Interview

Toward Empathetic Digital Repositories: An Interview with Diego Pino Navarro

Sophia Ziegler

ABSTRACT

Diego Pino Navarro is a systems architect and open source software developer from Chile. He is the Assistant Director for Digital Strategy at the Metropolitan New York Library Council (METRO) where he manages the digital services team. Diego is also the Lead Architect of Archipelago Commons, an open source digital library software for cultural heritage materials.

In this interview from October 18, 2021, Sophia Ziegler talks to Diego Pino Navarro about his work with Archipelago, and specifically how he invokes the role of empathy in ongoing digital library software development. Diego talks about his efforts to build a digital library system that empowers local control of digital items and metadata, thereby maintaining unique characteristics represented in their digital collections, while simultaneously generalizing for various schemas. This empathetic approach requires resisting capitalistic impulses for continuous growth and prioritizing generosity and care.

Keywords: Open-source software, digital repositories, metadata schemas, local communities, empathetic practices

Editors' Note: This interview is part of an ongoing series meant to draw attention to current trends and developments in critical digital librarianship and should not be considered an endorsement of any particular tool, approach, or development philosophy.

Ziegler, Sophia (2022). "Toward Empathetic Digital Repositories: An Interview with Diego Pino Navarro." *Journal of Critical Digital Librarianship* Vol. 2, No. 1 (Summer 2022): 1-10.

INTERVIEW

Sophia Ziegler: Thank you so much for taking the time to talk today. Much of what I'm hoping that we can discuss is based on a presentation that you did at the New York City Library Council. I don't have a date for this, but I was able to find the slides through the documentation section of the Archipelago website. The name of the presentation is "Archipelago: An Empathetic Digital Repository." I was hoping you could talk to us about what you mean when you talk about an empathetic digital repository.

Diego Pino Navarro: That's actually a really good question. The concept is to be an empath or empathetic, my English is not the best there. It helps to know the origin, or the reason that we started working on Archipelago. So I'm going to do some background here. Basically I came to the US to work on open source digital library software. I was trying to un-jail a nonprofit from a very strict and complicated vendor. We had no capabilities to ingest our own assets, or even define or decide on our own metadata practices because the vendor was mediating everything. And for me it was very frustrating because I'm very naïve, and I grew up thinking about open source as an open community effort, not cold code encapsulated and black-boxed by someone who uses us as an advertising opportunity. And as time passed, I started getting very deep into open source communities. I became a release manager, code committer for that particular repository community's software and was trying to help and also learn from other people.

I started to figure out that open source, the way that I used to envision it, again very naïvely, has some deep flaws based on capitalism. People with knowledge of what was happening had full control over the future and the roadmap, everything. And the community was extremely unbalanced between a lot of users with specific needs and a very small group that's actually making decisions. And while these decisions are based on user needs to a degree, they're also based on existing vendor needs and fears. They're also based on a certain level of pragmatic self-indulgence.

I also learned the hard way that, and this is something that's complicated to become aware of, even if I'm a man and have that privileged position in technology, I'm a Brown man and I have an accent. I have ideas, and that's nice until I actually try to change things or have people think about change. Everything would be fine as long as I don't make too much noise, or don't vocally criticize the existing establishment. And as you probably know, software development is driven by a male, 'dudeness' attitude. There's a set power dynamic, and when you become aware of that, it's like 'Okay, well maybe I'm trying to fight a monster, and maybe the solution here is actually to go to the other side.' It's like, 'Why are we fighting a big monster? What are the alternatives?'

The software we were using was becoming a big monster. It was extremely complex, driven by technology instead of user needs. We decided that we actually needed to stop. We needed to stop and think: what are the core values of our work? So we stopped and thought about this for six months. And it's funny, I had this moment where I actually said, 'Am I doing this because I was not invited to the club? Or am I

doing this because I feel the club should be driven by the people who actually need this? Am I being generous, or am I being selfish?' During this process, we were talking to institutions here in New York City that need digital repositories. The city is huge but the majority of these institutions are single staff archives, sometimes underpaid with a lot of stress on their hands, and varying levels of digital cataloging skills and cataloging practices.

And when I say varying skills, it's not that their skills are less; their skills are different. When you learn to do things in a certain way it often comes from precarity. Which is something I knew well because I was coming from South America, where we have been doing everything, basically, with glue and tape. During this time I was also going to a few conferences and seeing how larger academic institutions were spending tons of money on resources and people to migrate from one system to another, to another. I figured out that the key as a software developer is not getting more involved, it was actually removing certain aspects from the software that are part of our power dynamics -- parts that were making decisions for people about their practices. So Archipelago is born from the idea that the only way we can give people freedom in cataloging and classification is if we remove the hard coded decisions that software developers were making in the system.

These decisions are included in the systems because, first, software developers always think they know better. I'm also a developer so I have felt like that many times. Secondly, because we are not on the caring side. We don't care about the postcard, we don't care about that syllabus, we don't care, actually, about that family archive. We are just acting on other people's care. The people who truly care are the catalogers, and the managers of the cultural heritage organizations who deal with the people who donate things, and experience the emotional trauma of cataloging racism. It is possible, I think, to extract yourself and say, 'I need to listen, and I need to build tools that allow other people who are experts, who are deeply involved in those realities, to act.' And that's where I get the idea of empathy and empathetic digital repositories.

Basically, Archipelago's a lot of tooling. A lot of things that you can reuse to bring your analog, previous experience-based workflow into something that makes sense for your work. Then you have our support, not to impose a specific MODS version, for example. But our support to allow you to decide if MODS is something that matches your values and needs. If you want to use IIIF version three or two or five or six in the future you have the tools. So, this empathic feeling comes from building the tools, but also doing a certain level of our own knowledge and practice transfer.

It doesn't work to say, 'Here are the tools, just take them.' So the empathetic part requires actually passing experience, passing knowledge, giving tools that make sense for questioning, and questioning over and over, what you're doing. It's not to focus on balancing the community of, for example, developers and users such that you'd have 50% of both. But rather acknowledging that software developers are just a tiny tool in achieving something that is not software-based. You're not putting things into a repository because you believe repository software is the best. You're actually putting

them in because you want to share these valuable assets and stories with other people. It's a narrative, so we need to tell the story together. That's where everything is born. There are so many technical details how we came to this, but that's the driving factor.

Ziegler: That's wonderful. Thank you so much. And yes, you're right there's a million technical things to say. But for our purposes, it's the thinking behind it that I was hoping to focus on. So I really appreciate that. I'm going to circle back to this one thing you said. It sounds to me that what you're saying is that you want to foreground the experience and needs of the people actually uploading content to digital libraries. That makes it sound like from your perspective, other digital libraries, software solutions, aren't doing that. I wonder if you're willing to speculate a little bit more on why that is.

I know you said a little bit about capitalistic impulses and maybe even entrenched fears of deviating from prior patterns. But could you say a little bit more, because it almost sounds so simple. You know what I mean, Diego? It's like, 'Yes, we should be building this for the people who are working.' But I think I'm in agreement with you that it doesn't look like anyone else is doing this. Can you say a little bit about that?

Pino Navarro: Yeah. So I think the most important part of what I'm trying to say is that even when you try to do what we're trying to do, it feels like you're still imposing something on people. Why do I say that all our systems aren't doing this? Because most of the systems that are built for libraries and museums are driven by specs. So, specs like MODS, IIIF, Dublin Core, Darwin Core are made at a specific place, in a specific context that current catalogers don't share. Specs are created by people talking for years about semantics, and those people likely get involved in actual use cases and they interview practitioners and get feedback.

But time is not a constant. From when you start building a spec, finish a spec, and publish a spec, a lot of things have happened that actually drive cataloging practices. Then you take the spec, the end result that can be understood by a software developer, and you implement it exactly the way it is written. Now you have a cataloging system that is MODS-driven. Okay, MODS can accommodate a lot of things, but MODS is like a tiny abstraction of the universe, agreed on by a group of people that are, of course, very smart and very deep into the field. But, they have a different perspective, different lived experience than the lonely cataloger in the basement dealing with postcards, old photographs, manuscripts, and other things.

Again, this view of cataloging and spec-making can be debated and I might be wrong. But it still seems to me that we tend to leave out one main character in the process of development, which is the actual cataloger. Even if software developers and schema creators do a lot of interviews, it's never possible to do enough. What I'm saying is that a system that can generate MODS: that's good. A system that can generate Dublin core or IIIF: that's good. But the act of doing your work through the spec is the problem.

If we think about the way the human brain works, even with all our complexities and feelings we envision the world normally in very simple ways. But we are still very complex. We cannot escape the fact that we live in dimensions like space, time, and also our own mental dimensions. One of the things I noticed while I was actually doing migration from one system to another, other than that the intent of the catalog was getting lost (for example, provenance and any vestige of past practice), was also that every spec is trying to reinterpret, put different formats to, the same buckets of understanding. Everything that we try to describe, always fits into this idea of the 'what,' the 'when,' the 'where,' the 'who,' and there are like subtleties in between. But in the end, MODS is still speaking about the people involved and like the subjects are basically the way what we envisioned at some point in the past, right? And that 'what' can have many ways of being expressed if it could be un-jailed from a spec.

What if we used an analogy, one that comes from our personal life, but which is extremely social: cooking. Let's assume you're an inexperienced cook. When you cook, you're using a cookbook. You have a certain amount of things in your fridge. You want to cook this fancy meal from the cookbook, but you don't have all the ingredients. You can either give up and say, I'm going to go to the supermarket and buy all the different things and I'm going to follow this recipe perfectly. And then, what you were going to get, if you do that exactly, always the same, you're actually building MODS. I'm using MODS because MODS is very North American and it's not a bad schema at all.

But what happens if you actually don't have the money or the resources to get more ingredients? Well, you have to cope with what you have, right? You still have to cook, you still have to come up with something. Then probably the next time, through that experience of cooking with ingredients you had, by improvising, you will probably make it a bit better every time you try. Still, not like your original goal, not that special, beautiful photograph of this perfect dish from the cookbook. But you're working with what you have and you're getting better all the time. So, let's flip. Now you have a fridge full of things. Let's say you're trying to make chili. There will be a lot of different ingredients that you won't need. These are still valuable ingredients, you just don't need them for your chili. But of course, they can be used for other things.

So, metadata is basically the same. It's like you have ingredients all over the place, very close to you, locally sourced, right? Things that only exist in your institution. And there are things that are common for everyone, salt and all that stuff. You cannot dismiss the ones that are local because your final dish doesn't have space for that. You're not going to throw away your very specific Louisiana beignet, right? Or your pickles, because your chili recipe doesn't call for it, right? But you can also start experimenting with new ingredients in recipes, or even whole new recipes. That can happen if we, digital library developers, give people the tools, the support. We can give access to more cookbooks and also ways of understanding how the cooking works so they can actually invent their own recipes, right? You give them that, you will actually see how that becomes like a social impact thing. Your metadata will permeate into your communities

because maybe you're actually adding the sentiment of the cataloger. And that's important for the community that you're trying to reach out to.

So, I'm saying this from a very personal perspective, because I know there are large teams that need to be generalists. I'm thinking of places like LYRASIS that are built on the idea of tools for many different types of institutions. These systems need to cover all the bases. The trend is to generalize everything, to aggregate all data. And by doing that, you're letting all the differences, all the identity out. That identity, right now, feels lost. Or you have it now, maybe kept in a place outside of your digital repository, but will you still have it in 10 years? So, I'm interested in the opposite. I'm thinking of this recipe/ingredients idea deeply. I want you to have full control and access and knowledge of your collections, and to know how you built your own input methods.

You're going to have web forms that allow you to put linked data or manually describe content. Everything goes into this big bucket of JSON. Then you have the recipes and the recipes are actually ways that you can get certain elements of your metadata and put them into MODS, for example, in real time. So then you have MODS and you're probably fine with that most of the time. But then, of course, a new version of MODS will come out. Why don't you just change your recipe? Luckily, your ingredients are already there. The system doesn't need to be upgraded, it doesn't need to migrate, no need to hire a new consultant to move your data from version A to B. So the goal here is to make systems that are flexible enough to meet lots of needs. They do require teaching and learning. We cannot encourage others to use the power they already have without knowledge. Just like in politics or social justice or anything, right? We always need to start by learning and gaining knowledge, but then you grow with it.

Again, Archipelago has two good things. The first is the people, the people writing the code and the people using it. The second is the non-profit that enables us to make this bet on the future of digital library development. Because we have a goal at METRO, we need to serve our community, right? So we can fund Archipelago without making money for a long time in a sustainable way. I think that's the reason we can do this. I think that's the reason we have accomplished what we have accomplished, so far.

Ziegler: I really liked that you're using the example of cooking, and I particularly like that you're acknowledging that it's an example that comes from your own life. It's an example that you're using because it's an example of a type of thing that you do. And that's maybe a good segue to one of the broad topics that we try to cover in these interviews: why are you the one doing the work? You mentioned before that as a Brown man with an accent, there's a limit to the amount of pushing you can do in different environments. And I just wonder, and this is going to be a purposely broad question, because I'm very interested in your answer to it: Why do you think you're driven to build such a system? Why are you driven to think about this so empathetically, where, perhaps it hasn't been foregrounded by others?

Pino Navarro: Okay, that's a super good question. I might go deeper or less deeper, depending on how this evolves [laughs]. I don't think I'm actually the only person that has attempted this or something similar in different contexts. Actually, like the people behind the idea and implementation of the semantic web, all the people working on ontologies, we're all basically trying to do this. It's like a basic function of the human brain, of human knowledge, the way we rationalize things into systems, so that we're as abstract as possible. I will say I was really privileged and lucky to be pushed into corners where I had to find my way to be useful for other people. Let me explain that. I was like, again, very deep into these open source communities. Suddenly, I felt a bit like an outsider, right? Bringing in ideas from people that are not in the conversation.

I was trying to advocate for the people METRO serves because that's the people we have close to us in New York. I didn't have here in my tiny little universe access to those deep thinking, MIT, Stanford scholars. I had smaller institutions dealing with very simple things like their dates could not be formatted the way they need it, right? They couldn't actually add things because the system told them, this is not the way. For example, they might be told by the digital repository, 'Books can only be this,' well a book has many shapes, right? It can be like almost anything, right? By being in this place, in this time, I needed to actually find a purpose. What do I do with my new situation? I cannot trick people into using a system that doesn't work for them, it would not be fair. On the other side, how can I actually find a solution?

And then, another personal aspect: I started working with software development when I was 12 years old, or something like that. I had my first company when I was 18 years old, and I was totally consumed by capitalism. I thought, 'Okay, I'm going to make it. I'm going to make money because my family never had money. I'm going to get a Porsche and stuff.' That didn't last many months because there's something that you cannot ignore, when you are aware of your origins and you're still connected with the people you live with.

You can either trick yourself into forgetting about your past and reinventing yourself in this new image, or you can accept that you belong to this other world, right? This other world where character leads invention. The character leads to discovering new ways of doing things. Still in the larger IT world, I changed professions many times. I used to be a filmmaker, I used to work with movies and post-production and documentaries. And then I was working with archeology and geology, I worked with cultural heritage. My mother is an archeologist and I was always immersed in museums. Getting older, I'm led to a deeper awareness, and this awareness exploded when I came to the US, that I'm part first nation.

This is true of many many people in South America, but in my case I could actually track it back; I know who my people are. I know who they had to fight against, how they were oppressed by Europeans, how they managed in terms of cultural loss. Engaging with that is how I started to build care. It's like, you understand yourself and you start caring about things, right? Not technology, not money. I was also very poor and I'm very proud of that because it was a decision. I didn't want to work at a Telecom

Company or a bank. I wanted to keep doing these different things. In each profession, each new job, I always asked myself, 'Okay, what is my role as a software engineer here?'

And my role is to let other people tell the story. It's not my story. It has a caring factor. I go deep with you, learning about reefs and biodiversity on the water. Or I go deep with you, learning about archeology and first nations, my own people. But it's your thing, right? It's your field, how can I assist you? So basically, how can I put my part of the brain to good use. I think that's what led to this. The same with cooking. It's like cooking can be seen as a very selfish act of like, 'I want to show you the best photographs of my food. So you can see how cool I am.' Or can be a social act, when you make your food for a lot of people that don't have access to food.

I think of cooking in a way that goes back to archeology, back to cultural heritage. I'm thinking of it as one of the first acts of love. It's not only just eating what you found, while walking in the fields. You're actually bringing things together from different sources, acknowledging different ways of working: the hunters, the gatherers. Then, you bring that with love to other people. It's social, it has a direct impact on others and you can actually get better at doing it. So I think that's the source of everything I'm doing now. I'm also a very flawed person and very stubborn. So, probably that's also a driving factor.

Ziegler: In my experience, stubbornness always has so much to do with the story. Now, I'm thinking a little bit wider. Software suites come and go. I think all of us who work in technology in any way, especially those of us who ever touch digital repositories, know that. Thinking beyond Archipelago, how would you like to see the field change? What would you like from digital librarianship, maybe writ-large, or even developer work within digital libraries? What would you like for that to look like in 10 years or 20 years?

Pino Navarro: I think probably what I would love to see is that we stop thinking about globalization as a solution. I think globalization, normalization, huge discovery portals where everything is normalized and rolled together is an excellent way of obscuring smaller realities, important realities and identity. Really what makes us interesting to other humans is our differences, not our commonalities many times. Actually being in charge, owning our differences, it's extremely important. The driving economic system is trying to make us look all the same. It's trying to bring everything together because it's easier to parse, it's easy to process and it's easy to sell us a blue jean bag because we all look the same. What I would love is smaller endeavors to understand localities better. A bit more self-esteem and love for existing practices. Understanding that while we're reshaping our environment, older ways that were less constrained or less defined by economic needs are extremely still valid. And should be kept.

Also, what I would love to see is more flexible ways of cataloging and labeling. The moment you actually catalog is important. When I decided I'm going to use this URL in this label, that's the moment of awareness. What happens afterward if the label evolves? If the linked data I reference changes? Well, that's actually what humans do,

right? We evolve terminology. We make things change. So maybe this is where we add a bit more of Digital Humanities, Anthropology, Sociology into our acts of cataloging and catalog software. Acknowledging that the cataloger is shaping data, the same as a journalist is telling a story, right? Everything is a narrative and it's a personal narrative. Schemas sometimes hide that, they make everything feel very consistent and precise.

But by focusing on the communities that are being positively affected by your cataloging, I think we can from that point start thinking about, 'Okay, how do we make this larger for others to see?' But, I think this making-larger for others to see, it's not necessarily about sharing the data. It's sharing the practices on how the data is built. I would love to see a community, basically in 10 years, where knowledge, experience of doing things flows much easier than just data, schemas, and metadata. I think that's something that's missing sometimes in our work. Why are you preserving this? What's the impact on the people who care about this? That's my hope and maybe I'm just dreaming of things that will never happen. But yeah, I don't know.

Ziegler: Well, that's everything that I was really hoping to cover. I just want to pause now, Diego and ask if there's anything that you would like to say and touch on?

Pino Navarro: I think, probably the last thing I want to express is that people talk a lot about building community, right? I don't think you can build a community. I'm speaking about communities of catalogers, archivists, or software developers. What you can hopefully do is believe in something and just keep working on that, but also question your beliefs constantly. One of the things I noticed when you do that, there are often other people locally, not globally, that find something in common with those values. When you do that constantly, community starts to get stronger, not larger, just stronger. We become more empathetic to other people's things.

I feel that's where we, as open-source developers, have failed badly. Governance is a thing that drives me really crazy, it's like pay to play. You have to pay to be able to make decisions on software roadmaps. Then it's always the same players, right? It's the same as an economy. It's like, it's only the millionaires that have a say. You can go and vote, but in the end very little changes. So, open source communities need to be closer to real communities where your effort in your daily work becomes relevant to other people. I think a beautiful example, again, going back to my personal interest, is how community gardens work.

I think community gardens are an example again, coming from our origins as human beings: sharing space, caring about other people's things, acknowledging that someone can leave and come back. There's this common work to get something that helps not only the people in the community garden, but also expose flowers to people passing by. Community gardens lead to food and to cooking. So I think that's a thing that I'm thinking about a lot. It's how we, as Archipelago, can be part of nourishing a future community without going into those many issues that open source communities and

practice-based communities have gone into in the recent past. So, that's all I have to say. Sorry, I speak so much.

Ziegler: No, this is an interview you're supposed to speak so much. Diego, thank you so much for this.

Pino Navarro: No, thank you. It's good to be able to talk about things that are just not JSON and IIIF.