Metadata Model

EaaSI Webinar Series

September 2019

Ethan Gates: Let's just get things started for everyone here today. So, welcome everyone. Thank you so much for joining us today for our Eaasi webinar, and our latest in our ongoing series. My name is Ethan Gates, I'm the software preservation analyst here at Yale University. For those who aren't aware, at this point, Yale University is the host of the Eaasi Program of Work. Today we are talking up the Eaasi Metadata Model. It's going to be a round table discussion between Michael Olson, Seth Anderson, and myself. So, a little housekeeping before we get started on the technical side. If you have any questions during this presentation, please type them into the chat box in your zoom control panel. Jessica Meyerson is going to be hanging out in the background watching that chat, especially to be sure that she can gather up your questions and bring them to me, so that we will have some time for our Q & A at the end, and we'll be sure to address those. If you can also please try to mute yourself and turn your video off if you haven't at this point already, to help us maximize the quality of the recording. All the webinar recordings for the Eaasi webinar series are going to be made available on the Eaasi website with transcripts once the series wraps up. So, now I'm going to introduce our guest speakers, the members of the Eaasi network for today's presentation. First up we got Michael Olson. Michael is the service manager for the born digital forensics lab at Stanford libraries. He works with librarians, faculty donors, and archivists to develop policies workflows and procedures, to acquire, preserve, and make Orin digital content available for researchers. Everything comes in threes for Michael. He is currently responsible for the implantation of forensic tools and library workflows, defining security policies, and audit processes for acquired content, and automating processes for depositing preserved software into the Stanford digital repository. Michael has a Master of Philosophy in history and computing from the University of Glasco and a BA in medieval studies from the University of British Columbia, I did not know that Michael, we're going to have to talk about that at some point. Our Eaasi staff lead for the day is Seth Anderson. Seth is the software preservation program manager at Yale University library where he oversees efforts, collection, and preservation in software resources as well as tools for access to preserve software additional collections. Seth received his master's in moving image archiving and preservation from New York University. He has previously worked as project manager of the museum of modern arts, electronic records archive project, and worked with Carnage Hall, the Smithsonian Institution, and the United States Holocaust Memorial Museum as consultant with AVP. So, with no further ado, it gives me great pleasure to hand it over to Seth for an overview of the Eaasi Metadata Model and the role of Wiki Data and the Eaasi Metadata universe. I mean it is a universe, like a galaxy. So, take it away Seth.

Seth Anderson: Thank you Ethan. I appreciate the great pleasure. Alright, really quick, let me share my screen, and let's go full screen. Alright so, greetings everyone, thanks for attending our webinar and if you've attended in the past or even if this is your first one, welcome we're happy to have you. So, yeah, I'm here today to talk about the metadata model that we'll be applying in the Eaasi software, that will be implemented in upcoming updates to the front end of the system.

If you've attended any of the webinar's in the past or have seen us talk about the project previously, you'll remember that one of our, one of the main pillars of the project is to take a look at the descriptive practices for software and preserved software. As well as computing environments in order to improve and expand upon mechanisms and best practices for describing these items that we work with so much in the Eaasi system. So, for our purposes we collect metadata for a number of different reasons. Of course, we are collecting metadata to improve discoveries so, as we add more and more resources to the Eaasi system and the Eaasi network that is shared, you know the resources that are shared between the nodes, we want to make sure they're easy to find and that you can perform simple key word searches and find a software application that you need to render a file that you have or to find an environment that already has that software installed, or an environment that could serve as the basis for new software installations etc. We also, of course, want to track all of the resources that are going into the Eaasi system and into the Eaasi network so, making sure that we're documenting, kind of whatever, we call the provenance of where these things are coming from and where they're going, so that overtime and for digital preservation purposes or for auditing or reporting we have the information that we need. We're also of course collecting information just to administer the system in general. So, making sure we have the necessary information for tracking, or knowing what users are up to, knowing what resources are in and out of the network, what is and isn't deleted, all, you know all of those details that go into running a system like this, are kind of getting poured into the model. And finally, and I think this is the biggest and most challenging piece of developing and implementing the model, is that we're looking at ways of using the metadata that we do capture as a means to automate certain workflows. And the goal with the Eaasi system is to, is yes to provide access, Eaasi access to emulators, so that you can configure them and use them in your workflows, but we also want to make it as easy as possible, no pun intended, to get from point A to point B. So, instead of requiring a user to always do the leg work, we want to see how we can use the metadata that we capture to move us from, say having a file and getting to the software that we need as quickly and easily as possible. So, with all of that said , alright I'm going to ignore the chat…what is in the model? There's a lot, but I want to start by taking a step back to just talk about, of course, what it is that we're emulating and how that applies to metadata. So, right, if we are looking to recreate a physical computing environment virtually in our software. So if you think about a physical computer, either the one in front of you or one that you had in the past, any computer you can think of, there are a number of different components that all intertwine to create this computing environment, that we interact with through the tools and with Eaasi through our browser. So, you might have a monitor, you have a mouse, you have the processor, different cards that can manage the visual capabilities of the computer, the audio capabilities, the networking capabilities, etc. Even of course have the bootable elements of the computer, so the, within the environment that contains a bootable system which would be the operating system and all of the various component parts of that. And of course, you have the additional elements that store data that needs to be interacted with through the system, so you might have floppy disks, optical disk like CD rom or DVDs, and other versions of hard disk storage so external drives, USB drive etc., etc. and of course, we are looking to emulate those and so those get reinterpreted as, for the hardware piece an emulator, software configuration so telling the emulation software what devices to emulate and how to construct them in this virtual computer. You have rebootable systems, (inaudible) emulator. And then all of those additional storage software content items that get loaded into the computer,

which again are disk images. And so, in this model we have to describe all of these pieces, both as kind of conceptual elements and their sort of applied version. And so, if you think about this the scale gets quite big because you're often duplicating information and connecting all of them through relationships is a challenge, but interesting in my thought. So, about a year ago, probably about six months into the project in June of 2018 we began sketching out what this model would look like. As we started to realize, there were a lot of complexities and details that we needed to start capturing in order to achieve some of the goals that we had. And so, we looked at existing schema and some of the great work that's been done by others in the community, one in particular, while being the totem project and while we haven't used it as a bases, these projects inform of course our approach. But we have also looked at the game set project out of, that was at Stanford I believe. As well as some of the cross-walking work that was done by one of the working groups at the software preservation network. And then of course we have a team member, part of the Eaasi project team, Cat Thorton, who has started this Wikidata for digital preservation project, which proceeded the Eaasi project and she's done a lot to define models for describing software versions and file formats with the goal of using those within Wikidata and we're intending to recreate those within our model so that we can exchange information all together. So, I'm not going to dive into the nitty gritty details of the model, I will show you what it currently looks like in my modeling software. This may be an indictment of my ability to build a database or might just give you an idea of how complex this all is, but if you boil it down to the simple details, it essentially tracks across the types of resources we use in the system. We have software objects, being the operating systems applications drivers etc., that are applied in some way in computing environments or emulating environments. We have those environments which we describe and then we have content objects which can be joined with computing environments for rendering and access. And I'm going to quickly, because I know that I am going to run over time, move through these. So, we break down software generally, we adopted an existing Ferber based hierarchy for describing software and are applying it in our model and essentially it moves from the four tiers of Ferber's, so you have a work expression manifestation in item. In order to separate out various attributes of a software application from its kind of highest-level conceptual instance to its lowest level, kind of applied and distributed instance. So, in the system we have this concept of the software product, which is kind of the high-level grouping of specific software releases, so you can think of Microsoft word, Adobe Photoshop, Libreoffice, or yeah something like that. Those are then expressed as versions, so all of those like Microsoft Word has Word 2000, Word 1997 and so on and so forth. Those are then manifested as either an object, so that would be in the physical comparison, the installation media, the disk, the CD Rom, the floppy disk etc.; required to install those items. Those could also just be files, that are needed to compile and run that software version, and then once installed we have this manifestation of that version as a configured piece of software within an environment. And then, the lowest level of course, being all of the various instances of those software objects or configured software within the Eaasi network, so of course there will be one item at Yale, maybe another one at Stanford, and so on and so forth. So, if you look at this, I'm not going to, I ran out of room to include the item, but if you think about this just as an example, you have autocad at the top level. Autocad of 2000 as a version or expression, and then you have the installation disk as a manifestation, and again once installed it is a manifestation as well of the same software version. So, this way we can structure the software so it inherits necessary information and can kind of compile all of that to display it to the end user in a convenient and user-friendly way. So,

I have some definitions here which may not be as helpful, but I do want to give some example, whoops, Jessica or Ethan remind me when I'm running low on time.

Ethan: I want to bring our discussion around at 11:20 so if you can, wrap this to your convenience.

Seth Anderson: So, an environment is a combination then, of a collection of installed software, so an operating system in a software application and a configured machine. So, a defined set of values that are related to the actual hardware of the computer that's being emulated. And then once you combine all of those in the emulator software, that is what we consider a computing environment. Are we having audio issues? Okay. And, these slides will all be shared of course, so, and I'll also be sharing the document spreadsheet that has all the definitions and the metadata models, so that will also be available. So finally, the last kind of conceptual area of course is the object or what we are now kind of calling a content environment, and that's just very simply, just the connection of a file or a constant object that's been added to the node and an existing competing environment so it can be rendered. So, it's like, just like a simple connection between the two. So, I will fly through this, but just to give you, and we'll talk about this a little bit later, but we're doing a lot of leg work up front to populate the system. One of the things that we're trying to avoid is user fatigue. So, instead of requiring everyone to key in a lot of information we want to prepopulate the information, so that you can add drop downs and visual cues for selecting the settings that you defined in your computing environment. So, in one way that we're doing that, is we're pulling data that already exists from the Wikidata knowledge base and using those as controlled value sets within the data base. So, for things that are relatively static like CPU architectures, file formats, keyboard layouts, etc. we're pulling that information in and we'll periodically update it with new information that Cat Thorton will be providing us in populating the database and other database's in the network, to have that information to apply. Also doing some upfront research to define, for instance, what settings are available within the various operating systems that we have in the network, what software applications support specific formats, or defining in a structure those elements. Like how they define them and what formats those are tied to, what hardware and emulator can actually recreate and defining that within a structured data set, and also structuring system requirements from the applications that we are loading in from Yale's site. And finally, just some data captured methods, so if you haven't checked out the Wikidata for just digital preservation portal, this is one mechanism for adding information about software and final formats to Wikidata. We will be encouraging the users of Eaasi to use this to update information about software once it's been published, the network, there are some rules that we'll be applying, just for quality control, but we want to push folks to use this portal, because of the great work that Cat's been doing. But we will of course also have some editing modes for the new UI to update information about software and environments that have been configured in an Eaasi node. And, we're also working on serialization of the metadata that we capture both for exchange between the nodes, but also as a means of submitting or importing data into an Eaasi infostructure without having to do any, like, data entry or filling out all of the various forms that we'll have in the system. So, trying to speed up… and that's time Jessica says. So, challenges, I've got one more, I think this is the last one. Controlling data quality, of course, big one is the structural differences between operating

systems, computer platforms, so the different ways that different computers and computing environments actually use information or structure information makes it really hard, to kind of create a universal model for all computer types. Especially as we get to older computers, these things change a lot. Demands of specific metadata about software applications and how they're used. And, making this easy for users. So, I've gone over time, so please ask questions, there's plenty to share.

Ethan: Yes, that was awesome. Thank you so much for that overview, Seth. As he mentioned, like if any questions are already coming up from the group, from that overview, I mean please just put them in the zoom chat. We'll be sure to try and get to them in the Q & A session at the end here. But, right now I do want to transition into our moderator discussion with Seth and bring in our node host, Michael Olson, so that we can ground some of this high-level conceptual overview that Seth is talking about in everyday practice at Stanford and at Yale. So, first off, I do want to just directly bring Michael in and if you could talk about how, do you at Stanford, you know create, manage, and discover metadata about software and related resources right now. What is your state in terms of implementing the concepts that Seth has been discussing?

Michael Olson: Hi everyone. Just give me one second here I'm actually going to share my screen. Good morning everyone. Thanks Ethan, thank you Seth. I just wanted to transition a little bit. This is a little bit of old data that we have from a disk imaging software preservation project that we ran roughly four years ago. It's a spread sheet, surprise, surprise. But, a pretty masterful one that was actually created by our metadata department and our arcadia outcome amongst others. But, as you can see, there's literally, you know we have genres, and there are over eight thousand lines, and this is just one collection. So, our metadata creation process currently, as it currently stands for software, and this is primarily because a lot of our software actually comes in through our special collections workflow as opposed to a regular cataloging workflow. We do receive some software there, but the vast, huge sum of it, probably comes through a special collection's workflow. So, we're not, we haven't in the past necessarily done item level metadata on all of the software we received, we do, do it for some. But this is, this is how we've been capturing our data for the (inaudible) project. And, it's a little, a lot, a lot of data, but as you can see here, we've been trying to figure out…we have genres. This is a lot of the, the publisher data, and then, well if I can find it here real quick. We have manufactured data. I'm sorry if I'm making everybody sick. But I though giving some really good examples here of, and then more importantly we have operating systems that you can start to see here. So, our workflow up to prior to emulation as a service coming online, has been primarily to capture, to capture our descriptive metadata for software in a spreadsheet. That's what it looks like when we create it. This is, then of course we've had this whole process of importing our descriptive metadata, along with the digital objects that we're preserving into our digital repository. So, this is the view of what some of these titles look line actually in our repository. What's, what we're missing right now is a way of actually delivering a lot of these contents directly from our catalog. And that's where we're really hoping that the Eaasi emulation as a service can really sort of help us, actually be able to better identify string in the catalog record and then will take some folks directly to the actual software, running in its environment. So, we've got a lot of questions about how do we go from some of these systems you're seeing here, in the Eaasi environment and Seth has started to talk a little bit about some of the different, how we are envisioning this going

forward as the work evolves and develops overtime. But this is just kind of a little bit of context. We're not starting with nothing that's catalog, we're kind of hitting it mid-flow and we have lots of questions and challenges to work out with our other, with our other node hosts, and with all of you to figure out, you know, what are some different workflows for actually streamlining and getting content that's already described, described in Wikidata, and into the emulation as a service environment. So, I'll stop there, I don't want to take too much time for folks, but yeah.

Ethan: That's a great, I think you've brought up a really great point here, in sort of like the interplay between Eaasi and local descriptive practices that might already exists. And particular looking out at the data you were sharing, you know to me, it's like the source and ID fields stood out because this is a challenge that I know we've had working with Eaasi at our Yale node. Around naming conventions because Eaasi from a technical backend perspective, basically assigns a unique identifier to every saved and configured emulated environment, it assigns them to individual disk images, and associated software objects, so from a backend perspective, all Eaasi needs is these, UUID's Universal Unique Identifiers, in order to assemble the pieces it needs and move them around. But that obviously is not very friendly on the human end, in terms of finding and discoverability. So, I'm curious how, both Michael and Seth, you see this moving forward, whether it's in the new UI work, how do we make it so that, when we're talking about pieces of software or configured emulated environments that potentially have extremely small difference between them, we can gather all the metadata we want, but how do we define and surface those differences and make them discoverable, the particular objects that students and scholars are actually going to want to see findable to them. If anyone has an immediate thought on that and just wants to jump in.

Seth Anderson: It's, it's a question that we, we struggle with a lot, especially with, I mean I think I've titled previous talks around kind of what all these significant properties of a computing environment, and what do we need to record that's useful to a user. Because I think what we've focused on a lot is, what's actually useful to the system. Which, given the kind of parameters of a project makes a lot of sense, and I think we won't really know until we see more use of the data model and the system and kind of surface what it is that users actually need to know to find a computing environment and software object. We're also looking at, and we're working with (inaudible) Media out of Madison's Wisconsin as developers, who are designing and implementing this new front end, we've talked about ways of kind of doing, providing visual comparisons of environments, and once we have information, more information about the environments, we could, I think potentially, yeah guide users a little more effectively and saying, the thing you're just about to create, probably already exists, you know is this the environment you're looking for and kind of surfacing ways that we can provide a more guided experience. It's a little challenging now, with the kind of lack of overall information that we have already collected.

Ethan: Yeah Michael, where does the identifier system you were using confirm exactly, and then how successful has that been with integrating with your catalog systems?

Michael Olson: It's been, apologies, I was just booting something up there, in the background. It's been, and you know we're making what we think are educated guesses as we're going with

how we construct identifiers and things like that. The obvious ones that most librarians are aware of in preservation circles, like you know no spaces and (inaudible) names and that sort of thing. I think the real challenge is not only with figuring out how we're going to interoperate with these multiple systems, but I'm going to share something really quickly, because this might actually spark some, spark some additional discussion at some point. And that's, we're not just talking about the executable software running in an environment. There's all the paratextual materials that actually go with it as well, so I'm going to show you what that looks like for just one example. So, you can see here, this is actually a CD Rom collection from the IMF. So, it dates from the early 2000s, roughly, yeah late 90s early 2000s. But there's all this associated data that comes with it that we're not, you know we think we know how researchers are going to actually access this and actually use it, but it's a little bit difficult, and this is maybe not, this is a specific example, but you can imagine it as like a video games, where there's a lot of box covers and associated documentation that comes with it, and fan stuff. So, its, that's definitely one of the challenges is how do we actually integrate this sort of material into the delivery environment, or make it available, how much work does it actually take to do that. And, more importantly who's responsible for it? We're actually really hopeful that we can push some of this work in the Wikidata and the emulation as a service environment, up to librarians who have a better, or a little bit closer to the actual researchers so they can help make those decisions.

Interviewer: Did you have a thought there, Seth?

Seth Anderson: No, I was trying to piece it together in my head. I mean just looking at those items I mean the interesting piece here is that those already have existing identifiers, they belong to another system probably, and so one of the challenges again, that we're facing as well is just think about interoperability. We have some plans and preparation in the model right now to digest local identifiers so that you can connect this object from your collection to the virtual version in Eaasi. But, that's still a somewhat tenuous relationship because they're currently completely separate systems. And so, I think one thing that we'll be looking into coming down the road, is just how, how to more tightly connect these two existing systems? We've already done it in one instance with Yale's Preservica Repository, so that a lot of these questions around like how do you identify and manage all of this different information, can be tied into, existing structures and infrastructure at all the various institutions who are using Eaasi.

Ethan: Yeah, drawing off of both of those points I'm really interested because Michael you are bringing in the other question of like paratextual sources and where we source all of our metadata from, you know speaks to the possible convenience and real advantage of using Wikidata in terms of centralizing some of this, you know often scatter, like a lot of this information exists but is just scattered across many places, whether it's local catalogs, internet, etc., etc., physical objects. So, that, again, speaks to the building of the communal resources, but then Seth, as you're saying there is this interplay between possibly locally unique information and, you know Eaasi itself having its own peculiarities within that metadata model, that you demonstrated that not be, that are very important for building an Eaasi database and discoverability within an Eaasi system, but may not play with or be, may not justify inclusion in an external data base like Wikidata. Especially when we're talking about derivative environments, again with these possible minor changes. Can you think of strategies for, like how

do we address this within, within the wider community? And like how do we raise awareness for the need for these small changes, you know within Eaasi? While, you know being flexible for vocal changes. That's a big question.

Seth Anderson: Yeah. I think, I mean our hope is that by tying Eaasi, I mean not tying but, coordinating Eaasi, I guess. I don't know what the best way to put it would be, with the Wikidata and Wikidata for digital preservation project, that you know, it also serves as kind of an advocacy element to this. So, we want, not just Eaasi users to supply information to Wikidata, but anyone to, you know, if you have information about software, use Wiki, the Wiki DP portal to put it into Wikidata so that we can, we can reuse it and vice versa. I think we are really hoping that we can act as valuable or instrumental members of the Wikidata community by providing the information that we gather in Eaasi, back out to, Wikidata, and you know create this interchange that would be valuable to both of us. But, yeah, I think our hope is that by gathering more and more information that we kind of raise more awareness of the need to document these things, focus the community on pursuing better practices and best practices for documentation of these items.

Ethan: Yeah, I want to, I want to push to Michael because I'm wondering if there are strategies or like lessons learned from starting to gather the metadata that you have in the first place. Surely, that was a you know point that had to come up at Stanford. Is there anything you can take from those sort of local processes and apply to the network within the larger community?

Michael Olson: Yeah, that's a great question. I think that one of the, well I'll back up just really quickly. A lot of the data that you actually saw is actually going to end up in Wikidata, kind of through a secured route so, so we're, I think there's definitely a desire on behalf of the different nodes that are a part of this initial rollout for Eaasi to try and populate as much of their content as possible into the Wikidata. Just to be able to make it available to the community so people can react to it and interact with it. I think that's really key. The other thing is I think that going forward we're definitely going to, we're talking kind of a little bit about legacy collections and things that we've already sort of had a processing path for and I think going forward we definitely have a bunch of different collections kind of cued up, where we're going to try to figure out what the workflow looks like natively. We're actually creating the metadata and Wikidata and that sort of thing. So, that's about as far as I can think right now but thank goodness, we have lots of brains on the call that can help. So.

Ethan: Yeah, we'll definitely be continuing to crowd source this discussion. But yeah, I wonder if Seth you could build a little bit more on, you know, you're sort of implying the value that this metadata might have to the wider community as we build Wiki DP. I mean what is, what is the value possible to the community, the digital preservation community, beyond building out Eaasi, in terms of the metadata we were looking at in the model?

Seth Anderson: Well, I think, I mean if you look at emulation as an access mechanism for preserved collections, so for digital collections, you know if we get, OISE, if we have representation information, which would be the software that's required to render these materials… the more information we have about that software, means that we have kind of better

digital preservation packages about our digital collection. So, by consolidating and structuring that information, we can apply it, not just in Eaasi but across other programs and systems, so that, there's kind of a second order value to all of this and we're not just creating another silo I think is the general goal.

Ethan: Yeah. And building on that point I would also want to bring up our, the work of our summer intern Clair Fox, who is with the program. She's also from New York's Universities moving image archiving preservation program and if people on the call want to check out her blog post about the work she did over the summer. Preparing like what a preservation package from Eaasi would look like, in terms of like exporting out data so it could be reused in other systems in combination with the metadata that Seth is talking about, that's really a fascinating post that's worth looking at in terms of this discussion.

Michael Olson: Yeah if I could just jump in--

Interviewer: Please.

Michael Olson: With one really quick comment. One thing that's really obvious even with the IMF, the International Monetary Fund's CD Roms that we're talking about, is we were actually able to, some of the work that was previously in the system, that was provided by the Yale node, was something we could leverage to actually make that available. So, you know there is value in creating base environments and for making that and describing that, making that available through the description in Wikidata and stuff like that. But, there's already value there, just because their base environments that we can grab and actually see if it works.

Seth Anderson: Yeah, and I think, you know that, the underlying, I keep saying the underlying goals, but we have a lot of them. Of course, is kind of this community approach to software collection and that, that's a perfect example of how, the metadata that we've captured or the, you know software we've already configured has been of use, at Stanford. And, before we move on to Q & A from the Yale end I would just want to point out, that so much of the information that we've already captured, and that will be populating the system, soon-ish, would not exists without the great work of our student software configuration workers. So, all of those fields that, you know, we've been preparing in the model, students have already been using that model as a means of capturing information as they go about installing and setting up the software in the Yale's Eaasi node. And it's not trivial, as Claus would say, and they've done a really great job, and yeah, we really appreciate them. And we are thrilled the we have many new students joining our team, this semester.

Ethan: Yeah and I'll definitely be diving a little bit more into their work next month for the configuration webinar. That work will come up there as well. But, yeah, our students, which who run the gambit, from freshman to masters and grad students have all been fantastic. Yeah, I think that's a great point for us to transition into our question and answer period from the audience. We've already had a great discussion going on in the chat here that I want to bring in. so, Cindy asks, what was the decision factor for choosing Wikidata and Wiki DP over Pronom, the Pronom registry. Considering, especially that premise, the preservation metadata schema that's primarily

widely adopted really pushed Promon. And we had a response from David under down that could probably cover some of it. Saying, that a lot of what's needed for Eaasi goes beyond the existing Promon data model, but yeah, I think that's true. And Seth maybe you want to talk about that specifically.

Seth Anderson: Yeah, so I wouldn't consider it an either/or question because we do intend to incorporate Pronom ID's for file formats into the model. And, essentially, I mean the benefits of Wikidata over, you know, is that Wikidata serves as, to consolidate information like a Pronom ID, as well as say a mind type or any number of registry identifiers under one record for a filed format, or a software application, etc. So, it still incorporates Pronom, but it incorporates as well as a number of other useful and valuable information points, you know, that can then, we're linking all of this information that's the idea. So, that's why going with Wikidata.

Ethan: Great, thanks Seth. Pretty straight forward, I don't know if Michael has anything to chime in, otherwise I'll just move on to the next question. From Patricia, we've got, is Eaasi looking at custom written software, for instance artist software or scientific research software, and how would metadata, you know for those particular communities possibly be different? How could it fit into our model? That's also a great question, if –

Seth Anderson: Yeah, so, so I think that I glossed over this in the challenges slide, but domains specific information is tricky of course and the challenge with creating a model for Eaasi was determining, at this point, I mean we may expand on it and extend it a bit, for specific disciplines. But, right now, its intended to be as universal as possible so we also didn't want to step on the toes of other systems, so the focus of the model is what specifically is needed. One, to set up an emulation, but also what information is most useful, you know across the board, so universally, about software, for discovery and fact finding. So, it's not kind of always want to keep pushing that like, Eaasi isn't your collection management systems, so there may be a lot of information that exists in other systems that would be helpful to how you would set up an environment. But, for our purposes, our primary focus is on, you know, the simple information about the software and then information about its functionality and its requirements.

Ethan: Yeah, that's good. Michael, has this come up at Stanford? In terms of gathering software in metadata, and you know working with possibly domain unit department even level specific metadata. People wanting to learn certain things for their own needs.

Michael Olson: Yeah, definitely, so I mean I can't speak specifically to the nitty gritty details of each sort of potential use case for how software can be represented. But we've definitely been working, one of our efforts has been to work with, to try out as many different use cases, as we possibly can, using the existing Eaasi environment. So, to give an example something that's a little bit different. One is we're working with the university press here, so digital publications that tend to be, well they exist just as digital publications, and the technology stack that these researchers are creating, tend to be quite custom and that presents some interesting challenges particularly because the technology is moving so fast. We're also, some of the use cases that we're hoping to test out in the emulation as a service interface, and we're just really starting this work, some of our use cases are research projects, so from a center of interdiciniplary research

here. So, interesting things that some of our humanities professors are working on and of course we have scientific data sets as well, so that's a whole other challenge. So, yeah, it's definitely something on our radar.

Seth Anderson: Yeah, I think it would be useful to describe, I mean the intention or the scope of the Eaasi system verses what will be accessed modules or access services built on top of what Eaasi generates. So, I think like the primary system that we focus on daily, here at Yale, and what we're actively working on and developing and then upgrading to front end for, is a tool for staff at the library, at the university, whatever, to generate the emulation environments. Which are then provided to an end user patron, whatever, through another service. Now, we're working on developing some of those access-oriented services, but they are separate, you know conceptually from Eaasi, as a system. Which is why, if you look at the model, once I share the data, when it comes to the actual content object that's loaded in, so the disc images are files that are meant to be rendered in an emulation environment. We don't actually capture much information at all, so no descriptive information, just a name and the local identifier, and then maybe some technical information about the actual objects. You know, what type of media was it originally, is it a file set, is it a CD Rom, disk image, etc. and then, we would rely on another catalog or collection management system or some access system to provide additional contextual information to the end user. That raises certain challenges we're trying to of course address, but that's the idea.

Ethan: Yeah, it's really interesting Seth. I want to point out to the audience that we're sort of moving into our last few minutes here so if there are any more questions please put that into the chat and we'll make sure to bring it up before we wrap up here. But, this tension, that we're basically talking about between designing a system that works for many people but can be customizable to the few, think is a really difficult one, that is not unique to Eaasi. So, I'm trying to think how to frame that into a question, but it's not, like boy, isn't what we do hard. But, can, let's see we have, I do want to point out for maybe those who are just listening in on the call, that Jessica posted in the chat that in a previous software preservation network quarterly forum, Patricia from the Tate, Dana Boquin, Claus from our team here at Eaasi, talked about how at a certain level use cases require the same metadata just to get these environments working, but then there is this additional level that Seth is talking about. The level of the software objects themselves, or collection items, that these become local policy concerns. And, you know this is another place to point out that while Eaasi manages creation of emulated software of any of the environments, and of software collections we do emphasize that we, the system does not necessarily manage digital and collection items and wasn't intended to. So, how we can like build a system that interoperates with digital repositories that are a little more set up, you know to actually perform those duties and gather metadata about those kinds of objects, is a really interesting question. Are there, what kind of like systems and integrations would, you like to see perhaps Michael like from Eaasi in terms of how, you know Eaasi metadata interfacing with, with other systems or services that you use.

Michael Olson: Yeah, its, it's really a challenge and I mean I think that's one of the challenges that we're working on right now and that's why when we were standing up our Eaasi node and how we kind of constructed our participation from the Stanford side of things has been, we want

to pilot different collections that have, there's a little bit of a different flavor, with the realization that not one system can do everything. And, that it's going to have to interoperate with existing systems that we have here for descriptive cataloging purposes etc. etc. but, you know, we realize that these integration points can sometimes be extremely difficult to do, so we're going to try some out, we're going to use some very specific collections, run them through the systems, see what the catches are for actually being able to integrate something like Eaasi with our existing catalog system. So, yeah, we don't, a lot of questions at this point, not a lot of answers, but we realize this is an ongoing challenge.

Ethan: Ongoing challenges feels like a, a bit of our motto.

Michael Olson: Yeah but it's really cool I mean--

Interviewer: Yeah.

Michael Olson: We're really making progress here, I mean I really want to stress the folks that yeah there's a lot of challenges that are articulated in this, but at the same time we're getting there, we're making progress. And, yeah.

Ethan: Coming up with tools to address them, as we improve the tools will be the ongoing challenge, yeah. This is really great, honestly, we're starting to get some people dropping out of the webinar, so I do want to sort of move into wrapping us up here. This has been an awesome discussion. So, is there, Seth and Michael are there any sort of like lingering points here from our discussion that you want to be sure to put in front of people before we start closing out?

Seth Anderson: I would just say that I think, once we've, so I just wrapped up an update to the data model which was sorely needed as we move into actually developing the front end, and I will hope to release that, or at least post it on, through our OSF site and will provide a link to that in any sort of follow up posting about the webinar. And, I would be curios to hear feedback from anyone on the call, in the community. You can tell me if I've put things like, you know, the GPU cards IRQ setting in the right table, what have you, these are the details that keep me up at night or keep me up late. So, I welcome more eyes and more input so, please take a look.

Ethan: Thank you Seth. Yeah, so everyone keep an eye out on the Eaasi OSF site, keep an eye out on our project site, on the spinsoftwarepreservationnetwork.org, and our social median through software preservation network. This has been fantastic and be sure everyone, you know, to join us next month in October, for our next and final, final unfortunately, for now, episode in the Eaasi webinar series. That's going to be entitled Eaasi Configuration Workflows and we'll be joined in that call by Lauren Work from university of Virginia and I will be the Eaasi staff lead talking about my configuration work with the students here at Yale that time. And Jessica Meyerson will take over hosting for that. So, thanks again to everyone in the chat for great questions, thank you for joining us, and we will see you next time. Thanks Seth, thanks Michael.

Seth Anderson: Thank you.

Michael Olson: Thanks everyone.

End of audio.