Steve Sheng: Thank you for coming to the session. We're about to get started. For the space issue would we want to…it would be good? Okay. I think we are fine on space, so if you are sitting in the back we welcome you to come forward and join the group.

Male: So welcome. I am not Elise Gerich; Elise is tied up in another meeting so I'm going to proxy for her. And what we're going to do today is talk about methods by which we could enable future directory services for domain registration data.

So the objectives here are really to share information and to try to approach a problem that has been persistent in the community for over a dozen years; certainly before I came to ICANN. And over time we've been using the WHOIS protocol for support of access to domain name registration data and at various times through the cycle of analysis of the adequacies or inadequacies of the protocol or the service different parties, including ICANNs Security and Stability Advisory Committee, have asked the community to consider not only requirements for a registration data access, but for a broader directory service access that would include and facilitate access to domain registration data.

So, the first goal is to actually try to understand what those requirements are and to think about those. And the second is to better understand the existing technologies, including IRIS and RESTful WHOIS

that might be interesting candidates for a service that would support a directory service in the broader sense. What we're going to do is actually begin with a presentation of the RESTful WHOIS service that ARIN has been experimenting with and is piloting, as I understand, right now and Andy will go into more detail with that. And then we're going to talk a bit more about requirements and we'll talk about how the technical group that's been working can proceed and consider next steps.

Andy Newton: Alright. Can we go ahead and switch over to – am I hooked up? Okay. My name is Andy Newton, I'm the Chief Engineer for the American Registry for Internet Numbers. I'm going to talk about the RESTful WHOIS service that ARIN has fielded. I wish I could say that I just wandered into this space but that's not true. Some time ago, and I can't remember how many years ago this was, there is an RFC which has a date on it, I did an LDAP experiment trying to put WHOIS data into LDAP. Then shortly thereafter there was the CRISP Working Group in the IETF and my name is on a lot of those RFCs as well.

Beyond that, I've also had the pleasure of working with some of the people who did other efforts, notably RWHOIS and WHOIS++, when I was at VeriSign. But most recently at ARIN we have done the RESTful WHOIS web service and that's really what the focus of this presentation is about. But, because I

mentioned all those other protocols and because we are talking about something other than Port 43 there's going to be a lot of natural comparisons between different protocols. So let's start with the basics of how these protocols are all put together – and this is true of almost any type of data exchange protocol on the internet.

You have, essentially, a control part or plane of the protocol which deals with framing – how many octets do you send across, blah, blah, blah – security; that kind of stuff. And then you have the data part, which is really the thing, that's really the part that users are interested in.

So let's compare these protocols. So we have LDAP which is a directory service protocol; and LDAP has a pretty thick control plane which heavily dictates what you put into the data plane. And by that I mean the data part you put into LDAP has to be a tree structure, it's [ASNBER] encoded, all that fine stuff about RDNs and whatnot. It's not what I would consider a rich dataset.

Then you have IRIS, which was a product of the IETF CRISP Working Group. The data part there is a little more free form then what you get with LDAP although not as free form as what you would see on the web. But one of the central things you should

take away from how IRIS is put together is both the control part and the data part are very much glued together and they were meant for each other.

On the polar opposite of that we have this thing called Port 43, the nickname protocol. Originally specified in RFC something before DNS was standardized, but essentially the way WHOIS nickname protocol works is you have a control plane which is extremely thin and says almost nothing about what data gets thrown over the pipe.

And then you have RESTful web services, or in this case it's actually just http or the web as we call it, which has a very nice control plane that allows very extensible data parts to be exchanged. So, why would you want to go to a RESTful we service? The answer there is actually found in RFC 3707, which is one of the CRISP Working Group documents, if you want to do things like internationalization support, you want to do referrals, you want to do security – Port 43 is not really going to help you.

But if you want to do those things that lead you to one conclusion when you look at the thin veneer that Port 43 is over the data, there are no hooks there to hang your hat on for extending the protocol. If you want to do it and maintain backwards compatibility and not break half the things that are being used out there, you're going to

end up bending over backwards to get it done. So, that really leads you to one conclusion, which is go beyond Port 43.

The Port 43 solutions – we have RWHOIS which is seen as a very problem specific technology. It's used by a subset of the ARIN community. WHOIS++ which was in the 90s in the IETF – it didn't really focus so much on the data part but mostly on centralized indexes. LDAP which is actually widely used, it's just not widely used as a public internet service; it's mostly used within enterprises and intranets, but it's not really considered a public internet service.

IRIS which was the creation of the IETF CRISP Working Group, which didn't really get a lot of adoption and it really is kind of a protocol designed by committee and the requirements were over lawyered. And then the RESTful web services, which is what we're talking about here, which is nothing more than a simple reuse pattern of http.

What is REST – we didn't make this up – REST is the Representation State Transfer; it's a thesis by Roy Fielding. And it's basically when you apply it to the web you've got accesses or two premises going on there. One is to take the http verbs – post, get, put and so forth – and you apply them to the CRUD operations

you do on data. Of course we're talking about WHOIS here, you don't really do "hosts" and "puts" with WHOIS, that really leaves only "get". And then the other access is you have resources – basically everything is an addressable URL.

RESTful web services are very popular with Amazon and Google. If you were a software developer and you wanted to use Amazons S3 data services – that's a RESTful web service. What does that mean in context of WHOIS? Well for us, we have a bunch of different types of what you would call, if you were data person entities, we have POCs and ORGs and NETs and so forth and each one of those things we modeled as a separate URL that you can cut and paste and hand around to your friends.

You can put it on the back of your business cards – here's my POC at ARIN – because everyone wants to do that right. But what that does is that gives you a very simple programmatic API into the data set. Compared to nickname Port 43, you therefore have some concrete inputs that you can get in; you can standardize them if you want to. And you also have a way of expressing what type of output you're going to get. And then the side benefit for that is you get to reuse all that http infrastructure that everyone's paid for and developed over the years.

More information – if you want to know more about REST there's a good book by O'Reilly Media called *RESTful Web Services*. This is a technical book but if you picked up, if you were a non-technical person and you picked this book up the first two or three chapters you'd probably read them and go "oh okay, I get this". It's not a heavyweight book at all. So if you want to know more about the subject in general, I suggest you go get that book.

What does this mean as applying to the problem domain that ICANN is looking into – the ICANN community is looking into? Well, RESTful web services in general are not an out of the box solution; it's not something you just say go do RWS and you're done. Someone's going to have to define and decide the different patterns you use.

But once you've done that those patterns and how to express them and how people understand them is well within the mainstream of internet communications. You can go hire programmers off street who if they're worth anything they'll understand this technology. And then, on the other side of that, the RIRs, such as ARIN, we're fielding these things today, so we're actually showing how it can be done.

ARINs service – we announced I think in late 2009 and we went full production in July of 2010. It was so successful that we had

people asking us to go much further than just a WHOIS service and so we are on the eve of doing a provisioning, a RESTful provisioning service as well. The RIPE NCC also has a RESTful WHOIS service. They announced it in March 2010 and it's now in production. And while APNIC, I don't believe they have any public RESTful services, they have been using RESTful services behind the scenes for years for gluing their systems together.

Again, and the reason we all did this was the big advantages here, we could take our three and four programmers that we have and throw them at this problem because it was an easily understood problem. And everything we had, we just reuse that whole web technology, the http stacks we had and we had a solution. This is also why you have a lot of very large corporate companies, like Google and so forth, who've adopted this. And anytime you pick up your Facebook app off of your iPhone and its talking back to the Facebook server it's using a RESTful service.

What does it look like? I kept talking about the resource – all resources are URLs. So this is a URL which identifies a point of contact in the ARIN system, it's for the gentleman sitting next to me Mark Kosters. If you go ahead and put that into – I notice everyone's got their laptops open so go ahead and put that into your browser and you will actually pull up the information on

Mark Kosters. Everyone's doing it – oh, you want to see it? Mark puts a lot of information in WHOIS records.

[background conversation]

Andy Newton: Uh, no we don't. We have a very good reason for not doing that. But again, we're talking about addressable URLs so we can identify Mark Kosters as the person; we can identify ARIN the organization by saying /ORG/ARIN; we can even hang relationships off of these URLs. You don't have to know all the POCs that ARIN has connected to its ORG, you would just say /ORG/ARIN/POCs boom. There's all the points of contact that ARIN has. So you can continue doing this.

The other thing you can do is you can start formulating different search patterns as well. so if we wanted to know, we didn't know that ARIN was the handle for the organization ARIN in our database, if you wanted to search for all organizations with the name ARIN you could enter the search pattern ORGs and then I use a matrix parameter in order to specify the name ARIN. Or I could do organizations starting with ARIN and so forth. You can get fancier – we can say things like I want the points of contact that have a first name of Mark and the last name of Kosters and so forth. So that's one way of doing searches.

On the other side you have outputs. So, because this is the web there's a variety of outputs we can have. XML's a pretty good example, it's very common. And the reasons for that are because it's easily digestible by computers; it's easy to work with for programmers and you can apply style sheets to it so everyone else can understand it. But JSON's also there – JSON's a very popular data serialization format used by a lot of Java Script programmers – a lot of people who do AJAX programming and so forth. On top of that what we at ARIN support is we do XML as our primary output – the one we put a lot of emphasis behind. We also do translations to JSON, text and plain html. So we output all four of those things.

But again, showing what you can do with the XML is you have on the blue part – if you can read it – a bunch of XML data which the machines can read and people with personalities like machines can read. And then what you can do is you can take that and you can throw it into that browser and boom – you have something that's user friendly for the constituencies that you have.

Notice I said browser – that answers one of the questions people have when you talk about replacing Port 43 or doing something beyond Port 43, is what are we going to do for the clients. We can all, the different operators can write servers but if no one's going to write a client then it's kind of useless. Likewise, the people who write clients, some of them do it for free in their spare time, they're not going to do this unless there are servers out there.

So it's kind of a chicken and egg type situation. We don't really have that with RESTful web services because the clients already exist. Like I said, I showed you your browser is one of them; there are other command like clients like "get" and "curl" which a lot of Unix programmers and systems administrators now. And when I say you can go out and talk to, you can get any programmer worth his salt to help you with a RESTful web service, they'll come back and say well I need an embedded client of some type and there's dozens and dozens of http clients out there for people to choose from depending on their platform and language.

Again this is all just applying everything we do with the web today to the problem of WHOIS using RESTful patterns. So when you use the infrastructure you also get things like caching – http caching is a well known part of the infrastructure so let's say for instance – and this is a real example – we have people that query the ARIN services, I think our WHOIS services 99% or our WHOIS queries are "I have an IP address, I want to know the network." That's very common.

And it happens that you have security analyzers like intrusion detection services or spam analyzers or so forth that are getting hits against them and what they want to know is they want to know the information behind where the attack is coming from. And as opposed to constantly querying us for the same IP address over and over again, if they took

commercial off the shelf caching infrastructure and put it between us and them they would actually get much faster analyses.

You can embed referrals - with http you can embed referrals right at the http level if you want to. And some of our, if you use our WHOIS service, especially our UI front end, you'll get referrals at the http level. And if you wanted to you could actually embed them in the payload that you return as well. so what that allows you to do, that allows you to link between your entities within your own system and if you wanted to link to entities in other people's systems.

Furthermore http has authentication and authentication is the first piece of the puzzle if you want to do authorization. And what that means is you can, the community at large can then start thinking about authorization policies that don't start off with everyone gets the data. And finally one of the other benefits of reusing all this infrastructure is you can get versioning. One of the problems with some of the protocols is we want to, you've got a data schema based on the type of data or the type of data model that you had today and then in five years you need to add something or change something or you want to up version it or something like that. And what happens is you have to have a mechanism to version those schemas; version the output of the data you're using. With http you get that with a simple except header; it's built in right there.

Again as I said, to do this for the ICANN community, you can't just say go do RESTful web services, someone has to actually create a standard or a specification. There's a couple of things you want to do – you want to define URL patterns, you want to define schemas of some type – I don't recommend XML scheme if you're going the XML route. You could also take the work that the IETF did in the CRISP Working Group to create DREG and then you define your patterns and boom you're done. That's what it takes. And that's what we did actually.

So, let me talk to you about ARINs experience with a RESTful service here. I would actually not be truthful if I told you what we did when we came up with this RESTful web service was we went in the office one day and said "hey you know what the world needs is a WHOIS RESTful web service." The world does need that but that's not what our primary motivator was. What happened at ARIN – we were doing a lot of re-engineering and a lot of it was around changing the data model in our registration database to better support DNSSEC and lame delegation checking. But what that meant was we had to rewrite just about everything in our system and we're in fact still doing that. Everything, I think, but the billing system had to be redone and we just did that too as well.

But that meant we had to redo the legacy WHOIS system and so when we took a step back and said "hey we're going to do this legacy WHOIS system, first off let's go above and beyond what we already have and let's do it better." We also wanted to be simple about it too. So the

approach we took was we had already come up with this thing called ARIN Online which is basically people can log into portal – I don't know what you want to call it – a self service website for ARIN and they could administer their data. We wanted to take that code and reuse it for the RESTful web service as well.

And what I did was over a matter of I think a couple of months, it wasn't a long time, in my spare time when I wasn't attending meetings that Mark scheduled me into I was actually hacking against this trying to come up with a technology demonstrator. It didn't take me a whole lot of time to do it.

But the original idea was we were going to have a RESTful web service that reused our core libraries and talked to our database. And then to continue Port 43 service we were going to write a Port 43 proxy which then just took the current WHOIS proxies that we had and made them RESTful calls.

The level of effort on that – we didn't really break it out into protocol versus non protocol, but once it was greenlighted we didn't have a large team on there. We had a single senior engineer for a couple of months, I think only two, and then once he got going we actually assigned a couple more engineers to it. And lo and behold we had it. But like I said, we were also solving some non protocol problems as well.

We created a new replication technology specific to ARIN; we also at this point, until we did our RESTful WHOIS service, we didn't actually support CIDR queries either and that was one of the things we wanted to add in there; so we now support CIDR queries. We had to come up with ways for doing efficient CIDR searches.

One of the other things we did is we spent a non trivial amount of time trying to make sure that when someone queries Port 43 and did these calls into the RESTful web service that the data that got put back on Port 43 looked like what they used to have on Port 43. That was quite expensive.

Adoption – so I said we announced this in 2009, I think also at the time we put out a pilot service. If it wasn't at the same time it was shortly thereafter. What we had immediately was we had a couple of people who wanted o reconcile, basically they were doing auditing of their own data. So they were auditing the data they had, that they had told us about their networks and their infrastructure, versus what they had in their own database.

So they were actually using our programmatic interface that we put up as a demonstration immediately. We also had someone who wrote a Flash application – I never wanted to go prying around and you're not

really supposed to be investigating who's querying you like that, but we could tell it was a Flash application. So someone before we even went to production with this actually had a Flash app on the web for this thing.

After we released, our numbers were kind of hard to come by because things kind of went berserk. Our query rates skyrocketed both on Port 43 and Port 80. We went from 600 queries a second to around 5600 queries a second the day we did this. And we've kind of had some up and downs in the query pattern since then. Things have settled back down a little bit, but they're still higher than what they used to be.

I believe in the back of the envelope, what I was just checking on today actually, our RESTful queries are now 18 to 20% for 2011, 18 to 20% of our overall query rate. So we've got quite good adoption. One of the other things is, based on immediate feedback from the website, the people coming over from Port 80 before we actually launched the service our weblog statistics show that it wasn't well used; once we launched the service we got feedback from people saying "hey you know what, I like to use the Port 80 stuff but here's a way of making it better."

So we came up with this thing called PFT which basically is a pseudo resource in RESTful terms to help people who are using browsers to get our data better. And again, as I said, the demand for the WHOIS RESTful

service nearly immediately after going to production with this thing we had several people approach us and say hey this is great we can finally see our data in a scriptable way in your registration database; now what you need to do is you need to allow us to modify it via RESTful web service. So we're actually on the eve of doing that – in fact, Mark and I've got to fly back tonight because that's going out in a matter of days.

In conclusion, if you want to see our service, it's production – whois.arin.net – we also have a technical mailing list for all things technical at ARIN called arin-tech-discuss@arin.net. You can send questions there or you can send questions directly to me if you want to. And that's it. Questions?

Male: Hello. It's great to see that you are adapting to this great technology from the web actually and I was wondering that there are actually a few data that are expected from the WHOIS service like name – organization name, first name, last name – and other than…most registries are also using some custom data like the K4 and sunrise issues and other cases.

But I was wondering if there's going to be, are you going to use for example, name spaces and there would be a naming space for the common data and registries can define their own naming space for additional data? And also I want to know if there is going to be any, (inaudible) that are going to be used here too, or it's just XML and that's it?

Andy Newton: For starters, we have a specification – it's an ARIN specification, it's not a standard. We haven't gone to a standards body with this. The way we did do it is we created a relax ng schemas, which we publish for XML, and we defined a name space for the core data sets that we have. We also put hooks, plenty of hooks, everywhere, every place we could in the XML to say you can put in XML elements from another name space and we don't care what it is, it will validate.

So later on, and we have done this, we actually had to come back just two months ago when we did another version of our WHOIS RESTful service. We actually added in other elements for our delegation queries which we're about to introduce real soon now. So that's the first part.

We also, because we're doing output, and I guess we could do this on our provisioning side, but I'm not really all hot about doing it there. But our WHOIS, our RESTful WHOIS service supports both XML and JSON and plain text and plain, well I guess it's application HTML, straight HTML, actually that's XHTML but. So we will actually output in those four formats if we want to.

And the JSON layer, which is probably the most interesting one versus XML, is really just a translation library that we have – it's called BadgerFish, we didn't write it, but we found it and we went hey this is

kind of cool. So all we had to do was create an XML version of what we wanted and we automatically get the JSON version as well. For the plain text and the HTML we had to write style sheets and that's how we did that.

Male: But what I meant was for example (inaudible) you can create the lists and you can actually talk about those data on the web for example and additional repository and you know?

Andy Newton: I'm not following.

Male: Okay.

Steve Sheng: If you can, maybe take it offline. Gentleman on the right next.

Lutz Donnerhacke: Despite I am not in favor of putting everything on http and creating an overlay net over the internet, I'm pretty impressed by this service here and two remarks. First remarks – please provide specifications to the XML so that we can validate the response automatically – first point. And second point – it's already missing on the right side – and second point, provide a version identifier. So if you want to change this interface and you are aiming to automatic tools here, you can still run the old interface without changing it.

It would be fine, we can refer this to saying oh we like to have the version 1.3 for all communication and then if you the other in production and we have changed our tools, use the other version. It would be fine if we can have co-existing versions but we do not have on WHOIS service at the moment.

Steve Sheng: Thank you. I forgot one thing – next time when you ask a question please indicate your name and your affiliation.

Andy Newton: Yeah, so as far as RIPE NCC goes they kind of did this at the same time that we did and it wasn't a whole lot of collaboration going on. And they told us about it and we told them that we were doing this. And from the perspective of both organizations it wasn't a "hey let's try to standardize something", it was a "hey let's try to see what we can do." The RIPE NCC and ARIN have somewhat differing data models, somewhat differing ways of provisioning the data. So the original effort wasn't to try to come up with a standard, it was to try to see what each organization can do.

If they don't provide a schema, I can suggest that to hem next time I talk to someone from the RIPE NCC, I guess. We do, we actually have a zip file somewhere on this webpage, I don't want to go searching for it, but

we actually took all our relax ng schemas and zipped them up and they're publicly available.

Oh, Mark's pointing it out – it's right here. But if you go to the whois.arin.net on the sidebar, there's a thing that says WHOIS technical information, API documentation, and you'll get this webpage. And basically it describes a FAQ and then there's an actual specification – I guess you'd call it our API documentation on how to do this is in here somewhere. So, that's what ARIN provides.

Steve Sheng: Thank you. Jim?

Jim Galvin: This is Jim Galvin from Afilias. I want to make a comment and just have you confirm it to make sure that I heard you correctly. I think what you actually, you did at one point mention that there are two things here that you have talked about and presented – so we call it RESTful WHOIS but you've actually provided two things. One which is a description of the service level, which I think is what's all about building something on top of a RESTful framework right?

So there's some data that's back there but you've got this layer on top which is the service which provides all the access to it and the mechanism for querying that data stream. And you did comment at one point that there's also a data model that you had defined and in fact, I

believe you were saying that what motivated this work in the first place was that you wanted to change your data model and all of your data – you're nodding your head and that's good. So I'll go on.

I think what's important here is just reminding ourselves from the Cartagena meeting, we had actually said before that there really, when we say WHOIS and every time we say WHOIS, what's important here is there really are three things, at least, that we could be talking about. One is the service level component; one is the data level component; and then of course there's the protocol underneath. And we need to be careful about those three things and recognize that we're talking about three different things and what they are. So, thank you.

Andy Newton: Yeah, it might be RFC 3707 – one of the RFCs, there's some RFC somewhere that actually does spell that out and I can't remember which one it is, but yeah.

Steve Sheng: Dave?

Dave Piscitello: Andy, I have a favor to ask of you. If you could, you're on a MAC right, so if you could open up a terminal window and you could go through the following demonstration – just do a "curl" command on that same URL that you grabbed like on 192.168.00 to show, you can show the XML output so you can probably see that.

Andy Newton: Let me do one better, let me use xmllint.

Dave Piscitello: That's even better.

Andy Newton: So, if you use xmllint, which is a program which basically if you're a C programmer you use the lint, xmllint is kind of the same thing. It lets you know all the bad stuff you've done. So I believe this is the one for Mark and let me…So, there, that's all I did to get Mark's information. And if I wanted to be, depending on how perfect I wanted to get it or whatnot, I have written simple, very simple CSS style sheets and there's another command line program called xsltproc which you can run, you can give it that same URL, run it through a style sheet and it would just pull out his name or address if I wanted it to; or whatever I want.

Dave Piscitello: Yeah, the reason I asked to have this done is for those of you who are not technical and just try and appreciate some of the subtleties. This is the raw data that someone who is trying to use automation or a program to analyze WHOIS, for example, someone who is looking for a potential point of contact for a phish domains, a phish domain or the lie, would be able to sue this structure which if it were standardized across all TLDs would be a tremendous benefit to anyone who had to use WHOIS in large volumes.

Steve Sheng: Thank you Dave. Danny and then Jay.

Danny Macpherson: So, Andy I've got a couple of questions. First, you talked a little bit about this up front – there's three parts, I'll just list them all and you can whatever. Anything unique with regards to IDNs where this position is better – I know you commented on that. Second one is from a standards or a publication perspective, what might need to be done to get this to a more wide audience or with a standards scheme – whatever that looks like at each of those three levels Jim outlined. And finally, what you think may be challenges in particular if this were to be adopted for domain data for example.

Andy Newton: So, I put in the presentation "internationalization issues" which actually encompasses more than IDN. IDN is actually a very particular type of internationalization issue. There are – the CRISP Working Group actually touched on IDNs. They had a thing called IDN Variants which came out of, also reused of something from the IDN Working Group as well. and basically you could express, you could query against a variant in IRIS, you could query against a domain name and get its variance – that type of thing.

And I believe that was the support that was put in for in IRIS. It's all XML in IRIS; you can do the same thing here. You'd do the exact same thing here. I'm not an IDN expert so I don't know all the intricacies of

what you would need to support it, but that's what IRIS does and so you could easily replicate that here.

As far as going to standard or specification or whatever, it depends on the standards body. I guess if we were talking about the IETF we would have to go through the standard process of doing a BOF or whatever and then getting the ISE to agree to a working group. Yeah, I mean, a standard is really just what people read off of a piece of paper though when it gets down to it, but if we wanted a formal standards body to seal of approval, that's probably the way to go.

And as far as challenges go, like I said, the implementation of the RESTful side, the actual web service, for us was fairly simple and straightforward. Our biggest challenge was trying to get Port 43 to act right. We rewrote our Port 43 demons and we, they're basically these big proxies that take the WHOIS query and turn around and ask for RESTful query in response. And our biggest challenge was trying get that information back to the users in the way that they had seen it previously. And trying not to, what we thought would be breaking WHOIS clients.

But that's a huge, that's one of the reasons why Port 43 is such a hassle is because you really don't know what you're breaking when you change Port 43. There's no specification out there other than send line of text, carriage return, wait for the socket to close on you – so we had no idea

what the client behavior was. And we are still today discovering different things.

And one of the things we did, and this actually intersects with IDNs, was during our analysis we realized in some certain WHOIS clients when you put a dot in the query stream for the WHOIS client, in some cases for some reason it interprets that as an IDN. I don't know why. I don't know why "space, dot, space" is an IDN, but it is to some WHOIS clients. Turns out that "dot" was actually one of our flags for signaling, I think, names – I want to query a name record, I want to query the name of a record for our query formats. So what it did was it meant that some people couldn't actually query our WHOIS records properly. So we actually had to come up with another flag and work around that. That was an IDN issue.

One issue we kind of came to grips with just recently was some WHOIS clients, when they query ARIN for AS records, they query us differently then what they query RIPE and APNIC for. And the clients don't tell you they're' doing this they just do this. And so we had some unexpected results of people with different WHOIS clients doing the same query but getting different results back and no one could figure it out until we actually went and did some snooping and went oh wait a minute, the actual WHOIS client is changing the query out from underneath people. And that's because the client rider had a misconception of the way WHOIS records, AS records were formulated within the ARIN system.

Steve Sheng: Thank you. Jay?

Jay Daley: Jay Daily from .NZ. Two things – first of all I'm right in thinking, well tell me if I'm right in thinking that the only bit of the technology that you've presented here that can't be replicated in WHOIS is the use of standard web based tool such as caches, security analyzers, statistics gatherers and web libraries. But everything else in terms of specifying syntax for queries, specifying syntax for responses, and on the other side of things can all be replicated through WHOIS if we chose to do that.

Andy Newton: I would say wrong. Security can't be done; I don't know how you're going to do, unless you want to create another Port for SSL over WHOIS or something like that. If you wanted to try to put whatever hooks in for things that are equivalent to a user name and password, that would be kind of, that challenge response would kind of be difficulty over WHOIS. The other thing is that if you wanted UTF8 or UTF16 over WHOIS, you have no idea which clients you're breaking. IETF actually went and discussed this, that very issue before and they determined that you could not actually say that a WHOIS server can return UTF8 safely.

Jay Daley: Okay two things – on the security side I think it's possible, but through things like client certificates which aren't necessarily practical for a

different reason. And on the UTF8 that's possible again through a standardized flag, but again it's not necessarily practical to do that; that's a different matter.

But the second question then is would you say then that there is a tradeoff here in terms of the ease of use with the complexity for a developer now, for a developer to create something that talks to WHOIS on Port 43 is trivial in that sense – open the Port, send the stream, get the response back and that's it.

With web protocols that's more complex and it's also more complex for the implementer of the server to do that as well. They can use a tool kit to do it but say they want to add something like token bucket algorithms for flow control – that's going to be much more complex then to do with a web based interface.

Andy Newton: No, I would also disagree on that too. Speaking as someone who just went through writing a Port 43 demon, we had to basically – I forget the name of the package we ended up having to use – but we had to do a lot of special tricks with non blocking IO in order to get it to perform right. For the web side, the Port 80 side, we used Apache. So there was no implementation from the demon level that we had to do for the RESTful web service at all. Period.

We basically took our core library that talks to our database, put a RESTful engine on top of it and boom, we were done. It was a matter of weeks for me to do that as a side project. The actual Port 43 stuff was a lot longer and more involved. And you have to be, if you're going to be someone who writes TCP demons that take load you have to know what you're doing.

As far as a client rider goes, it really does depend on who the programmer is. There are some people who are programmers who have no idea what a socket is a TCP socket is. I've run into them. They're not stupid. They've just never done it before. However, they know what a URL is. They know that in certain environments they can just say here's a URL, I hand it to this thing, I get back data – and it's that simple. Whereas with a TCP client you have to know I'm going to open the socket, I have to know how to read off the data – and for WHOIS I especially have to know not to close the socket prematurely.

Steve Sheng: Thank you. Dave.

Lutz Donnerhacke: Lutz again. I love this presentation here, this output. It shows how common this misunderstanding this created. Because you can read the output doesn't mean that you understand it. And this especially it's an [owl], it's completely easy to misunderstand by saying there is field called "street address" so everything there must be address of a street –

it's not true. You need the semantics. You need the syntax definition here. You need the schema.

Without it this is output is completely useless. So please always provide syntax, semantics, provide API documentation and then – and very, very good documentation here and then say we can use it. Otherwise we will run in to the same problem as we have in the WHOIS now – people look at output and say oh I understand it, use some regular expressions and say I have a programmed declined, it's doesn't work with XML.

Andy Newton: First off, I know plenty of programmers who've used regular expressions against XML and HTML over the years. That's a very cheap way of parsing it if you want to do that. It's not what I normally do, but I know plenty of people who have done that. In our documentation which I've got up here – I can try to increase the font – basically, we talk about our data model – the different types of things. I don't know how many people are familiar with the RIR space.

It is a somewhat different data then what you would have in domain names. We have things like autonomous system numbers which if you're a domain registry/registrar you've never heard of before. But some of the information is the same – we have ORGs and contacts like you guys would. With DNSSEC we have records for supporting our [R sets] in DNSSEC and so forth.

But we talk about our different data models in here. We talk about the command line clients you could use, basically how to specify different data formats and somewhere in here is our zip file which contains our schema and so forth.

So we actually do provide documentation on how to talk to our service. Is it complete? I don't know. I'm sure that if this was to go to a standards body you would have people who are actually professional writers or professional standards specifications writers coming behind us and saying well you did a good job here, but you could really beef this section up. And I'm sure of even our documentation.

Steve Sheng: Thank you. Any additional questions?

Dave Piscitello: I just wanted to respond to Jay's analysis of the difference between WHOIS and RWS. I think one of the things – you can say one of two things if you go back to Andy's analysis of the control plane and data relationships. If you look at the original nickname protocol the control plane was literally nonexistent. There's only two criteria – you send a message and you end it with a carriage return and/or line feed. It doesn't even say that you have to send ASCII 7. It simply says the carriage return and/or line feed is ASCII 7. Well that's as slim a control plane as you could possibly imagine.

So as a protocol architect one could say I could construct any signaling protocol for the control plane over that channel so long as that channel continues to have the properties that the message is terminated with a carriage return and/or line feed. In fact, that's exactly what's happened in a non standard fashion in WHOIS among all number of permutations and combinations of clients and servers.

So if you want to go and get Conge back from .jp, you can query on a command line with WHOIS and you specify like a minus H, but if you used a minus H at a dozen other WHOIS servers you would not get back Conge. So one of the problems that I think we do have to recognize is we have a legacy issue in trying to take WHOIS. There's no version. There's no standard error codes. There's free form and per – almost per server data framing in the message response.

So when Steve Shang and I looked at a lot of these issues, when we were first asked to look at the service requirements and catalog them for the GNSO, one of the things that we did try to identify were all the things that were not standard and free form that might be better served if we had a schema, if we had some standard semantics and syntax. So that was one of the things that led us to really take a better and closer look at RWS.

Steve Sheng: Thank you. Any other questions?

Jay Daley: So just one more. Dave, isn't that really the point though – a standard schema and standard semantics and all the things that we really should be talking about and that comes first and then after comes the technology by which we use to represent it?

Dave Piscitello: And I guess my response to you is that we've done that. And it has been presented to the GNSO almost a year ago. I'm certainly happy to continue and we're in a transition from what we're doing now and look at some of the high level of that document, but it is a substantive work that was produced by staff at the request of the GNSO some time ago. So maybe we're all at a different point in time in terms of looking at that list of requirements and whittling down or expanding according to what the perceived need is.

Jay Daley: Well of course, I'm not from the GNSO; I'm from the ccNSO that is not involved in that process at all. And I also have some colleagues here as well. And I think that's interesting but I don't think it legitimizes a decision that those are the set of new features that we should be putting, you know, we should be looking at. I think there still needs to be a wider discussion.

Dave Piscitello: Well, let's understand that what we have is a list of features, not a requirement set. Okay? And I think that the choice in the document that use the word "requirement" is one of those pitfalls that you can spend an awful lot of time in the snakes and weeds in an ICANN community. But when we were putting those together it was a list of things that were desirable. Not necessarily for everyone, but that someone at some point in time in the chronology of WHOIS discussions said was desirable.

Steve Sheng: Okay.

Michael Young: So Dave, I think this document, I've been looking through it since Steve sent it out to the list and the pointer and it's actually a really great document. There's a lot of good information in it. but I think you touched on the fact that it's not really, it's called a requirements document, but there's still I guess some room for interpretation on some elements. Some things in it are kind of recommendations almost versus hard core requirements. And there's almost in the way it reads right now, there's a bit of a prioritization implied in which element comes first; what's discussed.

A natural prioritization in the document itself, but I don't know if that's an agreed upon prioritization because obviously there's a lot of work in this document and you wouldn't do it necessarily all at once. I don't know that it would even make sense necessarily to do it all at once.

What happened with the GNSO with this – I'm not really familiar with whether or not there was a mandate set by the GNSO to go ahead and produce work against this document and actually move ahead and orientate some type of work force towards it, or working group towards it.

Liz Gasster: Hi I'm Liz Gasster from the ICANN policy staff. The Council hasn't talked about that paper yet and it was written in July of 2010. We've been trying to get it back on the agenda because they've been so overloaded and actually Steve and I just briefed them again on it earlier today at the open meeting. So what we're going to discuss in tomorrows after meeting is how we're going to discuss that further. The Council needs to talk about it. But I think one thing is I'd be kind of interested in what you all think the Council should do with it. And one possibility is to do nothing itself and to let you all work on it for a bit; the community, meaning the broader technical discussion community.

And I'm just throwing this out because first of all they're swamped with policy work and because I'm not sure I see this as a policy, the next immediate next steps as policy steps; I see them as something much more exploratory and you all know better than I, but a different kind of conversation. So in a sense, what I want to make sure is that that report doesn't just languish with no one reading it and no one deriving any value from it since the Council asked for it and we did it. But my own reaction is to think it might

be of more use to many of you then it would be for the Council right now.

Michael Young: Sorry, forgot to announce my name the last time I spoke. I'm Michael Young and I'm with Tiny Planet. I used to be with Afilias and some of you might know me better from those credentials. My thought, Liz, to your question is that I almost want to take a bit of a clue from the IRT stuff and I think if we're going to get work done on this there has to be some type of real focused momentum. And I'd like to see if we can get someone in the GNSO to put forward a resolution, or propose a resolution.

And I'd be happy to draft one and put it through the Council, see if we can get someone to put it on the table, for a working group – yeah – for a working group that would be having a very focused time period to go through this document and actually determine agreed upon requirements out of it so we can get some work going on it. And I would say like a 60 or 90 day period – put a real – and the material is all here it's just a matter of getting consensus and decisions made and obviously you'd want some representation from interested stakeholders in that working group – I think we can knock it out.

Liz Gasster: Could you just elaborate for me the consensus and decisions that you think could get teased out in that 60 and 90 days. Like what's your image of what that is for the Council to say that these are the

right requirements or the wrong requirements or modify the requirements in some way – is that what you're…?

Michael Young: There's some fuzziness in these requirements. We could do it this way – we could do it that way; so those kind of questions need to be clarified in it. There's some suggested requirements that I don't know really need to be done right now – maybe they do, I mean it's a nice idea. The WHOWAS in an example that comes to mind; and I think that's kind of – I like it personally. But on the other hand when I'm sitting here thinking about work we need to get done right now I'm not sure whether or not that's something that needs to be focused on in the short term or whether that's really kind of another service altogether.

Liz Gasster: You know, I guess I just really – this is a very candid, personal opinion – but I don't think the GNSO is ready for this. I don't think they understand in detail those requirements. It's overall – and I don't want to speak about them in a monolithic way so I hope everyone takes this with the respect that I mean. They're swamped with other issues. These are very technical elements.

I don't think they really understand or have read the report or even appreciate that list, that compilation list fully, that Steve put up. Some may really in a very deep way, but overall I think there is kind of an education necessary. And they asked for the report in May of 2009 – we wrote it in July of 2010 – they haven't looked at it since. I don't think they know what to do with it. And my own

sense is if it would be at all possible is for you all to help them. Take a look at it yourselves and say if you were them what would you change or clarify or make less fuzzy and kind of present that.

Dave Piscitello: I just want to make one important point. As Jay pointed out earlier, and I think it should not be under played, he is not – he is from a ccTLD and there are other, at least five other ccTLD participants that I can count and I don't know you all. And it seems to me that one of the values of having this committee or this group of people work together is that WHOIS that is implemented by the gTLDs or a successor directory service that's implemented by the gTLDs in a dramatically different fashion than the way that the ccTLDs would implement it is not a very desirable outcome.

So, it seems that it would be much better if what we have is technical people who want to get together from both the ccTLD and gTLD community to try to work together and come up with some – my best scenario would be some thoughts for how to do some experimentation. I'm not even looking down the road for policy yet.

But one of the things that we talked about – we meaning some of the folks who are experimenting at ARIN, some of the folks who are experimenting as ICANN technical staff who are protocol engineers – is does this work and how does it work? What would we need to do if we through together a prototype and we actually proved that we could do it, what can we learn from it?

And I come from the IETF side of this world where we put together rough code and then we sought consensus. So we tried to understand what it was we were going to throw over the wall as engineers to policy before we asked exactly what are the requirements. So I think that where we are at this point is we've got a very, very large pile of possible requirements and we've got this kind of protocol that from 50,000 feet looks like it might be able to solve a significant amount of the problems because we've seen those problems solved in the http world for a number of similar or like problems.

So, one of the things that raises my enthusiasm about getting you all together is let's see if we can just agree maybe the right thing to do is to play with this protocol for a while – see which of the…grab some of the requirements, a nice subset and say if we had to identify even like five of the requirements that are back breakers right now – like internationalized registration data – choose three or four of your own.

And then a standard schema, standard error reporting – those are the ones that seem to be really important to me. If we were to prove that it works for those and we were to experiment with it, would that be a useful way forward to say look, we've made some progress. We've headed down this path and so far we've got some success.

Steve Sheng: Jay, go ahead.

Jay Daley: Yes I think it is very useful but I think that we can actually try to narrow this problem slightly even further. To me there are, of all the requirements that we've seen people discussing and documenting, they come into categories – those that can possibly be implemented in the current WHOIS without breaking it very much if at all and can solve some urgent problems, which specifically I mean things like UTF8 and other things there. Then there are the other ones that we all know simply cannot be done in WHOIS and which need to be done in a separate directory service.

And those to me are two very different streams and what I'm worried about is seeing the urgency of issues with the former, with the WHOIS things, trying to drive the establishment of the latter. And that's what I don't want. That's why I want the little bits of the WHOIS that we can do, that we can fix, that we can sort out we do, we leave that to one side and we then give ourselves time correctly to do the new directory service with all of those feature sets in it if necessary.

Steve Sheng: What do people – Jim, do you want to say something?

Jim Galvin: So I heard Liz suggesting that this group should form an opinion and offer some comments back to the GNSO, but I'm going to ask a question that I asked in Cartagena, and I think it still applies here too – from where does this group derive any accountability to do anything? I mean it's just a workshop. We've been putting stuff in front of a group of people but to what end? And you also

comment about, with all due respect, educating GNSO Council members and I mean that quite sincerely, but if we're going to do that why aren't they here just sort of taking all of this in like we are too? I mean it seems to me if that's the process we're going to go down we should be in their meeting.

Steve Sheng: Go ahead Mike.

Michael Young: First of all, Dave, jumping back – I think ideally, and I'm not exactly sure how these things would work across different supporting organizations, but that would be ideal right, to make sure that any stakeholder supporting organizations are in agreement or in consensus or in support of an effort around this.

So, I didn't mean to focus on just the GNSO, but when I heard Liz saying, and maybe I misunderstood Liz, was that the work that was done to date was kind of driven through the GNSO. So I was thinking full circle back to them, but it's not just exclusive, the interests are not exclusive to the GNSO at all. In fact, the cc's probably have a more immediate concern or need with IDN registries going up, IDN .IDN registries in the cc's.

And I think Jay's point of trying to separate out both like the short term stuff that needs to be addressed and then maybe working a longer project, Dave, which is kind of where I would see doing prototyping and proof of concept and so forth, would be important. Because I see value in what you said Dave, but I'm concerned that

we have a clock issue here and if we don't get something addressed you're going to end up with a bastardization of different technologies again because people just need to get the job done.

Dave Piscitello: Well that's what we already have – we're there whether people want to admit it or not. So I…

Liz Gasster: I wanted to respond to Jim.

Dave Piscitello: I wanted to respond to Jay so that trumps-

Male: I think one comment is necessarily why aren't these people showing up. I think they just simply don't care. What they say – give it a spec, we'll do it. Somebody please give us one. But they're really so tired of going from one meeting to another and talking about the way things should be done, ideally for most of us we'll be like when the escrow specification was developed by someone.

You can say who cares what kind of escrow we do, we just send it down there. And of course what worries me, we say we make it a long process of defining things the right way, most of the stuff that's been done hasn't been defined the right way. They've just kind of somebody proposed them and they were done. and maybe to find the right way would take too long that it's not going to go anywhere. So, about addressing the existing WHOIS protocol – that's nice.

If we had done that 10 years ago it would have been okay. But right now we want to have a good pretext to say we just sent down a Port 43, the domain name and say that's it, it exists and nothing else and if you want to have anything further for to these in protocol to pick it up. That's the thing that we would like to be able to in the context of gTLD registries and registrars who also run the WHOIS services.

Dave Piscitello: I'm going to trump you because Jay spoke first so I'm trying to re-establish sync here. I actually like the idea of trying to identify the things that, at least you speculate we could do with WHOIS today, because we're probably going to have an interesting discussion about that.

` And I have no issue with doing that in parallel with looking at a successor because I think that's what we tried to do with IPv6 and what we ended up doing, for those of you who don't know, is that we ended up trying to retrofit and put everything into IPv4 that we could, that looked like it was going to be in IPv6. And one of the consequences was that we stalled on IPv6 for nearly 15 years. So I would consider that an undesirable outcome but I would accept it because I'll be 70 at that point and I wouldn't care.

So with that little caveat, it sounds to me like there is the possibility of a couple of different ways to approach this in a very tech-y way. And back in the day, in the IETF, we used to do these things and have skunk works projects or experimental projects in

that we do some bake offs – we think we can do this in WHOIS, let's see if we can go prove we can do this in WHOIS. In the meantime we also have the same challenge of doing it in some other protocol and seeing what the complexities are.

We used to sit down technically and intellectually go this is how much time it took, this is the amount of code it was, we shared the code – and one of the desirable outcomes of those is usually that there were implementations that could be referenced implementation or open source implementations that might be made publicly available for anyone to use. Which means that there might be some reasonable good quality and inexpensive or free code available for everyone to build on. And that's usually, I know that lots of people don't like very, very closed gene pools, but in these particular cases that's also a very useful way to build things; monocultures aren't always bad.

Jim Galvin: Before you respond to me, this is Jim Galvin, let me respond to Dave and to Jay here then. So, it's okay if you don't get to my question, I'm not…You know, in this issue of experimenting – so you're talking about experimenting at one particular level, which is at the service level. I mean at least that's what it sounds like because to me the foundation on which all of this is built is the data model.

And I would assert that we have no agreement in ICANN as to exactly what is registration data and what that data model is. And

the obvious distinction to put out there is that the rules are different from registries than they are from registrars if you look at the WHOIS requirements. I mean the differences aren't necessarily big, but there are differences. So, if you're going to experiment, you're adding another variable into the experimentation. Maybe that's okay from the point of view from an experiment, but I think it's important to call that out. I mean that base is missing.

Steve Sheng: Kathy.

Kathy Kleiman: Yeah, I'm Kathy Kleiman. Sorry to be late to the meeting and I'm going to issue a big disclaimer which I hate doing, that I'm doing this in my personal capacity and not as Director of Policy for PIR and not as vice-chair of the WHOIS Review Team. We've got a big problem here – the GNSO doesn't know what it wants and I'm a veteran of the WHOIS wars of the last eight years.

We should have done this 10 years ago, but the GNSO couldn't figure out – and the DNSO before that – couldn't figure out what they wanted in terms of requirements. And that's really hard because as a former programmer how can you create a system when the community can't give you their requirements. This is a big problem. I wouldn't put it on the fast track into the GNSO. You're still, you know, if after eight years we still can't resolve it we're not going to suddenly resolve it if there's a quick deadline.

One of the things the WHOIS Review Team is trying to do is pull out and try to figure out why we've been stuck on the issue of the data and the control and the directory service controls for so many years. The ccTLDs may be the answer. Jay – there is certainly, one of the things that the GNSO will be looking at that the WHOIS Review Team is looking at is do the ccTLDs provide some kind of model, some kind of array of best practices, some kind of array of variations of what's been done looking at the different problems. So I think that may be an interesting way to go is working through with ccTLDs and ccNSO, other models outside of the gTLDs to move forward technically with new options.

Liz Gasster: I think actually Kathy is saying much of how I was feeling in wanting to respond to both Jim and to Michael. I'm very – you can tell I was very hesitant about the GNSO primarily because the GNSO is a policy development making body and I just don't think this is ready for that yet.

I think many of us want to put our discussion groups into homes. They just, it's the ICANN way. You know it's an SSAC thing or it's a GNSO thing or it's a ccNSO thing or it's an ALAC thing and latterly we're getting really big on these cross community working groups and that's another way to handle something like this – to do it in a cross community working group.

The thing that's making very uncomfortable about formalizing any kind of group within the ICANN auspices is because so many of

you have told us that it's not ready for policy development so what's ICANNs role. Or that's it's a standards development issues and so why is ICANN assuming a role in standards development. I'm very sensitive to those concerns.

I think what we are trying to do is share information we have about the limitations and some of these alternative technologies and thank you so much for coming today and sharing your experience because we want to facilitate progress, but we're also very respectful of the fact that ICANN does have a limited role, a limited scope. And that the SOs are policy development institutions. Do we think the SOs – that this is ready for the SOs to work on?

So there are different constructs for how we can continue a dialogue. The one I kind of started with was just let's set up a wiki, let's set up this discussion group, let ICANN provide you with the forum, the facilitators, the note takers, and the expertise that Dave and Steve and Francisco bring and you help us figure out where everyone should go.

If people are still thinking that there needs to be some auspices under which this occurs within ICANN, just the discussion, than I think it's also possible to come up – you talked about a resolution – it's also possible to come up with a model where a cross community working group could be established with requests for technical contributors with a lot of warnings that we're not going

in the policy direction and that the purpose is to funnel the information that you developed into the process you think it belongs – like the IETF or wherever you think it belongs. But I'm just conscious that that raises concerns that we have to answer – Steve and Dave and I – about what ICANN Is doing in this bailiwick.

Dave Piscitello: I'd like to make the observation that there are people here from ALAC, from SSAC, from the GNSO, from the ccNSO – its' already a cross constituency, cross SO/AC initiative. And from the fact that I didn't see a lot of people glazing over through Andy's presentation it seems to me that a fair number of the people that are in the room, especially the ones that have remained have enough technical inertia to want to continue to dialogue. So at the risk of over stepping my role, perhaps what we could do is try to formulate a request to the Board and say we would like in the same way that SSAC made a request to the Board in SAC 37 to form something, some group of SO/AC experts who would look at internationalized registration data to sort of frame out what it is that we'd like to look at.

And I think that Jay's pointed out at least a couple of the things we could mention – what are the things that we need to identify that could possibly be solved in the current WHOIS environment; what things do we acknowledge are not going to be solved by the current WHOIS protocol and service environment; and is there an indication, at least from some people here, should we experiment

with some protocols in conjunction with the IETF. So I think that those are the three formulations I would put forward as possible elements to this request to the Board.

Steve Sheng: Data model.

Dave Piscitello: And data model, yes.

Jay Daley: Yes if I could just add – thank you Steve. We need to be careful about this view that people aren't ready for this. I agree that for some of the things we're talking about people aren't ready for it, but for things like solving UTF8 or in order to help internationalization in WHOIS, either we do that very quickly or we say we can't do that very quickly. And I don't think that that can hang around; that one. And there are a couple of them that need to be done relatively quickly like that. You know what do I mean, I think they're solvable but that's very separate from the other one – standardization of the data model; adding on new features – that potentially is a very long discussion. I agree with you Dave, we need to start in parallel, but we mustn't let the potential scope of that one put us off the urgent bits that sit there at the front.

Steve Sheng: Michael.

Michael Young: So I think we're all aligned in what we'd like to do Dave. I mean we've had conversations about these sorts of things before and I

personally would love to see a new WHOIS server out there in a protocol that makes sense to everybody. I guess the question is again, we're just going around in circles maybe how to get it done. And it's an interesting – I guess when you're talking about getting multiple supporting organizations involved it makes sense to have it be a Board level directive because what else, that's the next layer over multiple supporting organizations, but I worry about it. I worry that – sorry, go ahead Liz.

Liz Gasster: I was just going to say that it's not necessary though. There's no reason why a sponsoring organization that feels that this is a priority, whether it be the GNSO, the SSAC, the ALAC, all of your supporting organizations – pick one, start there. Draft a charter. We can help you with the drafting of the charter. And the just circulate it. It can be done. I think there probably needs to be some caveat in that language about what ICANNs role in this is just to the inevitable questions about it, but that can be done in that route. And if that makes people feel more comfortable, we're happy to support it. It's the kind of input we want in terms of what the next steps should be.

Michael Young: Yeah. So the point I was going to wrap around to, whatever mechanism that gets applied to make this happen and I'm happy to try in help in any way I can in that regard, but I do worry that – when I've worked in companies in the past sometimes we push up ideas or concepts from the technology groups and those business people that think they're supposed to be putting the requirements together have failed to do so or failed to move or failed to

understand what the technology can do for them or can't do for them.

And there has just frankly been a little bit of railroading from the technology groups and we just get the job done. And the term that always gets bounced around at that point is a lot of anger and accusations like you've made a technocratic decision – that's the term I've heard many times in my life that really, really gets my goat. But I do worry that we need a strategy around the GNSO in particular because I've seen them get their back up over things that they feel might be railroading.

And maybe that's just socializing, talking with the Councilors, making sure that the work moves forward in a non threatening way. But in so much as it's difficult to get them all to agree to a mandate of work, maybe if a different supporting organization is officially the sponsor, we also don't want the GNSO to be perceived that they feel threatened and they need to stop it somehow. Because really, in their perspective on the contracted parties house and there in particular is going to be that this impacts their business model – this costs them money to do to deal with.

Dave Piscitello: Marika Koenig is indicating that there are still some seats left if people want to go and see President Clinton. I would like to thank everyone, because this was, I think a really, really worthwhile expansion. No comments – wait.

Michael Young: I do because I've made some annoying questions and asked difficult things and I just want to take a step back and offer that I thank Dave and Liz for your comments and your answers and I appreciate them. And I actually do like this forum. I think the fact that we're having a dialogue is a very good thing and I want you to know that I'm supportive and I just wanted that on the record because I've been annoying up till now.

Dave Piscitello: Okay, it's on the record, but just let me get this out. I will take as a responsibility to capture what I believe were the points that Jay made and that we think we are in agreement with to see if what we can do is then on the tech WHOIS list, formulate the kind of environment and way forward that it sounds like we came up with this afternoon. So I'll put it on the tech list. If you're not on the list, its tech-whois@icann.org – is that right?

Steve Sheng: If you're not on the list just talk to me. I'll just add you right now.

Dave Piscitello: Because it is open, we are transparent, we are legend.

[End of Transcript]