

IIIF A/V - Toronto

12 October 2017 - 9am-12.30 EST

Part of *Presentation 3 API day*

On October 12, we have the AV session in the morning (this document) and Presentation 3 session in the afternoon. This gives us a full day of Presentation 3, as any spec-related issue for AV that requires ratification or discussion is also a Presentation 3 issue by definition. In order to maximise the time we have together, we suggest that those Presentation issues that are most relevant to AV are ratified or discussed in the morning session.

Slides

<https://docs.google.com/presentation/d/1XN5-GRaiMBG6NkTroGfHvHJshifv8fKSvXSnDOYxYXc>

Agenda

Session leader: Tom Crane

Assign Timekeeper and Notetaker:

- Timekeeper: Simeon Warner
- Notetakers: Karen Estlund, Jon Dunn

Notes at bottom of doc

9:00 Background (10 mins)

- Relationship between AV and Presentation API work
- Timing Proposal

9:10 Ratification of Presentation 3 issues related to AV

The following issues have been discussed extensively in the community and are marked as ready for ratification in Github. We aim to present each issue in very brief summary form, and ratify. If you have objections to any of these, please communicate them in advance, before the meeting

[Issues on Github](#)

- <https://github.com/IIIF/iiif.io/issues/1069> Add duration to canvas

- <https://github.com/IIIF/iiif.io/issues/1075> Add timeMode
- <https://github.com/IIIF/iiif.io/issues/1176> Non Image thumbnails
- <https://github.com/IIIF/iiif.io/issues/1191> Canvas to Canvas annotations
- <https://github.com/IIIF/iiif.io/issues/1190> Spatial canvas on duration canvas?
- <https://github.com/IIIF/api/issues/1070> Ranges
- <https://github.com/IIIF/api/issues/1255> First range hierarchy should be displayed by default
- <https://github.com/IIIF/iiif.io/issues/547> Authentication on non Image Service

9:40 - 5 minute break

9:45 Discussion of Presentation 3 issues related to AV

(10-15 minutes each)

Issues on Github

- <https://github.com/IIIF/iiif.io/issues/1194> Add choiceHint
- <https://github.com/IIIF/api/issues/1263> posterCanvas
- <https://github.com/IIIF/api/issues/1259> Thumbnail-navigation viewing hint for ranges
- <https://github.com/IIIF/api/issues/1264> RenderingHints as applied to duration canvases

Annotation related:

- <https://github.com/IIIF/api/issues/1257>, <https://github.com/IIIF/api/issues/1256> - text rendering issues (in AV because captioning, subtitling)
- <https://github.com/IIIF/api/issues/1258> (and #511) motivations for associating text and canvas; captioning
- <https://github.com/IIIF/api/issues/754> Summarizing various features of annotations in the annotation page or annotation collection.

11:00 - 30 minute break

11:30 - 12:30

Continuation of discussion of issues above.

12:30 Adjourn for Lunch

Notes:

Background

- Originally had thought that there may be an a/v bitstream api like image, but a/v works differently on the web! Parameters to make derivatives on the fly are too expensive for things like video, too.
- A/V happens in presentation land (e.g. Shared Canvas) rather than bitstream APIs

Ratification of Presentation 3 issues related to AV

- <https://github.com/IIIF/iiif.io/issues/1069> Add duration to canvas
 - **So ratified!**
- <https://github.com/IIIF/iiif.io/issues/1075> Add timeMode
 - Stretching stuff of time dimension is weird! Hard for clients and would produce off effects
 - timeMode (trim, scale or loop) determines the behavior of the client when the time doesn't match for annotation and canvas; "trim" would be default
 - Scale and loop must be explicit
 - Extensible if you want your own behaviors and make your client handle it
 - **So ratified!**
- <https://github.com/IIIF/iiif.io/issues/1176> Non Image thumbnails
 - Loosen thumbnail so that anything that is useful that represents the client (e.g. 30 sec audio samples, short video clips, etc.)
 - If you have both images and audio clips, you could list them all, and then the *client* by inspecting the type can figure out what it wants to use
 - You can't just say thumbnail.jpg; you need to say what type of thumbnail it is
 - Questions/Discussion:
 - Can you have different dimension images? Yes, you can put in multiples and allow the client to use the appropriate one to present; also recommend that you put in dimensions for images to render nicely
 - Was there a discussion about mimetype? You could add format property
 - What types? DCMI-types or you could add your own by URI
 - Propose allowing text (as-in alt-text)? The label of the thumbnail could serve that purpose; is that sufficient? From an accessibility standpoint, alt and label are not the same. Separate issue - RS is on it!
<https://github.com/IIIF/api/issues/1288>
 - If you have a resource text? What does that mean if not description or abstract? Separable issue for future discussion
 - **So ratified!**
- <https://github.com/IIIF/iiif.io/issues/1191> Canvas to Canvas annotations
 - Ability to take already-annotated canvas and annotate it within another canvas

- Important for AV because audio/video items (potentially with other annotations) will almost always be represented by a canvas, but useful for other use cases beyond
- **So ratified!**
- <https://github.com/IIIF/iiif.io/issues/1190> Spatial canvas on duration canvas?
 - ALLOW canvas to allow expansion when needed when body and target dimensions don't match or NOT
 - Only applies to "motivation": "painting"
 - Can't annotate time on a canvas with no time
 - Can't annotate an image when canvas only has time dimension
 - Questions/ discussion:
 - What about time dimension that extends a time of canvas? - different issue; much like adding an annotation to where there is no space for image canvas; differences in duration are controlled timeMode. The combined set of rules of canvas on canvas, and timeMode and dimension rules gives you an algorithm to calculate all transformation to get items on the base canvas
 - Is accessibility an issue, e.g. should we support painting an audio description onto an image-only canvas? Could instead create a new canvas. Different motivations and relationships address how you may add accessibility
 - **So ratified!**
- <https://github.com/IIIF/api/issues/1070> Ranges
 - Longstanding fun discussion. See issue: <https://github.com/IIIF/api/issues/1070>
 - Nicer for ranges to be more tree-like in the metadata
 - Several implementations were modelled for implementation and favorite chosen: nested over references
 - See example at https://github.com/zimeon/iiif-ranges-examples/blob/master/book/book_simple_ext3.json
 - Questions/Discussion:
 - Does same canvas show in range and sequence? Yes; that before has not changed
 - What happens when ranges overlap? That is unaffected. For example you could render a table of contents to go to the ranges that you want, regardless of overlap
 - What should the client behavior be when someone clicks on a range with start and end times? Should jump to start time of range.
 - In previous situation with flat list, working with a book of 1,000 pages with no hierarchy, debugging is also a lot easier
 - You can also have multiple hierarchical range structures
 - Gets rid of need for "top" viewingHint
 - **So ratified!**

- <https://github.com/IIIF/api/issues/1255> First range hierarchy should be displayed by default
 - Always start at the top; Gets rid of need for “top” viewingHint
 - **So ratified!**
- <https://github.com/IIIF/iiif.io/issues/547> Authentication on non Image Service
 - May need more discussion/proof-of-concept implementation - not yet ready for ratification
 - Same pattern as current Auth spec; extension to current spec, not breaking
 - For images, HEAD request on service description json used as a probe; HTTP status code returns availability
 - With AV, no service to probe
 - Initial proposal was to declare a service for each resource to be used in auth workflow - see issue
 - Current proposal is to declare explicit probe service with URI:
 - Probe service just needs to return status code in response to HEAD request
 - Could be URI for video/audio file itself or could be something else
 - Is it subservice of <http://iiif.io/api/auth/1/token?>
 - Should be at same level as token
 - If no explicit probe service, use service description json
 - Suggestion to close 547 and create two new issues:
 - Invent profile for describing a services document not tied to a bitstream API: <https://github.com/IIIF/api/issues/1289>
 - Allow description of a probe resource separate from the services document: <https://github.com/IIIF/api/issues/1290>
 - Is this essential for Presentation 3.0?
 - No, pattern could be described in auth spec

Discussion of Presentation 3 issues related to AV

- <https://github.com/IIIF/iiif.io/issues/1194> Add choiceHint
 - E.g. you have 3 different bandwidth versions; you don't want them to be played together, because they are all the same thing; who do we want to make the choice and how does the client know?
 - This solution provides a choiceHint that specifies who should make the decision (client, user)
 - Do we have enough information to do this? Do we have use-cases when we want the client to make the decisions
 - Not talking about mpeg-dash that allows the client to make choice by definitional structure of mpeg-dash
 - Questions/Discussion:

- Is this a discussion about when the user can make the decision? Avalon for example gives a default, then user can override. Avalon always picks medium, but how does it know it's medium?
- Without semantics of quality, do we have enough information to be a useful feature for client? Client can't really make it unless quality is specified. Client may also not be able to play the type
- Even on the user side; how to display to the user what to choose? Labels would have to be provided to express to users
- Is this also just outside of IIIF scope?
- Should it just be left up to client, e.g. client plays first choice listed by default or up to client to choose?
- In the case when you have a multispectral image or video, how to pick band of spectrum? No way to tell the difference, so have to leave it to the user
- After client makes choices it has to make (e.g. what can it play/provide), then is there anything useful that the client can do
- Defer or close?
- Currently for images in Mirador, first; behavior is idiosyncratic
- **Deferred!**
- <https://github.com/IIIF/api/issues/1263> posterCanvas
 - Difference between poster image and thumbnail
 - E.g. with netflix, when you're clicking on what to watch you're looking at thumbnails, when you click on it, then what you then see is a poster (it can be an image, image/text, etc.)
 - Proposal to separate out; if you make a canvas rather than an image, then you can do whatever you like with it (e.g. different dimensions)
 - Questions/Discussion:
 - Could it be an annotation?
 - It could be an annotation of that canvas on the canvas
 - The spec would be cleaner, with one less property
 - In order for annotation to be correct, then you'd need a motivation for poster-style behavior
 - Avoid artificially inventing width/height for audio to add the poster
 - What is the problem with representing an x/y thing with an image as an innovation? See related issue about limiting dimensions when dimensions don't match for "motivation": "painting"
 - Could you create a canvas and fill the use case by presenting the duration based media with x/y coordinates? Yes. Looking at the minimum rendering quality; why make a client render an image for an audio file?; could interfere with audio annotation tools if you have to display the poster (the poster is about the object being modeled, not part of the model)

- Note: not just audio; this is also a mixed media issue
 - You could have a variety of thumbnails and a poster for it
 - Is it valid for other objects? Yes, e.g. you could have a poster for a manuscript?
 - NCSU shows a poster of image while video is loading
 - Allow more flexibility, so you can have more than image? (Image, of course, renders a bit more quickly than pulling up a canvas)
 - If an annotation rather than a property, we can be
 - Some concern that this is overly complicated to add a property and motivation is smoother
- Proposal to add motivation with a better word for “postering”
 - 23 +1
 - 2 +0
 - 0 -1
- <https://github.com/IIIF/api/issues/1259> Thumbnail-navigation viewing hint for ranges
 - Easy to do with WebVTT and spritesheet - How do we replicate in IIIF?
 - Could have a top-level range that, for example, breaks video into 10 second chunks with thumbnails. Wouldn't want client to display it as a table of contents
 - Proposal is to add viewingHint on ranges to indicate client behavior for display of ranges w/ thumbnails
 - Questions/Discussion:
 - Do we need to do this in IIIF? Could we just rely on WebVTT sidecar and not deal with it in IIIF Presentation API? Are we reinventing WebVTT?
 - Not all players may understand WebVTT or this use of it
 - VTT will only represent video file - may want to have thumbnail navigation for more complex canvas
 - Want to bring this behavior into IIIF
 - Could generate very long ranges
 - Looking at issue this afternoon re. allowing externally-referenced ranges: <https://github.com/IIIF/api/issues/1218>
 - Clients don't have to implement
 - Will there be levels of compliance for Presentation API, as we have now for Image API?
 - Existing (closed) issue on this: <https://github.com/IIIF/api/issues/76>
 - Probably can't be levels, but perhaps profiles that correspond to sets of features
 - Proposal to continue to discuss
 - 17 +1
 - 2 +0 (contingent on external range)
 - 0 -1
- <https://github.com/IIIF/api/issues/1264> RenderingHints as applied to duration canvases
 - Example: may want to show something like a continuous image of a piano roll as well as the sound of it playing

- Clarification about rendering hints
- If you had a canvas in a manifest with a paged viewing hint and the canvas only has duration, then you present ; how to have an audio page? Don't do it, because it is meaningless; disallowing some combinations; the number of things the client has to do or decide goes;
- See issue for proposed semantics:
<https://github.com/IIIF/api/issues/1264#issuecomment-334216276>
- Questions/Discussion:
 - Example: disallow auto-advance for manuscripts; if you wanted a self-paced reader, you could do that with duration, then use auto-advance
 - It's a clean way to help the clients; very flexible and powerful (e.g. like with game agents)
 - Content gets there through annotation; behavior gets there from renderingHints / viewingHints
 - Like issue discussed earlier, you can annotate but not paint or create a new canvas
 - Would have to be MUST or MUSTNOT to be helpful for client (very MUSTY)
 - Are we leaving out too much that we don't know, yet?
 - This would have to likely be 4.0
 - A set of hints that only apply to one type of dimension? Necessary?
 - Perhaps add column to table in spec about which values are appropriate for which canvas dimensionalities (is that a word?)
 - Should you be allowed to say something meaningless? Should the validator give an error or a warning? Error.
 - Hints become directives if they have errors?!?!? Are they really much more deterministic?
 - Things like paged you could put on manifest or a sequence, and it just inherits
 - Entirely presentation logic of viewer not a user question
 - If I have a non-auto advance client, and it has an auto-advance, it can be ignored, which is why "hint" has been used in the past
- Question what to rename viewingHint? Both viewing and hint in questions
 - renderingHint
 - renderingDirective
 - renderingBehavior
 - renderingGuidelines
 - requests4Behavior
 - presentationBehavior
 - viewerBehavior
 - clientBehavior
- Agreement for:
 - Worth pursuing discussion

- Need to be cognisant of *breaking* issues
- Naming discussion moved to github

Annotations

“What does it mean to be alive?”

- <https://github.com/IIIF/api/issues/1257>, <https://github.com/IIIF/api/issues/1256> - text rendering issues (in AV because captioning, subtitling)
 - What should clients do with text annotating entire canvas (perhaps for limited time duration)?
 - Does it scale/stretch the text? Not appropriate for captions (obscures image)
 - Need to be painted but where and how big?
 - Impossible for client to do anything reasonable without more information
 - Close #1256?
- <https://github.com/IIIF/api/issues/1258> (and #511) motivations for associating text and canvas; captioning
 - Use case: you have many linked annotations; some contain captions in a variety of language and you want them to be surfaced easily and could also include others
 - Currently you have to load all annotation lists and inspect each
 - Notion of being able to know what entire list of annotations is from more than label, so you can generate better user experience
 - A possible solution: to consider “captioning” as a motivation
 - Web Annotation spec defines [12 high-level ones](#)
 - Note: does not include painting
 - [Extensible](#)
 - If you have a new motivation, should you require someone to also include something from the base list from Web Annotation or have two motivations on the annotation?
 - “motivation” : [“painting”, “captioning”]
 - “motivation” : “captioning”
 - Are we then getting into semantics in the presentation API? Oh, no! Not, because this is about presenting the info.
 - Summary of interrelated parts:
 - Anyone can make up a captioning motivation - community norm
 - Does captioning get called out in the spec as something that clients should respond to? Client behavior may vary
 - If calling it out, can you somehow from the manifest convey what you need without loading all the annotations
 - Strong use case for captioning

- Stronger than things like transcription, because of legal requirements
 - Viewers could establish a community practice to enhance user behavior and meeting expectations (like UV or Mirador)
 - Can we implore a test for including motivations? Captioning means the same across disciplines; whereas transcription may mean different things across disciplines
 - Do we have a place to put this outside of spec, like vocab? Namespace-wise it goes in SharedCanvas
 - Is captioning is important?
 - Majority +1
 - 0
 - 1 -1 not convinced that captioning is more important
 - Is it a recipe?
 - What we're really talking about is a prefix on a URL?
 - May be more useful to have a list that isn't depending on version of the spec
 - If external, would have to say both painting and captioning
 - If this was in the spec that clients MUST render captions ..., will the viewers be able to implement before next Dec. If not, then Prezi 3.0 is held up.
 - Implementation could be trivial with conversion to VTT
 - In context but not in spec?
 - Then create a community maintained process that is separate from spec writing (like profile process for things like external metadata resources)
 - Agreement
 - **Not put in spec but give it a prefix**
"motivation": ["painting", "iiif-motivation:captioning"]
 - **Create an issue to extend to a set of motivations**
 - **An accessibility-related document accompanying spec (recipe in cookbook)**
 - **Captioning needed but outside of spec**
 - **Discuss process, governance, committee to manage/test *things* outside spec (profiles, term sets, etc.)? #1284**
 - Some annexes may be replaced by cookbooks and others not so much
- <https://github.com/IIIF/api/issues/754> Summarizing various features of annotations in the annotation page or annotation collection.
 - Example: set of captions in dozens of languages. If each caption has individual motivation, need to process through all of that in order to present language option to user in UI
 - What features would need to be summarizable? Language, motivation, ... - is this a bounded list?
 - **Defer**
 - **It could be that we don't have this complex of a use case, yet**
 - **It is a non-breaking change and could be deferred to 3.1**
 - Ask the user to load?

- Are there specific accessibility requirements for the client to load captions? This could be solved with the *motivation* on each caption rather than the summary.
- Common use case: I need to provide access to captions in a particular part of the UI. In order to meet that requirement, the client has to dereference all and then put the [three] that happen to be captioning go in that list
- Is there a way to use layers instead? They become annotation collections. Can't have a motivation on a layer
- Summarizing is an optimization
- Could a viewerBehavior on the annotation collection do this instead, if only needed for specific case of captions?

Recommendation to discuss and clarify these notes on upcoming A/V TG call (on Oct 31st?).