QE INTERNATIONAL SOCIETY FOR QUANTITATIVE ETHNOGRAPHY
QEsoc.org
WEBINARS

http://qesoc.org/webinar-series

Note this was automatically transcribed by https://otter.ai so there are errors, and the timestamps do not match the video.

August 2, 3PM (CST), 10PM (CET), 6AM+1day (AEST)

QE Fireside Chat: A Conversation about Designing Tools for QE

Cody Marquart
University of Wisconsin-Madison

César Hinojosa
University of Wisconsin-Madison

David Williamson Shaffer
University of Wisconsin-Madison

Abstract

A fireside chat with Cody Marquart and Cesar Hinojosa about designing tools for QE. This is an opportunity to talk with two of the lead designers of ENA and nCoder, and see how they integrate technical and aesthetic perspectives with the mathematical and theoretical foundations of QE. In the session, Cody and Cesar will talk with David Williamson Shaffer about their design processes, the trials and tribulations of building tools for large audiences, and the need to develop a more community-based participatory design process going forward --- with plenty of time, as usual, for questions and discussion with the audience.

Brendan Eagan 00:15
All right, David, do you want to go ahead and start our fireside chat? Is that an Americanism do people get I mean that everybody has firesides I guess, but people kind of get the the, the trope of a fireside chat. Imagine everybody that it's maybe cold or you want to at least be by, maybe you have a nice drink or something or something to snack on. We're having an informal discussion. I'm relaxed more already. We're in like, maybe very comfortable chairs or something like that. Some whiskey.

David Williamson Shaffer 00:44
Yeah, um, Andrew, it goes back to FDR, right.

Andrew Ruis 00:49
Yeah, he's I don't know if it had earlier usage. But But yeah, FDR, initiated a policy of doing fireside chats, these national radio broadcasts where he would essentially talk directly to the American people about things that were going on, especially during the Depression. And then that was something that was widely lauded as politically genius, because it sort of brought it sort of brought the government into the living room of, you know, everyday Americans and created that sense that, you know, even if it seemed like your world was falling apart, somebody somebody was, was out there working to fix it. And so I'd had that sort of idea of sort of comfort or, you know, support Bill 10, but also, the access, right, like, you know, even as an everyday American, you see, you had access to the inner workings of government and what they were thinking and that kind of thing. So it's it's a nice model for this sort of thing, where we are all getting access to Cody SSRS, as they think through these big and gnarly problems, and then how to solve them.

David Williamson Shaffer 01:53
Yeah, I was pretty sure that's where it came from. And, you know, before we start, I just like to say, how important Cody and CSR have been, not just to the work that we do in the lab, but really to the community. You know, there, it's one thing to have lots of good and interesting ideas and to think about how to solve, you know, analytical problems, it's quite another to reduce those to practice in a way that people can engage with the ideas without necessarily having to, you know, follow all the mathematics or learn all the D, you know, learn all the inside details of how to use our statistical, statistical computing or those kinds of things. So I just like to start by, by really appreciating the two of you, not just for, you know, the, the pleasure I've had in interacting with you, but also for the role that you've played in making the community what it is. So thanks very much. So having said that, I mean, he just said that, like, there's this pivotal role that you guys play in helping the community learn to use these technologies and through the technologies, the ideas, right. I'm wondering, like, when you're sitting down and thinking about, you know, building something that any tool, the encoder to RNA packages, whatever it is, what do you guys see as your biggest design constraints? Like? What's the thing that that is kind of makes the problem the most challenging?

Cody Marquart 03:38
Do you wanna go first? Yes or no? You're? The, I mean, I think there's a couple things. I mean, one is typically, always time. And time is mostly a constraint. Because there's lots of things we want to do. And trying to accomplish all of those within, I mean, typically, a lot of the things we're working on have a defined constraint by some sort of funding. And so we're trying to get moving quickly, and trying to get to a prototype or something that we can actually see functioning rather quickly so we can know where and how we need to pivot. And then I think also, the use cases are, it's hard to, we try to envision as many different use cases that people will have that are going to hopefully be using the tools and the things we create. And so trying to accommodate those within set time is always difficult because not only are we trying to make something typically that's never been done before. And it's kind of

groundbreaking in lots of ways. We're trying to do it quickly and so trying to take all those things into consideration is continually difficult to buy, which I think, in injecting CSR into our process want now like five years ago, it made that much easier. But yeah,

Cesar Hinojosa 05:20
yeah. So just kind of going through what you said, I think, you know, leading the design, I mean, leading the way of like, how we design stuff, our tools like ama, usually, when you design something, I tend to go look for, you know, comparison, like other competitors that we have, right, but there's not really a lot of it. So a lot of the times we it's kind of hard to kind of design something from scratch, especially because we are a small team, you know, designer developer. So yeah, I think that's one of the constraints.

David Williamson Shaffer 05:59
Okay, so just, just having just asked you about the hard part, the constraints? What part do you think is the most fun?

Cesar Hinojosa 06:11
I think, to me is a, I like the puzzle, and solving the puzzle of how to make something complex, simple. I think one of my main roles in the design process is having something complex and throwing stuff out or hiding, or, you know, trying to make it to the most simplest form. Luckily, I'm a person that's not too much into the, into the weeds of the math, and then the medics and all that kind of stuff. So one of the one of the things that I do for myself when I design is, can I understand this without knowing, you know, all the all the analytics data goes with it. So for example, ENA, I know, I might not know how, you know, we get to the in the beginning, and I know how to get to the, to the network. But I have to make sure that the network is the focus point, I got to make sure that the network is, is the center point that it looks nice that, you know, it says what you're supposed to be saying. So the puzzle is pretty fun.

Cody Marquart 07:16
I think I agree, I think there's a puzzle that comes with making new things. And for, for me, and a lot of ways, it's, we ended up getting to us finding new ways to implement existing technology in ways that we haven't, or people haven't necessarily done before, at least in ways that we haven't done. I think that's, that's, that's always fun, internal, kind of baked into our processes, pushing the boundaries in every way, whether it's making tools and new, you know, processes for people to, you know, do their analysis with. But that also requires that, you know, we find new ways to solve those problems as well. And it's not always easy, but there's not always an existing technical solution either. That fits what we need to do. And sometimes there is, but then still, sometimes there's the door open to still play around and find new ways to solve the problem technically, as well, which, which is fun. And then, I think also, just seeing, especially as this community has grown, seeing people actually use the tools is always rewarding. Especially from our I mean, I think I'll speak for CSR a little bit here. But like, we're we're at, you know, normally kind of behind closed doors often, especially now, in the last year and a half. The, like, you don't always see that, right? Like we're not like, I mean, obviously, with the ICT week coming up, like, you know, it was nice when we had the in person when you could walk around like poster sessions and, you know, see na graphs, you know, screenshots from tools that we sat around for years and tried to figure out best ways to make a make a system that people could use. And seeing that was, is always rewarding, I think.

David Williamson Shaffer 09:03
Yeah, that's my favorite part to actually, although what I really like is every once in a while, like somebody, either somebody new or somebody will come along something and you can literally just see that their brain kind of goes. And those moments are really wonderful

because you're just sort of seeing somebody shift the way they're thinking about them. So we're talking about like, what's hard and what's fun. What or I don't know, one or two or three things that you guys feel like you're the most proud of that you've that we've designed and the time you've been working and I don't mean like you could say ENA but I'm thinking about like, oh, there we put this thing in or we did you know this specific thing and it just really, really clicked

Cody Marquart 09:53
Well, I can't say no. I mean, I think The early on in the ENA, probably when we were so years ago, I mean, this was like a decade ago, ENA was an Excel program for the most part. And then we had a first version of it that, you know, we read, we wrote in Java while also using our to do simultaneous, you know, una models and, you know, being, you know, assays are mentioned, and we're very, we're in Madison, we still, you know, a small group. And so, the idea of maintaining two separate things, you know, someone wanted to do, you know, you know, very analytical work, like in our they had their, our package or, and we had a, you know, a web version, early on, we'd made a decision, like we had realized that, you know, that wasn't sustainable. And so we we decided to make a web version of our tools. And this goes for encoder as well, where that actually relied on the our package itself, where it limited the amount of things we had to maintain. And to see, to see that decision actually worked out in a way that, and this kind of ties back to like, in a way, the way that any web tool works isn't, there wasn't a lot of precedent for the way we implemented that. And the way we interface with the our package specifically. And to see not only see that work, but then Now, that doesn't mean it's been sustained for now, like, three or four years, in a way that people have been able to use that for the most part reliably. Everyone's had their fair share of run ins with some bugs throughout it as we grow, and more people bring different data in, and we get more eyes on tool, but for the most part that that decision worked out really well. And it saved us time, but then also allowed the community I think, to grow faster. Well,

Cesar Hinojosa 12:04
you know, it was also my proudest thing, you know, just because I, I didn't know much about it, right. So being able to see, like, the reception that I get from, like, the gaming, like people saying, well, I like you know, like how it looks. And just the different obstacles and like the complexity that he had, right. So whether it was the network's whether it has like, you know, the model tab, how much information that's in there, right, and how distill, we had to make it to make it work, and make it so that it's intuitive enough. So that that I'm proud of, but just to go a little bit a little bit away from that, from our tools. Another thing that I'm proud of is the the illustrations that we have little sticker illustrations. I don't know if you guys seen them, like the roar monster, and like the spider and stuff like that, I mean, I think those are cool. I like the fact that it brings this kind of joyous, joyous kind of feeling to the to the tools because not not everything has to be like, you know, super serious. So. And I like the fact that a lot of people like it, and I like seeing the stickers on people's laptops. And I like that we created those, I think, I mean, there's even like some type of like clan, like people have to decide which one they want to take. I like the fact that people like those

David Williamson Shaffer 13:29
as a clan affiliation, I mean, I do like the stickers. And I'm always surprised at how much other people like lapped them up. But I would say, you know, one of the things that's always struck me about Ian and I think has been an important part of its success is that the, the, the diagrams themselves have a certain aesthetic to them. Sometimes that's problematic, right? Because, as you know, Mike and other people who go and her team, right? who work with people who are not researchers, right? It's not exactly end user friendly. But but they just they're they're aesthetically pleasing when you think about network diagrams, or at least they are to me, there's actually a paper at ICP. That's going to be that's about why those

network diagrams are constructed the way they do and like how that how that conveys information. But I think that was a really that's been really, really embed the look and feel has been really important.

Cody Marquart 14:31
Yeah, it was a very, it was a fun time, because we had two undergraduates, basically like head to head. They're both experimenting with different variations of what types of things we could do with the networks and which were like wind things actually played well together. And I think ultimately ended up being a combination of both of their efforts. That that has made it into the tools but it was it was a fun little work experiment there.

David Williamson Shaffer 15:00
Yeah, it strikes me that, you know, there's a lot of things that are that go on kind of under the hood, meaning not even in the tool itself, but in the actual creation of the tool that we probably could write a paper about. But we, you know, we tend not to because we're not necessarily focused on like the, you know, the the building of the tool itself and Cody and say, sir, I know, you guys aren't particularly interested in becoming academic writers. But I wonder if maybe you could, like, give us an example of the sort of the design processes and how, like, how that I how that magic unfolds? Me, maybe they pick a specific example, or, you know, pick something that the sort of a generic description of the process just so people get a sense as to what's actually going on, when we when we sit down and say, Okay, now we're going to build this.

Cody Marquart 16:02
Um, I mean, I think the, not surprisingly, everything is it has evolved. And it's, I mean, even when, you know, we were presented with the idea of doing this webinar, it was interesting, because it never it, it does continually change as we take on new projects, and you start to think about what we're going to do next. And so it's never reflecting back over the years of the things we've done, and never really thought of there being a strict process or technique or whatever that we typically use. I mean, so I mean, it definitely does change. And we take into account things that we've done right or wrong in the past. But I mean, I think one thing that is consistent is, in some form, we like to prototype things out and kind of see how things are going to work, whether it's undergraduates that are messing around with visualizations that we can see and how to test things out and see how that how that'll work. Or, you know, I have, I mentioned before, precise, our pre having a designer, in our, in our group, it involved a lot of actual functioning, to some extent I use functioning loosely, that I would make an order for us to kind of see how things would, when we have ideas, you know, its ideas are really, really sometimes easier to come up with than it is to actually make them into something that functions and then the costs of having to technically create those prototypes to then only be like, Oh, that's actually not not going to work was high. And so bringing in CSR had completely changed our the way in which our process actually worked, because then we could, we could iterate on visual designs, as opposed to stop somewhat functioning prototypes, which I think allowed us to come to not only get verb prototypes out faster things that we could actually, you know, release to some some, some group of people quicker. But I think as a lot of what, you know, the success that we see in ENA came from one of the first iterations of that process that involves ASR. And that, you know, we were able to see how things were gonna come together and how things were going to flow without having to put a lot of that technical debt in early. And so we could kind of iterate visually and then ultimately make changes. And then the changes we made technically, after that were smaller, which I also think gave us a better idea of how, what the tool needed to do once we got there, which I think has played a lot into the success of ENA in that we knew more of what we wanted to build when we actually started to build it.

Cesar Hinojosa 18:50

Yeah, and the process starts when Cody just tells me to go into his office, and tells me what the next project is. So for example, encoder, right? I go in, we look at stuff that, you know, we previously have built, we identify what the problems are. And then we start mocking stuff up for nCoder in this case. And we not only tried to solve what the problems are, then we also tried to have iterations and go part by part. And sometimes after many iterations, after many meetings of, you know, us just kind of arguing or just a little bit, yeah, that works. We started going for the next part. I remember that encoder. In the beginning, it was like an utterance, right? Every single time you say yes or no. So, an example of this would be you know, we decided that instead of having one others we'd have like the list of multiple 10. And, and then we decided that that would be a better, better way to display the others. And have them all, like not even selected because there was also a time when they were all saying no. So he like, forced the user to actually read it and make a McCain election. So those are the those are like the the type of things that we do we like to go into specific thing, see the words, if it doesn't work? When do we try again, we all come with, you know, ideas and solutions. And I think that's what makes it, that's what makes it work, I think I can, you know, I can say that you guys, David, you can help me like you helped me like, think of stuff that I don't think about. And so then I take that into consideration and try to like, iterate into something better. But a lot of people are looking to me.

David Williamson Shaffer 20:46
I mean, one thing that I've learned, again, it's you know, Cody said, we've sort of gone through many iterations of the design, prop design processes, and now we've been working with Cesar for a number of years that, you know, there's a, there's a kind of, well, not very flattering picture that people sometimes paint on of people who do graphic design, which is that they're sort of putting lipstick on a pig, right? But the thing is, they're and then they're just putting in the, you know, a nice looking skin on that. And, and what I've learned, actually, is that the process goes exactly the other way. Like, first we design the lipstick, and then we build the pig behind it. And I guess you're still getting lips to kind of pig but you're actually it's actually doing what you wanted to do, because you've thought through, not just this, how it looks, but actually the functionality of it. So I really appreciated learning that from Well, from you, in particular system, but from the whole group kind of working together. So you've raised the encoder, let's say, sir, um, and, you know, I, I think there's sort of a sense, I guess I haven't since met Cody, maybe I've said this before, too, but that it's been a little bit the poor stepchild of na. And I'm wondering if you guys, just from that, sort of, from your perspective, is that, is that really the case? And if so, why? Like, what, what are these two two things, and they're both really important, but it seems like, you know, one of them gets all the one of them's the golden child, and then the other one sort of languishes a little bit. Um, it was my favorite. I love that I love them.

Cody Marquart 22:36
I think in a lot of ways, it's unfortunate. I mean, it is true in that I think encoder was always a necessary component in our group's analysis, but it was it hadn't grown to the point that it was necessary and other people's analysis. And I think that showed in that room, we had funding as well to work on ENA, but not encoder. And so what we needed to bring encoder along for the party, because we required it. And so in a lot of ways, we didn't actually go through similar processes for encoder that we had with ENA, like we didn't start from the beginning, we had any. Again, I think it was also some variation of an Excel program that was and that was the original encoder, that then we made into an our GUI of some sort. And then, one week, I all of a sudden, you were at David, were starting a class and you knew you were going to use it in your class like five years ago. And so then I spent like a week or two, just throwing together became the first web version of encoder. And I think you survived with the class. And I don't think he had, I don't, I don't think I don't think they all dropped out in mass. So that was a success in itself, I think so. But then, I think the Netflix version that is the current encoder web tool, in a lot of ways is mostly lipstick. And we had a, we had a

version that kind of worked. Cesar provided a lot of design around the workflows to help people understand what the process was asking of them in terms of like, wow, like, where the first Raider and the computer were interacting, and at what point they were supposed to, like, you know, involve a, you know, a second human rater. But we didn't, we didn't, we didn't spend as much time thinking about the underlying components that make that would make the tool successful in the same ways that ENA had. It is still built in a lot of ways on an AR package like na is but even the AR package for encoder has a handful of issues that make it much less stable, and we just hadn't dedicated the same amount of time to it. I think that's where I think that's, that's where that shows I think, well isn't,

David Williamson Shaffer 25:10
I mean, I don't want to jump on, on top of say, sir, but I mean, I isn't part of it that like we're in this environment where we're always struggling to figure out, like, do we one way either when we encounter a problem, or when we there's some new necessary feature, we saw, there's sort of this this tug between wanting to, you know, fix it now, and, or put something in now, versus recognizing that, really, the whole thing would, you know, should be redesigned? And so we're, you're constantly like, either waiting to put something on the, the bigger, better version, it's coming, or you're, you know, kind of using scotch tape and a little bit of glue and some chewing gum. The whole thing together, we don't even use duct tape. We are using scotch tape. Yeah. Well, that's part of the problem. Wonder it's barely held together.

Cody Marquart 26:04
Yeah, I mean, I think I think that's true. And in any of the things that we're in, we do, I mean, specifically, I mean, an encoder, especially in that whenever there's, you know, a request or an issue or, you know, a feature bug or whatever that, you know, we have to wait, I mean, typically, you know, it's we have other like, there is a time constraint, and we talked about earlier that, you know, we have other things that we're doing too, and you know, it's like now it's so you don't have to wait, we don't do a ton of active development on either end. So if something comes in, we have to wait, when or how we make the change. And I think encoders specifically, I think we've been hoping for a few years to finally dedicate some time to thinking through all that process works, that will be more, that isn't just something that solves our problem that does solve a greater problem for the field. And so I think we have kicked that can down the down the road for a long time. And so I think a lot of times, there were things that, like reworking the underlying are packages that people could use, and that fuels the web packet that with a web tool, I think taking the time to have done that would make a lot of things more, work more smoothly. And what happens is a lot of things, and we kind of put those band aids and the sticky tape on on the web tool that just to kind of get us by. And I think that's it's always tough, because I mean, once I go back to it, and I start looking at it again. I mean, Cesaro probably does it same, the same visually, like, when you see something you did before, it's like you really feel like get you back in it. And then you want to, you want to dig in and spend the time to make it better. But sometimes it's it's hard not to jump in those in those rabbit holes. Yeah, I mean, before we dumped too hard and encoder, there are some things in there that I think are quite brilliant. visualization of the three way agreement, for example, I

David Williamson Shaffer 28:03
think is really, really, really, really well done. I there's one more thing I want to ask about, but say sorry, I just want to give you if you want to comment on the sort of poor stepchild question.

Cesar Hinojosa 28:19
I don't have anything else to add, I think I think Cody says it all, like, set it all about like how we have to like make pirates prioritize like our time and like, seeing when we can do like big

pushes on some of the features. I know I know that sometimes we have to make decisions on what needs to be cutting what sometimes doesn't, sometimes stays. And I know for me, I have we have some features or some like functionalities that sometimes get stuck in the box. And they're just there for like maybe in the future, we decided to bring that back. But yeah, I also love a voter. So

Cody Marquart 29:05
Oh, features that get stuck in the mocs translates to things that I say, Oh, yeah, I don't think we have time to do that. Say sorry, we're gonna have to pull that one.

David Williamson Shaffer 29:16
So that leads me to sort of my final question that I really would like to throw the floor open. If people have things that they want to ask what I mean, I can talk, we could talk about this all day, just among ourselves. But, um, so one of the things that we you just raised this is sort of the question of building things for a community rather than building them primarily for our own use. And, you know, features that get included or don't get included. I think there's also raised the question about, you know, bugs that do or don't get addressed and in what way and how quickly. I mean, it feels to me like we've moved at just as the three of us even right move From a world where we were designing, and most of the clients were sitting in the same room, it was mostly for us. Right? And then we evolved to a place where now the designs that we're making are for people that we might not have, we might not see, or it's for people that we don't even know yet, because they haven't joined the community. Do you guys? Do you guys have some thoughts about like, how we start to move to a more kind of participatory design process with a community rather than just, you know, trying to make stuff that we think is one? Which, which, you know, there's something to be said for that too, but

Cesar Hinojosa 30:37
yeah, sure. So Brandon, and I have been starting to do a lot of focus groups. When it comes to like, I think, like we said, before, the process is evolving. And I think one of the big pushes that we're doing right now is trying to get more involvement from the community to get more feedback from the beginning stages of the process, right, so that we can start creating like user cases, we can start creating journey maps, we started to do focus groups. And I think all of that just makes it so that the design process goes faster. And it's it, it's better for everybody in the long run. And sometimes just kind of like the meeting thing that I said before they bring up like, things that I would never think about. Recently, we Yeah, so it's, it's just a, it's, I think it's a it's a process that can like help us in the long run, when building any of the tools in the future, or, or even like, I know, coding, you get like emails, right? Whenever there's a bug or something. So even even that type of feedback from the community to help us move forward some of the tools that we had.

Cody Marquart 31:51
Yeah, I think, early on, especially when before there was any, there wasn't really a user base of the tools we had, we would try to create, or think through all the use cases as much as we could, and try to create something extremely robust, such that like, you know, we wanted to make sure whatever we were putting out there, you know, was representative of what we how we felt we wanted to present ourselves, to the community. But early on. And I think the I mean, even bashing an encoder a little bit, just because I joke, because I look forward to actually redeveloping it sometime. But I think, in a lot of ways being like, the tools we've created are such that they're there, they're extremely beneficial in their current state. And but then letting them out there before like, you know, even though there's problems, there may be some use cases that we haven't thought of, or there may be some bugs that we haven't caught. Allowing the community to use them really helps us figure out more ways in which we can change your grill, the grill, the tools are working on it. And because I mean, I mean,

we are a small team. And so we would be would exhaust all of our energy. And trying to catch all of this stuff early, you're trying to design for every single use case. And so including people, either early on in focus groups trying to figure out what kinds of things they would they could benefit from in one of these tools, or if it's as they're using them, making sure we're reaching back out whether it's through automated bug reports with a bright orange button in ENA that, you know, sends me an email, or just, you know, kind of pinging people at like, you know, hey out, how are you using the tool, you have any feedback, that kind of thing? We benefit from that in a lot of ways. And I think there are changes, even in the current tools that have been that are that show that some?

Cesar Hinojosa 33:55
Yeah, one thing I would add to that is that having people from the community kind of helped us build these tools. It also makes it so that that worship is not just on us, right? So ownership is more for the community. And the more the more participation we have from the community that also makes us sort of communities more proud of the tools that they're helping make built or whatever. And I think that's, that's super important.

David Williamson Shaffer 34:24
So,

Cesar Hinojosa 34:25
yeah, I think it just makes it overall a better process.

David Williamson Shaffer 34:29
And I mean, I definitely want to throw the floor open here. But one of the things that, you know, I've been thinking a lot about is the fact that we're going to pretty, we are already at a place where we're not the only ones building tools for the community. I mean, gj and Sylvie are here and they have the flagship of a tool that was built by somebody else, but with the intention that it fits into the sort of landscape of these other tools and QE as a whole And I've been thinking about how we construct some kind of ecosystem such that those things can talk to each other without having to download one tool, do something, take the results of that, make sure it fits with the next tool, upload them into next door. You know, that process seems like it's a lot of science is done that way. But it seems like it's problematic. And so I think for me, that's sort of a next unimportant next step of kind of community participatory design is actually creating a space where people can build things that will talk to each other, talk to each other easily. I don't know, gj and Sylvia, do you want to say anything about creating rock, and you know, what it was like to build into it sort of with it with this other set of tools in mind? And you know, how that process went?

Szilvia Zörgő 35:57
Sure, that I actually had a question that I really wanted to ask, so I'll kind of I can answer you right away, and then maybe TJ can, okay, so, um, in our case, we wish we saw a process and analytical process that we kind of wanted to supplement in the beginning, in data curation and the way that we can prepare data to be used with ENA, but then it also became some something that's larger than that. So it became about qualitative research and how we approach that and, and, and how to make it more transparent and perhaps machine readable and these sort of considerations. Today, was that the question? That pretty much

Gjalt-Jorn Peters 36:50
covers it. I mean, in the beginning, it was mostly just to get stuff into the web in a tool in an easy way. Because we were thinking, oh, there has to be an easier way to have transcripts, and then have a spreadsheet on the other end. But Sylvia exactly summarized, you know, we were starting to think about this. And then we're like, oh, actually, it's nice if you have like, an open source tool that provides transcripts that are human readable and machine

readable, so that people can easily work with it in a simple way. So that's kind of where it grew from. But I am, we are trying to really try to connect to RNA. But it's exactly what Cody said, it's like, time. But if there's time, it would be really cool. If basically, you could just start your analysis and then directly interface with RNA at the end, or have one online tool that just helps you code stuff, analyze stuff, combine everything, and then give you both a machine readable script file that has all the steps that you took. And the results, that would be awesome. Like, in 10 years or so.

David Williamson Shaffer 37:57
Hopefully, I 10 year. So you said you had a question?

Szilvia Zörgő 38:00
Yeah, it's Oh, I haven't so ready, that it was actually in my text box. I was interested in what how would you define what a QE tool should encompass? What what what kinds of specific things are unique to QE and not necessarily to na, because we, you know, tend to conflate it a lot. So, yeah, what goes into creating a QE tool?

Cody Marquart 38:36
Good question. I've only been creating QE tools now for a while and so it's hard to remember what goes into other creations. Um, I mean, I think in a lot of ways it is that the area isn't so it seems well defined, but it's not in effect, that the tools themselves in the way that at least we envision them and that they can be used by a totally wide number of like a very large number of fields, I guess. And so, trying to create something that does some specific, something specific but then can be applied more generally, such that it doesn't just, you know, it isn't, you know, in virtual internships, you know, where some of a lot of the work we had originally done kind of originated from, but then it's become more broad, and so trying to, to stay somewhat more general and that way we don't just create something that does something very special one very specific thing. Yeah, I don't know

David Williamson Shaffer 39:44
that I hadn't thought about that piece of it, although that's certainly there. I mean, so I'm very I'm, we are in the process of building a prototype for pipeline, like QE. We don't call it pipeline now. We call it workflow, right? So cute. workflow, or at least I guess I should say, an Ian a workflow, let's be specific. It's not for all of QE, but where you can, you know, sort of assemble the elements of your workflow and see them visually. And then, you know, you're still opening up a tool, but now it's all either sort of one, you know, central location for all that. And we had, we had built that, and we're showing it to some, you know, some users in the community. And at least, at least one person, it was a couple pointed out that like, we hadn't put anything in for closing the interpretive loop, which then led us to thinking, Okay, well, what is it? What would it mean to build into a workflow that closing the interpretive loop part? You know, one suggestion was a checkbox where you just said, Yes, I closed the interpretive brain. And, and it strikes me that, that the, one of the things that we've hit had to do with ENA and within quarterlies, tried to do, and I think you guys did with rock, and a very serious way, was think about the levels of not just transparency, but readability. That has to exist at more points in the process than is happening often with, with analytic tools. It's so tempting when you're, you know, you're building tool, and it's going to do something quick to just, you know, have it do it, and then it's done. And forget about the fact that the whole thing has to not just be open to inspection at the level of the code, but somebody actually has to be able to read the journey that the data's gone through. And so I think that adds an additional kind of design consideration in the QE world, that isn't always, well, frankly, it isn't always required. I think that's a problem, right? I mean, that's, that's how you get really bad. You know, throwing shrimp at walls, and just putting things in black boxes, and, you know, reinforcing all kinds of stereotype bias. And then anybody else want to anybody else have thoughts about that? And Brendan? Yeah, I did. Rick, are both

Brendan Eagan 42:06
Hendrick. Go ahead, if you were gonna say something.

42:09
I don't have a thought on that, to be honest. What constitutes but I know that the same question backs me a bit like, is QA equal to Ian a it is obviously not because we need to want to define this q&a community, but ENA is so dominant, and is a is a cornerstone of this. And I think it's a matter of time to wait for some more tools that manifest what QA is, and maybe rockers one or two candidates there. So but but it's still something that that that also puzzles me, but I put down a similar question. And it's partly already answered, maybe it's back to, to Caesar and so on. So So what is the next low hanging fruit that could be done for the QE community? And we partly touched on this already with that. I don't know workflow where everything is nicely positive two together. But on the other side, I'm getting kind of afraid if it's so nicely connected, and suddenly becomes another SPSS tool for ENA that becomes so complex that it's I don't know, you know, integrated complexity in Greece, so much. So. So what is the Yeah, I was wondering what what would be what is the low hanging fruit you guys have, there may be an improvement of one of the existing tools. And I have one specific question for encode as well. Or, Watson, the next big thing to do in the q&a community would have been on question of mine. Yeah, I

Cody Marquart 43:43
think the two mean, the low hanging fruit at this point, I think, is, in some ways, I mean, I say this now, and I'm gonna regret it cuz it's probably gonna be more work. But then I anticipate but the workflow is David was kind of mentioning that in, in its early version, and the way they were envisioning it, it's likely going to be a workflow to na. But in a lot of ways, the way that we're thinking about it, and what will be the pig behind the makeup is actually going to be the, the technical side will hopefully may be able to make some changes that kind of more kind of generalize the way the tools even our tools, the current ones encoder in a speak to one another the kind of create a more standardized API of sorts that if someone wanted to inject a tool like rock or something else after n quarter instead of ENA, that there there might be something there for them to do. And I think that will open the door hopefully for something more, something other than EA to kind of mean maybe ENA stays the cornerstone for you know, Sometime, but it might open the door a little bit more for someone else to come come next and create something else. Because it might create a playing field, I guess, because right now it's kind of, it's harder to get into developing something that that plays. So

David Williamson Shaffer 45:17
yeah, I mean, I don't think I'm selling any state secrets, you know, one of the things you're working on, right is, is getting the tools off the web. So that you could there could actually be a desk, essentially, as something that sits on your desktop, and therefore all the files are actually right there on the desktop. And so it's easier to move things between different applications, because you don't have one entirely separate web service talking to another or the person who's running, it doesn't have to upload and download things all the time, from, you know, from one package to another. So that's part of it. I'll also just add that there's going to be a special session at the upcoming IC QE, sort of about this topic. I mean, I think this is something that community really needs to do, really needs to discuss more, and part of it is maybe, you know, proposing some solutions. But I think also part of it is just articulating clearly, you know, that this is a real community need. I mean, one of the things that I've learned in society can do is, you know, speak, speak with a voice that that lets the community kind of amplify what a need is, and helped us set some of the priorities. And so I hope that's something that will come out of that discussion. Because I think, as I said, I think this is really, this is really crucial. And not just because we want, you know, to dethrone E

and Ei, although that's a fine outcome too. But because the community can't grow, if if, say, sarin codea, that, you know, are the rate limiting step for everything. Right, you know, be other people need to be able to, you know, build something, and then let everybody else use it. Without it, you know, without it being either turning into SPSS, and you say Hendrick, or, you know, or it just being the Wild West, and everybody has to, you know, be in the guts of are figuring out which function calls to parse in order to get one thing to talk to another.

Brendan Eagan 47:19
Can I just, I would like to just piggyback on that and circle back a little bit to what Sylvia's question was, I think, I think that one of the things that we often focus on efficiencies, so like, I was really impressed how Cody made for anyone to use the old web version of EA versus the new one. I can't remember what the increase in efficiency was. But it was like I think orders of magnitude more more quick. And that that's great. We push on, like usability and design and understanding things. But one of the key things I think, is trying to foreground certain epistemic considerations when you're using a tool, right? That's a really crucial thing that I think is important in mean, when you're thinking about Iraq, being forced to define certain things, right, and say, This is what I'm doing. And thinking about that. Even just the, if we're doing this workflow thing. And you could Yes, you could, you could imagine setting up all these processes that are automated and just goes all the way from like, a transcript, automated coding all the way through a ENA model, right, if all parameters but being able to to think about why you'd be doing those things, and what kind of claims it would it makes, what inferences or implications sorry, that has for for the claims that you want to make. I think it's really crucial. That so that the transparency stuff is big, being able to talk about culture. So often we talk about ENA, we think about connections, but I think the fact that we have specific stances towards focusing on on human endeavors in certain ways, I think isn't. And the thing that's coming out to me, which I think is interesting is, who the tools are for, and who gets to engage with them. So that's another kind of key piece that I think, you know, there's been some work done to not to, I don't want to set up the next webinar too quickly here and segue to it. But I think that's something that we hear is not only happening at the design level of the tools, but also we're thinking about in terms of who's consuming the outcomes of them.

David Williamson Shaffer 49:18
representation. Yeah, but

Brendan Eagan 49:19
hopefully, that's something that continues to get to define. So we I know, like, Rogers and a few others folks have a paper that's looking at this. I know there's another submission to this IC QE that's getting at the specifics of what makes something QE and how people are using these different techniques. And so I I'm eager for that discussion to I think it's a crucial discussion. I'm eager to see how it evolves.

David Williamson Shaffer 49:42
I want to leave we probably have time for one last question, but I'll just vamp for a second while people decide if they have one. I mean, I think one of the things that happened is ENA was designed to enact you know, principles of QE and thinking about cold And so on. And so it's, there's been this kind of, you know, coevolution of the, of the ideas and some of the foundational ideas and QE and some of the foundational ideas in ENA. But But I think that, and so I think that's why it's, you know, so widely use, it's a really good example of what QE is trying to do. But I think it's definitely not the only one. And I think we're, I think we're at the point in the community now, where that we're already seeing it start to happen. And one of the challenges is, you know, it can't be, you can't run, you can't have a flourishing community with sort of centralized control over the means of production. Right, not to get too Marxist, but so that there's, Does anybody else have a question they want to ask? Or should

we turn it back over to Hendrick and Brendan to close up the session? Yeah. Oh, actually. Well,

51:00
I would have one, but first, all the others here any any comment or question to? To add? No, then I, then I'd step in, because we're running out of time. But there's one question about the encoder, because I really, I think it deserves more than the lipstick. And because especially when you look at the ENA process, I mean, it's it's really kind of untrue. It's, the coding process is so painful still. And it's, it's, but it needs to be so painful. And you need I mean, we had that in the webinars before, with Barbara Watson, who in some way that these COVID analysis of all the countries and they set how much time they went into the coding and the coding was important to raise the awareness of what they actually talking about. And then the Nicene a tool gives you them the pictures of that what you assume, but actually, in the coding is the most crucial part of it. And I'm wondering, seeing PhDs in my lab doing also working with encoder starting with encoder and then same find another workaround would not be something where with text mining in any way, or maybe a challenge could be pulled out also from the cure community to improve the text mining part two degrees for a little bit these efforts and coding, and and make encode that there a more powerful tool. And maybe it's not only Caesar recorded to do that. So actually, what do you put out a challenge on something, that there comes a package that helps us to better qualified text person in English, then I will prefer German to be honest. Many Germans, text we have to do, but that will be really an ad, I think this is really a unique selling point that needs to be more than an advanced excel sheet. And then it will be a very powerful and low hanging fruit. And I know it's hard text mining is hard. So but that's the question. Yeah.

David Williamson Shaffer 53:01
So first of all, Carl, who has been very quiet through all this has basically been pushing on this for a couple of years and has a bunch of ideas about how he would like to address it. And I can't say anything, officially, because we haven't had official word, but we anticipate some very major efforts going into encoder. And hopefully, as we've just been talking, not not just our lab doing it all ourselves. It's a it's very challenging, because there are so many issues that people pay no attention to, in terms of, you know, validity of coding and information leakage and so on. But yeah, we, I know, Carl, and CSR and Cody and Brendan and I and other folks in the lab will be spending much more time thinking about this in the coming five years. And then probably we have we have thus far. But yeah, I agree with you that the solution isn't just for us to go off and like solve everybody's problems. There. There needs to be a more open process than that. And I think that's a good invitation to to start down that road with something specific. I agree with you. By the way, I've always thought that the thing that was most was way more important than ENA was that was the encoder and thinking about how to make codes valid, especially in sort of a time when there's so much text mining people can use.

54:29
Yeah, I agree with Davey, and I'm so interested in improving encode I am always thinking about this has to be improved. And I understand coding, whatever you do, you will take time. In order to get a good code you will take time. The thing is, I heard your word, pan for I don't like this word. Well, once you take time, it's fine, but muscle painful. What if you take time for this enjoyable. Under your code, you feel like, Oh, I found this one. And you have more activities integrated into it. You feel like you're discovering something every time you find a new code or noon, do you feel excited. So that's kind of a tour that I'm thinking of building. Instead of say, you know what, I give you a tour, you can code your data, you just five seconds. That's not good. But you may take your two weeks, but there's two ways you feel like is a very Happy journey. That's Samsung Jamiel. That,

55:37
I think that's a wonderful, almost closing one designed to make them efficient to make them feel that you feel good with it. That's fantastic. I pass over to Brandon, in his last minute to announce the next webinar. Thanks. That was wonderful.

Brendan Eagan 55:52
Yeah, thank you to all of our folks who actually ran the fireside chat. And I also think some of the themes that we talked about today in terms of more participatory design sets up the kind of announcement for our next webinar in the series. So we're looking forward to everyone joining Hazel and Mike, we're here to date to talk about participatory QE. And I also anticipate that they're, I know that there's been a submission for a symposium that's focusing on the same topic, because there's a lot of different efforts in this area. So look out for tweets and for emails and make sure that you register and join us for the next the next iteration. Thanks, everybody.

David Williamson Shaffer 56:35
Thank y'all. Thank you. You