ought to be generalized to LDPR.

13. 7.3.2 Is this necessary, and is it even correct? What about a server that needs to do a redirect, for example?

14. 7.3.3 suggest rewording to cover PUT case: "from an HTTP POST request that created the attachment, or the most recent HTTP PUT which changed the attachment". I wonder if this is “obvious” to the point it doesn’t need to be stated?

15. 7.3.4 Is there a reason this is not a SHOULD?

16. 7.3.6 Standard HTTP. I don’t see this need to be in this document.

17. 7.3.9 Standard HTTP. I don’t see the need to be in this document.

18. 7.4.2 Standard LDP-C. I don’t see the need to be in normative section of this document. I would think this would be in the “basic outline” section.

19. 7.4.4 If a client violates “the Slug header SHOULD NOT include a file extension” what happens? What happens when there is more than one Slug?

20. 7.5.4 This is ambiguous since a client-supplied Slug, according to section 7.4, may be absent, ignored or altered by the server. I think a simple rewording which is that the dcterms:title should be whatever the server chooses, informed by the client-supplied Slug, if any.

21. 9. Is this spec trying to restrict the way in which wdrs:describedBy can be used when the object is an AttachmentDescriptor? The cardinality restriction doesn’t apply to the object of links. Instead, the spec needs to state the relationship between a attachment and its descriptor (each descriptor describes exactly one attachment) and an attachment has at most one descriptor. The relation between these resources is represented using describedBy. I don’t see why describedBy is described as an “inverse property” (inverse of what?) it is just a link between two resources.

22. 9. Are there any restrictions (eg as in 7.4.4) that restrict the dcterms:title of an attachment-descriptor? Can a server accept a change to dcterms:title and alter it as it does when a client supplies a Slug?

23. 9. I think http://open-services.net/ns/core#AttachmentContainer should be defined in this section (and as I have suggest above), http://open-services.net/ns/core#attachmentContainer.

End of review