

---

KATHY SCHNITT: Welcome to the SSAC Evolution of the DNS Resolution Work Party Teleconference on Thursday, the 18<sup>th</sup> of May 2023. Barry, back over to you.

BARRY LEIBA: Okay. Well, I'm going to throw it back over to Andrew just to set everybody here. Thanks, everybody, for showing up. We have a good turnout this week. A couple of meetings ago, Geoff and I had a long discussion about my ideas, Geoff had about how to reposition the documents some different focal points, and he wrote them up. Section 3 has a lot of new texts. Geoff, there's been a lot of changes. And then Andrew has tweaked it. I'm going to let Andrew lead us through some main points that we need to resolve and then we'll go through the section just in general to see if that's the direction we want to take. So go ahead.

ANDREW MCCONACHIE: Okay. Thanks. I just posted the link to the doc into the chat. If people want to join, they're welcome to. As Barry said, we're just going to start at the top of Section 3. I merged all Geoff's text and I made a couple edits to it. The edits that were just grammatical and whatnot, I just accepted. Geoff, you made some more, so I accepted those as well. The edits that are remaining are the ones that I thought we should talk about where I was unclear how to deal with them, if they were grammatical or they're more substantial. Let us just start right at the top. Merike isn't in the call so maybe we want to skip this one.

---

*Note: The following is the output resulting from transcribing an audio file into a word/text document. Although the transcription is largely accurate, in some cases may be incomplete or inaccurate due to inaudible passages and grammatical corrections. It is posted as an aid to the original audio file, but should not be treated as an authoritative record.*

---

BARRY LEIBA: We can see if there's something that we can talk about it. It's possible she made that in the document because she knew she wasn't going to be on the call.

ANDREW MCCONACHIE: Okay. I think what she's pointing out here is that we say in the sentence the DNS was designed hierarchically for technical reasons. Merike is asking, "Well, what about underlying organizational reasons as well? What about social aspects?"

RUSS HOUSLEY: Well, then I think that aligns with the comment I make later, which is it wasn't just the restrictions of the computers of the day. It was the ability to manage the space that made it hierarchical. So I think we're saying the same thing, she just say that differently.

TARA WHALEN: But some of the scaling was organizational scaling.

RUSS HOUSLEY: Right. I mean, remember that when the host file was there, SRI had a person whose job was to edit it and then make it available for FTP to everybody else, and that was the absolute single point of control for every aspect of the name. That to the creation of delegation from that one person who edited the host.txt file was important.

---

GEOFF HUSTON: I certainly recall—I was pretty young at the time. I recall the introduction of hierarchies and there were a number of them. There was the X5 and 9 hierarchy, there was the dotted hierarchy. In essence, there wasn't, to my mind, a big emphasis on organizational differentiation. At the time, as you point out, Russ, it was all done by whoever was running the host.txt file, variously, DARPA and bits of some registration body deep there and then across to SRI. The organizational convenience came later. It was an after effect which was convenient, but it wasn't a motivational factor. As we migrated away from host.txt and started to look at 9 hierarchies, as I vaguely recall—I was pretty young—it was about this whole issue of scaling the search space so that, in essence, you didn't have to bring everything all at once to bear on the search problem. It just cut it down and made it feasible. I think the organizational hierarchy came after and it was a beneficial side effect. But it was a long time ago, my memory fails. What can I say?

RUSS HOUSLEY: Yeah, It was a long time ago. Right. But I think they were both motivators. That's really just a question. But I think whether we say anything about the administrative scale being improved by using a hierarchical namespace I think resolves both my comment and Merike's comment.

WARREN KUMARI: I must admit, I'm kind of confused. Because if anybody's tried to load the .com zone into our current machine, there are still memory and

---

processing constraints. There is no possible way that you can load all possible names into a modern machine, let alone some of the big zones are tricky to do. So there are still memory and processing constraints that limit how many names a machine can keep in memory and they're fairly much always will be. One still would have to have a structured hierarchy, and the obvious ways of doing the structured hierarchy is both along organizational/administrative balance because I can't think of any other way you would do it. I mean, I'm kind of confused about what it is exactly we're trying to solve here if we just dropped off the "There were memory and processing constraints in the 1980s" if we just said, "There are memory and processing constraints that limit blah, blah, blah," doesn't most of the problem go away?

GEOFF HUSTON:

Well, I was actually temporizing that assertion that flat namespaces don't scale, and you're bringing it back. Well, they've had scale today. I guess you're right. But my point is I don't sense the same resistance to flat namespaces these days in some of these alternate schemes because of things like distributed hash tables. Now, you could argue that distributed hash table is just an invisible hierarchy because the hash gives you division of the space and that gives you what you're after. But the point at the time that I recall—and I think is still true at that time—when we're going, "Well, how does this scale?" the hierarchy gave us a convenient answer that we couldn't do with a flat host.txt file. So that was part of the, "Well, this whole distributed database stuff makes sense, doesn't it?"

---

WARREN KUMARI: Yeah. But even with a distributed hash table, an important word is there, and there is a distributed hash table. You take chunks of the hash table but it's still true. It was believed at the time that flat namespaces do not scale as well as hierarchical namespaces. I think that that's still true. A distributed hash table is still a hierarchy, so we still believe that. It was believed. It kind of implies that then people back in those days didn't know this stuff and we are no more now. I mean, that's true. I wasn't alive, then.

BARRY LEIBA: My sense, Warren, is that thinking behind using a distributed hash table is less for scaling than it is for distribution of administration and authority, that it avoids having a central point of authority. Well, it also does distribute load in some way. I don't think that's the intent of it.

WARREN KUMARI: I mean, I think it is because otherwise you would just have a single data structure, like big file or whatever, massive protocol buffer, and everybody would just have their own copy of it. You're doing the distributed hash table file instead of distributed file that you have some sort of like, "Bob has this chunk of the space and Fred has this chunk of this space and Mary has that chunk and Susan has these two chunks together." I still think that most people believe that flat namespaces don't scale as well as hierarchical namespaces. Although I guess maybe some of it is we're using the word namespaces here, which is not the same as flat data structures. I guess some of the new folks think that a flat namespace, if it's just a set of public keys does scale as well as

---

hierarchy of names because I think we're collapsing the concept of namespace and the structure that the namespace is encoded in. Like the namespace `www.example.com` is encoded in the data structure that has `.com` and then `example`, and then `www`. Whereas a namespace that is just somebody's public key, the name is distributed but it's not that the namespace has hierarchy, it's that the data structure has hierarchy. But I think we might be like, once again, disappearing down the rabbit hole of what's technically correct versus what are people going to take from this.

BARRY LEIBA:

I want to make sure that we're not giving a history lesson. But that when we talk about history, the purpose of it is to clarify where the document is headed.

WARREN KUMARI:

It feels to me it was believed sentence could go and there are memory and processing constraints that limit how many names a machine can keep in memory could be changed.

Personally, this might just be me—I think that `www.usc.edu` and `www.toyota.com` are easier to remember than just having a whole big collection of names. I think that for human purposes, having some sort of hierarchy is also helpful. Like `google.ie` is likely Irish version of Google, whereas `google.de` is likely to be the German one. And I think that that's easier for me to remember than it would be Germany-Google and Ireland-Google. But maybe I'm wrong. Actually, I don't know. Maybe that's how it's always been done so that's what I think.

---

BARRY LEIBA:                    Anyone want to follow up on that?

GEOFF HUSTON:                Well, I'll kind of defend the text in here, insofar as in trying to tell a story about the evolution of the namespace, it made sense to me to paint a picture of the birth of the DNS out of host.txt, and talk about, in essence, the reason why host.txt, even if you changed the way it was distributed from FTP to some kind of distributed lookup, if the namespace was flat, that was a significant challenge and it was a challenge insofar as discovery algorithms, etc., the whole problem was unbounded. The hierarchy made whole bunch of sense. It was a scaling problem. It scaled across a large number of names. In essence, there was no limitation at that point once you introduced a name. "I can only have 10,000 names because I've only got so much memory in my computer, blah, blah, blah." It doesn't matter anymore in the hierarchy. So it's trying to tell a story about then, not today. And it was trying to tell about how we got into a hierarchy and the natural outcome of the use of that hierarchy in a distributed databases method insofar as the hierarchy then said, "I've got to discover where to ask and then ask discovery and resolution."

Those first paragraphs were intended to flow together to talk about this idea that this hierarchical design facilitated the adoption of the distributed database, and we arrived at the DNS as we know it. Now, if there's some other way to paint that same story of the sequence of

---

thoughts, that's fine. This is one effort to simply brush over that 15 years of work in a couple of paragraphs.

ANDREW MCCONACHIE: Is it fair to say that the memory and processing constraints in the 1980s that limited how many names a machine could keep in memory at one time have not been entirely alleviated but had been partly alleviated?

WARREN KUMARI: Yes and no. When we had seven domain names, you could make them fit in a machine which had a tiny bit of memory. We now have orders of magnitude more memory, but we've just got orders of magnitude more names. So I could easily serve all of the names from 1989 on a raspberry pie but I could not serve all possible names on even the biggest pot-like machine I could buy now. Of course, I just like making—

ANDREW MCCONACHIE: But wouldn't it be fair to say that Moore's Law has outpaced growth in domain names?

WARREN KUMARI: Pardon?

ANDREW MCCONACHIE: Would it be fair to say that Moore's Law has outpaced growth in domain names?

---

WARREN KUMARI: I don't think it has. I don't think one can buy a machine that could serve every single name and every single domain.

GEOFF HUSTON: Look, it's possible these days—and we're talking about evolutionary pressure. It's possible these days to remove the external dependencies that are intrinsic to a hierarchy, A.B.C says C is dependent on... B is dependent on A. Remove those external dependencies and contemplate names that do not exist with that implicit dependency in its very structure. These days, we can think about that and there are namespaces out there that are flat as a result.

Now, I don't think that thought was tenable in the '80s. It didn't work. I think what Warren is saying is if you flatten the space and use the existing DNS style technology, it ain't going to work, and that's true. But that's not really what's being said here. What's being said is as you get more capable machines, you can use and bring in other computing techniques to give you effectively namespaces that have unlimited or even remove the number of intrinsic external dependencies down to just one, the space itself. There's no parent that you're reliant on with your own name integrity, etc. So it makes, I think, some sense to introduce the hierarchy as the structure of the time. And then a few paragraphs later, talk about the fact that we do have more capable machinery, we do have greater capability, and that gives you the ability to think about other thoughts that aren't necessarily, "Well, that's just impossible."

BARRY LEIBA: I guess another factor here is that we could not have imagined back then how many devices in the world would be making queries out of this system.

GEOFF HUSTON: God, we had no clue. We thought 100,000 was a big number.

BARRY LEIBA: Right. Actually, ironically, a flat host file was more suited to that in any kind of actual query system.

GEOFF HUSTON: You know, host.txt, it was there for a reason. It was just lightweight, cheap, and it worked for the scale that we were looking at. I see your edit there, Andrew, which reflects the last 23 minutes of conversation. But I must admit I had strayed away from making such generic assertions that immediately I think it's easy to go, "Yeah, really?" That was where I kind of temporize that with, "Well, it was believed at the time that this was the case." And left aside the dangling question, "Well, we can believe it now too or not," and just not try and cover that space. Warren is making the point, "Well, I think we believe it now." I'm not convinced, to tell you the truth. The whole issue around a universe of names with some kind of invisible implicit internal structure to aid search and discovery is not an outlandish thought these days.

---

WARREN KUMARI: It will still be Internet naming, though, or Internet name resolution?

GEOFF HUSTON: The whole idea of labeling things, Warren, is you need some kind of resolution function, whether you call it Internet or not. It's intrinsic.

WARREN KUMARI: I think I'm somewhat confused. Zooko's triangle, it feels like it has some applicability here. While you could have some non-hierarchical namespace, without the ability to carve off sections that certain sets of organizations are responsible for, I don't see how you end up with a coherent usable space. Foo.microsoft.com and bar.microsoft.com, I have some implicit trust because they are clearly under the organization of Microsoft. If you did away at the top level, it was just foo.microsoft and bar.microsoft, I would have the same sort of trust model. If you completely did away with any sort of hierarchy, I don't have a way of grouping together things. And I don't think you have names—not quite become meaningless. So they lose any sort of useful semantic human properties.

BARRY LEIBA: I agree with that. But I'm not sure to what extent. I'm not sure where the cause and where the effect is. For instance, we have protocols now that we—let's just look at things like Dcamp and SPF and stuff like that that are used for e-mail stuff. They rely on the hierarchy because that's how policies are specified. But I think that the fact that we had this

---

hierarchy enabled that rather than that we put the hierarchy so that we could do that.

WARREN KUMARI:

Yes, I agree with that. But I think that it is now so useful and entrenched, moving away from that would be a mistake. For example, my bed in my house has a meaning and there's a hierarchy there. There's, I guess, my bed and my bedroom and my house. I have my house, it is further subdivided into my bedroom, and then further subdivided into my bed. And there's an implicit sort of trust model in there. If you go along and climb in my bed, I would be surprised. There is a useful thing in that so even if one were to do away with names having an explicit hierarchy/cut point, you still would require some manner of grouping things. Clearly, I've been thinking about things in this manner. I guess it could always be instead of there being google.ie and google.de, there could just be—the name is Google and then /ie or /de. Instead of there being foo.bar.com, there's just bar/foo, and you rebuild hierarchy but the hierarchy is not built in the namespace layer, it is built in some other layer.

GEOFF HUSTON:

I can't help observing that we've spent the next 10 minutes arguing about a proposition that I think was true. At that time, it was believed that namespaces don't scale as well. The proposition is to remove at that time and make it a more generic observation about then, now and in the future flat namespaces. And we're having trouble with that assertion. And I think we're having trouble because it is a troublesome

---

assertion. It's not, except that even in this call, that's the case. If we draw back into what is a clear consensus, back then we thought that. That doesn't mean we do or don't think of now. But certainly back then we thought that and I would assert that that was true then without making any judgment now.

WARREN KUMARI:

Sure. But saying "People used to believe that putting leeches on wounds was a good solution" does come with the implicit and we no longer believe that, right? People used to believe that the world was flat. It carries an unpleasant thing that we have moved on from that. But also you say we have people on the call who believe that flat namespaces do scale as well as hierarchical ones?

GEOFF HUSTON:

This is a troublesome proposition with me in the context of today and now. If I was going to accept that as going, "Yes, let's say that," I would actually go, "Well, where are the references? Where is some pointers to related work that has gone deeply into this?" I think talking in more abstract notions that we have doesn't necessarily make it so. It's more than an opinion rather than I suppose what I would expect in this document are justified conclusion, getting to the point of even the truism.

WARREN KUMARI:

If we just remove that sentence, would it make our problem go away?

---

GEOFF HUSTON: Well, I guess it would, if that's your problem.

GEOFF HUSTON: In the previous sentences, the hierarchy primarily so it could scale, right? And if you left it clear then that was true then. It's true about then. We might as well put in a sentence about saying why we went that way in the plethora of choices at the time, but if you want to remove it, if it's troublesome to you, that's okay, too. But like I said, I wasn't trying to make a more generic assertion about flattened hierarchy that extended from then until now. I was not.

ANDREW MCCONACHIE: Is everyone okay with removing the sentence entirely?

GEOFF HUSTON: Well, we've got to have some outcome for the last 31 minutes.

ANDREW MCCONACHIE: Right. Is anyone not okay with removing the sentence entirely? Speak now. Sentence going away. Why am I not allowed to delete it? Okay. Good stuff.

In the previous sentence kind of led into this, Geoff, you'd suggested changing the word computers to names. I think we can say names there but we need to say support a large number of names, as opposed to saying scale across a large number of names.

---

GEOFF HUSTON: That's grammatically ... It doesn't bother me either way. It's basically how do you cope with a large number of names?

ANDREW MCCONACHIE: Okay. Then I'll just revert this.

GEOFF HUSTON: No, no. It's not the computers. That was the whole idea. It wasn't the computers that were the issue. It's the size of the namespace.

ANDREW MCCONACHIE: Okay.

GEOFF HUSTON: So, yeah, support scale, I don't care. It's names, not computers.

WARREN KUMARI: Yes, I think this is good.

ANDREW MCCONACHIE: All right. Then I guess we come back to this.

WARREN KUMARI: I mean, that's still a true statement, right? There are still memory and processing constraints that limit how many names that machine can keep in memory.

---

GEOFF HUSTON: That may be so, Warren. But it's not making a judgment on today. And it's neither negative nor positive. It's just saying what was there then? That's all.

ANDREW MCCONACHIE: We can add a footnote here kind of stating that. Because I hear what Warren is saying where he's concerned that there's this assumption and when you read that sentence that this is no longer true.

WARREN KUMARI: In the 1500s people believed that the world was flat.

GEOFF HUSTON: Well, when [inaudible] through the others, one of them calculated the radius of the earth.

WARREN KUMARI: I knew you were going to call me on that.

GEOFF HUSTON: But way B.C. someone believed the earth was round. That doesn't contradict a belief today, or not. It just doesn't.

ANDREW MCCONACHIE: Do you want to footnote here saying—

---

GEOFF HUSTON:                   The earth is round?

WARREN KUMARI:                I used to think that Geoff Houston was an idiot.

GEOFF HUSTON:                It still is today, Warren.

WARREN KUMARI:                It does carry an implication, but I don't really care. This is not a headline I'm willing to die on.

ANDREW MCCONACHIE:         Okay, cool. So we leave it as is.

GEOFF HUSTON:                Well, yes. It's my view.

ANDREW MCCONACHIE:         Perfect. I believe that, Warren, you as well. So then we come back to kind of the original question. Because we've been talking about technical limitations and whatnot, and both Russ and [Mary Kay] wanted to address the issue that there might have been more than just these technical limitations that led to the design of the DNS.

---

WARREN KUMARI: I mean, an obvious thing that we could do—I'm not actually suggesting we do it, but what we could do is somebody could poke—what's his face?—Mockpetris and ask him. Or we could read that like "DNS, the early RFCs."

ANDREW MCCONACHIE: For the Design of the Domain Name System, I believe, that paper by Paul Mockpetris?

WARREN KUMARI: Yeah. Domain Name Concepts and Facilities. There's a whole thing in there on the history. Proposals varied, blah, blah, blah. Based on the experience of several implementations, the system evolved into the scheme described in this memo. So like IEN 116, RFC 7999 from—

ANDREW MCCONACHIE: Okay. So do you want me to do that research and we can kind of move on?

GEOFF HUSTON: Well, again, I don't think it makes much sense in the context of this document. What those opening paragraphs are trying to say is this is a design of its time. The evolutionary pressures have existed today. Now, they're around today because things have changed. The constraint factors in the design trade-offs to the DNS at that time now have

---

somewhat different outcomes where [inaudible] to relook at it. Some of the evolutionary pressures exist because folks are doing that hypothetical exercise and are looking at the DNS through a different lens of current capability. And I say we can do X. And the issue comes if you sort of see that as being, “Well, let’s try X as an evolutionary pressure,” then that’s as far as this document at this point is trying to say. These pressures exist. It’s not trying to make a value judgment. They’re all crazy people. It has no hope in hell. It’s just trying to say the constraints that were made in the design trade-offs of the day are not necessarily still the overbearing design trade-offs that exist in this world today, right now. There are pressures to actually relook at those design decisions, and they come out, they surface as evolutionary pressures without making a value judgment as to whether better or worse or anything else.

So that’s all it’s what it’s trying to do, because the section is actually titled “The Motivations to Evolve Name Resolution.” It’s not about what they are. It’s really trying to explain that there are, if you will, legitimate causes for folks to explore those trade-offs and think about different points of trade-off in that space. So it wasn’t, if you will, that deep and it wasn’t trying to value judge. It wasn’t trying to justify history, it simply tried to say those motivations exist because the world has changed now, that those design trade-offs do not necessarily apply. And some folk think they can make a different set of trade-offs. Yay.

WARREN KUMARI:

I agree with you that yay, some people think they can make a different set of trade-offs. However, having a quick look at some of the

---

documents, it looks as though our first couple of sentences in Section 3 are actually completely wrong. The names that were in use prior to DNS were not hierarchically organized. I don't think that's actually true. Looking at, for example, RFC 799 and 819 and IEN 116, these were all pre DNS documents. And names used to be of the form, user.host@domain. That was before we had the DNS. We also had user@host which was before there was the DNS. And user@host is a hierarchical namespace. Geoff@server3 is a hierarchy that has a sort of branching point at server three.

So the design of naming was hierarchical before the DNS, it was based on organization and/or machine and/or entity that you worked at and/or, in some cases, network that you are on. Like if you were on Network A or Network B, and you needed to go through some sort of gateway. So, naming grew out of the sort of natural organizational hierarchy that people fall into, not based on the technology. Even when you had Unix [bang] pods or [UCP bang] pods, you'd be ... This thing, then this thing, then John or Geoff. So I think hierarchy, A, it's not true that names were not hierarchical until the DNS came along. And B, even without looking at that sort of before computers, you ended up in a fairly clear, hierarchical structure, as in you, Geoff, are the Geoff within the subset of Houstons which are within the set of Houstons who are within the country of Australia. And when we bump into people who are like, "Hey, do you know Geoff?" and they're like, "Which Geoff?" and you're like, "Oh, the Geoff whose last name is Houston and works for APNIC and is in Australia." So naming always has some sort of concept. Probably my saying "always" there is going to get me beaten. But especially names used for computers have had hierarchy built in. The

---

names are used and prior to the DNS are not hierarchically organized. That's simply not true. I'll need to go look at Roy's presentation. But there's clear evidence from RFCs that I pointed at, IEN 116, blah, blah, blah, 83 show that that statement is true. I think it's a fairly—

GEOFF HUSTON: I spent my time in the late '70s on a number of machines and a number of emerging networks. The names that we used inside the various incarnations of host.text were unstructured, flat names.

WARREN KUMARI: Yes. And host.text, I will happily copy that.

GEOFF HUSTON: This text reflects that experience. Now, at this point, I'm sort of sitting here going well, "I've tried to tell a story. The call doesn't seem to like the story. I give up. There's not much more I can do here. I have tried to say it from my perspective. If someone else wants to say it from a different perspective, yay, go for it. I give up."

WARREN KUMARI: Well, no. Hang on a minute. It sounds like you're taking a ball and going home. That's not what we're aiming for.

GEOFF HUSTON: I feel kind of pushed insofar as this wasn't the point. It was just trying to lay down a sequence of observations that lead into the motivations for

---

evolution are genuine, and that they exist because of an evolution of circumstances and environment and trying to color that assertion. If you've got a different way of trying to make that assertion that color, Warren, go for it. I tried and that's the best I can do.

WARREN KUMARI:

I like your way. I think that what we are stating there, the names that we use in that, we're not hierarchically organized. I think that that statement is not true as written. So people reading it could, quite rightly I think, figure that we quite don't know what we're talking about. But I think the statement needs to be maybe before domain names, machines on the Internet were named in a flat .text file with no hierarchy.

ANDREW MCCONACHIE:

Is there some distinction in the past which no longer exists between host names and domain names? And the examples that we're seeing here with Mills.USC-ISIE@ARPA, I mean, I don't know if USC-ISIE is the same kind of thing as ARPA there.

GEOFF HUSTON:

There was an evolution of naming discussions in the 1970s, and it was all around the X. It X400 or X500 naming system. One crowd of folks sort of sitting inside the CCITT space wanted to use names as relating to the real world—organization units, company names, geography names, blah, blah, blah. So the names were qualified with metadata, with the metadata had an explicit OU equals X, right?

---

There was another approach evolving from the issue that computers were servers, servers had names. Full stop. And the original incarnations of [inaudible] are a good example, where it was just a flat space. Pick a name. You have I think the six characters, or maybe eight. Just pick a name. And if you clash, oops, that was a bad idea. And you're meant to figure it out by yourself. The bigger the network, the harder things got.

The evolution into the dotted hierarchy was fought, was contentious, was not obvious, and it was, by and large, an Americanism and wasn't shared by others for some time. Because even things like [UCP], we didn't necessarily do the whole—this is the context of that name, A.B.C.D. It was you get to A by going to be B, B being A. You actually described how to get there, not what it was.

So the whole issue of having a separate domain pods that gave you the context that the server pod I assert was actually an outcome of this effort to try and make DNS get bigger and simplified and deliberately not go down the X500 path. But we're talking a little history here, Andrew, very, very old history. Most of the folks involved in that debate have died. It's a long time here.

WARREN KUMARI:

So I just put in text, which I think says the same thing as that just ... We're in a flat text. I mean, it still feels like there was hierarchy. Maybe I'm just being tied up in technical exactness. But I think person@machine is still a naming hierarchy.

---

BARRY LEIBA:

So let me try to pull this aside. Because repeating what I said earlier that we need to keep the history lesson here to the point that it's supporting the document rather than making it complete. And so maybe what we want to say is that the naming hierarchy at the time was in its infancy, that there certainly were names that were hierarchical, there were names that were not hierarchical, there were different uses of different kinds of names with different scopes. And maybe where we want to talk about it that way, that the hierarchy that was built, it wasn't there at the beginning, and it didn't just sort of drop in. It was developed over time with experiments of different kinds of hierarchies. So is that a reasonably accurate thing to say that gets the point that we want to make across?

WARREN KUMARI:

That works for me. I had just viewed the statement as it was worded as being technically incorrect and I thought we could make it technically correct without it being a big thing. I did not intend to annoy Geoff. Often I annoy Geoff but this time I didn't. I just try to figure out a way that we could say it that—yeah. The names that are used prior to DNS were simply a list of names and a txt file or something like that. But the original works for me, Andrew, as does what you said.

GEOFF HUSTON:

I'll say this a different way and I'll say it only once. Prior to the DNS, the positioning metadata, which was an attribute of the terminal name that [isn't] labeled, was explicitly called out as metadata. Folding that metadata into the name itself and using the period character as the

---

label delimiter between meta and endpoint was a DNS invention as far as I am aware.

WARREN KUMARI: Okay. Sure.

GEOFF HUSTON: That was the insight for the DNS. And it was a brilliant piece of insight, right? Because everyone else was fiddling around with metadata, and it just wasn't cleanly attached to the label. That was probably in my head when I wrote these perhaps simplistic statements about the hierarchy was a brilliant cut through because it made it scale easily. At the time, we were searching for that. We really were. Any other way of doing this didn't work as far as we were aware. I suppose where I was moving in the section is that thought, "Any other way doesn't work," is now open to some fundamental question. That's quite legitimate. It is open to question. Some of the motivations to evolve the name resolution process trigger around evolution of the space of names because we can think bigger thoughts these days, because we've got more capable machinery, blah, blah, blah. That's all. Yay.

ANDREW MCCONACHIE: Are we happy with the text as it now stands? Should I revert this?

GEOFF HUSTON: I'm actually not sure. I could point to a naming space in the late '70s, early '80s that had what I would call inline metadata in the way that we

---

kind of think about the DNS now with its dotted hierarchy. I cannot think of an example. The networks were never very big. You know, even the Fidonet relay networks that were emerging at the time, it was more the bang UUCP style. You get to hear from there, by there, by there, and they're all just like server names.

I can't in my head recall any kind of inline hierarchy whether it was dots or hats or, you know, whatever.

UNIDENTIFIED MALE: [inaudible]

WARREN KUMARI: mills.comsat@ARPA as being a hierarchy of the ARPA network and then the machine code comsat and then the user account on that called Mills. Right. It's the first, and yes, it's not inline metadata, but it is a switch of the hierarchy of, you know, ARPAnet and other networks, you know, ARPAnet and Chaosnet. And that's your first layer of hierarchy. And then the machine. And so I was viewing that as hierarchy and it's not encoded all in a single label system, but it's also like, this is probably one of those things that doesn't matter to almost anybody who's going to read the document. So I'm putting it back to how it was before I opened this can of worms.

BARRY LEIBA: That was my goal in suggesting the text that Andrew summarized here and what just got deleted. But the part where the names were not in a consistent hierarchy that we have today and we develop this consistent

---

hierarchy was because I wanted to keep it to the point that that moves this document forward rather than trying to be complete in the history.

GEOFF HUSTON: Well, I'm happy with Andrew's text [inaudible] organized in different manners. I didn't really want to make the point that it wasn't hierarchically organized, it was more ... Yeah, fine.

BARRY LEIBA: Okay. I'm good with the text that's on the screen now. Warren, is that fine with you?

WARREN KUMARI: Yep, that works for me.

BARRY LEIBA: Okay.

WARREN KUMARI: I mean, the flow into however, I don't know if we need the word however, still flows into the DNS, but purely editorial or whatever.

BARRY LEIBA: I agree. However is not necessary. So we have now spent the whole hour on one paragraph, talking about scaling issues. This does not scale to getting this document done. So we need to wrap the call up now

---

because we're almost at the top of the hour, but let's... Warren, can you find—

WARREN KUMARI: I won't be on next week's call, so that'll help.

BARRY LEIBA: Well, right. There is no call next week. So we meet again in two weeks. In that time, Warren, can you break out some time to go through this section and make comments in the Google doc so that we have stuff that we can bat around in between the call and on the call in two weeks?

WARREN KUMARI: Yes, I will definitely. I think I can do that. Yes, I will definitely.

BARRY LEIBA: Okay. Thank you. And does anybody else want to make any final comments before we wrap the call up? And yeah, Rod in the chat says, perfect is the enemy of everything. Yes, I can say as we develop this, keep in mind that there's a point here and we want to... This text needs to support the direction of the document, not be absolutely complete in every detail. So keep that in mind.

ROD RASMUSSEN: And we do need to be done in the summer.

BARRY LEIBA:

Right. So I'll repeat one thing that I said last week that most people weren't on the call for, that we are on the schedule for the... Well, we're tentatively at least on the schedule for the SSAC retreat, the SSAC workshop to have this document presented to SSAC to review before that, and then to discuss it in the workshop.

So the idea is that the document does not need to be complete then, but it needs to be complete enough that SSAC as a whole can give it a read through and we can have a meaningful discussion.

So that means basically mid to late August, we need to have something pretty solid. Seems like a lot of time, but not at the rate of one paragraph a week. Thanks, everybody, for coming. And thanks for the discussion. See you in two weeks.

**[END OF TRANSCRIPTION]**