ICANN Transcription

IDNs EPDP

Thursday, 14 October 2021 at 13:00 UTC

Note: Although the transcription is largely accurate, in some cases it is incomplete or inaccurate due to inaudible passages or transcription errors. It is posted as an aid to understanding the proceedings at the meeting but should not be treated as an authoritative record.
Attendance and recordings of the call are posted on agenda wiki page:
https://community.icann.org/x/rgBmCg
The recordings and transcriptions of the calls are posted on the GNSO Master Calendar page
http://gnso.icann.org/en/group-activities/calendar

TERRI AGNEW: Good morning, good afternoon, and good evening and welcome to the IDNs EPDP call, taking place on Thursday, the 14th of October, 2021 at 13:00 UTC. In the interest of time, there will be no roll call. Attendance will be taken by the Zoom Room. If you're only on the telephone, could you please identify yourselves now? Hearing no one, we have no listed apologies.

All members and participants will be promoted to panelist for today's call. Members and participants, when using chat, please select "panelists and attendees" or "everyone," depending on your Zoom update, in order for all to see the chat. Observers will have view only to the chat.

Statements of interest must be kept up-to-date. If anyone has any updates to share, please raise your hand or speak up now.

DONNA AUSTIN: Terri, I probably should say that I will be updating my SOI to reflect that I've become chair of this working group.

TERRI AGNEW: Thank you, Donna. Noted. Seeing or hearing no one further, if you do need assistance, please email the GNSO secretariat. All documentation and information can be found on the wiki space. Recordings will be posted on the public wiki space shortly after the end of the call.

Please remember to state your name before speaking. As a reminder, those who take part in ICANN multistakeholder process are to comply with the Expected Standards of Behavior. With this, I'll turn it back over to our chair, Donna Austin. Please begin.

DONNA AUSTIN: Thanks very much, Terri. And welcome, everybody, to the call today. We will primarily be doing a continuation of—and hopefully, finish of—Sarmad's presentation on the Root Zone LGR, including the demonstration that folks have asked for. So that will be the main focus of today's call.

Just in terms of chair updates, there is a call for volunteers for vice-chair. We do have one interested party, Anil, so thank you very much for your interest, Anil. That call for volunteers is open until the end of Friday. And if we need to run an election by Doodle, we will do that in the early part of next week.

So with the introduction to working documents that we will talk about, Emily is going to run us through some documents that will

help capture our discussions and guide where we are in the process. So Emily will go through that with us.

And I also want to talk to folks about ICANN 72. It looks like, based on the Doodle poll, that we could have a working group meeting during ICANN 72. I just want to confirm with folks that that is okay with them.

So with that—and I don't see any hands so I expect no questions—I will hand over to Sarmad who will finish off the presentation he started last week and also the demo. So, Sarmad, over to you.

SARMAD HUSSAIN: Thank you, Donna, and hello, everyone. We'll actually do this in two parts today. I'll try to finish up in about 10 minutes or so—the rest of the presentation, which we started looking at last time. And then I'll request my colleague, Pitinan, to take us through a demo. So let's get started.

We were talking about this particular slide last time towards the end of the call. So I was just starting from here so that we just remind ourselves that when a label is submitted, we get the original label. But along with the original label, we also get potentially allocatable labels and blocked labels, which are possibly generated through the tool.

And as is the design of the Root Zone LGR, we want to minimize allocatable labels and maximize the blocked labels. Maximizing blocked labels allows that any security—potential issues are

catered to through blocking of those labels because then those particular labels can potentially not be allocated to someone else.

Okay. So moving on, as far as the process of evaluating a TLD label is concerned, a gTLD label is submitted to the LGR tool as a gTLD or a variant gTLD. So there could potentially be two modes of submitting these. So again, before I actually go into detail, I just want to share that this is only a suggested possible process. Eventually, this working group will really be the one which will determine the way this really should be done.

All gTLD applications, ASCII and IDNs, would follow the same process. A tool evaluates the label, using the Root Zone LGR to determine whether it's a valid label or not. And then the tool determines if the label is available. So a label will be available if the label is not already delegated, of course, and it is not reserved, but also possibly that if a label is a variant of another label with is already delegated, or also possibly if the label is a variant of a label which is reserved. Even then, those can potentially block the label from moving forward. If the label is available, the applicant will be informed.

There is a discussion to be had. Even though today we will be demoing you a tool which can generate all the possible allocatable and blocked variant labels, that may not be the best way to design the tool because there can be thousands of potential blocked labels which can be generated and that can really slow the process down. So there are alternate engineering or architectural solutions suggested for this tool to make it better. So at this time, I think even though allocatable and blocked labels can theoretically

be calculated, in practice, how the tool will eventually function may be based on further discussions.

So if an applied-for label is not intended to be an original gTLD label but actually a variant gTLD label of an already-delegated gTLD, then the assessment will be slightly different. Then, the assessment would be that after checking whether the label is valid or not, it will check whether it is actually an allocatable variant of the specified gTLD, and then potentially, that the applicant for this variant label is the same entity as that of the delegated gTLD label.

If the label is cleared through Root Zone LGR, then the application proceeds for further steps, like the DNS stability review and then the string similarity review. But if the label is not cleared to proceed, then obviously, the working group would need to decide what will be the mechanism for such cases.

Other details which are relevant is that there are multiple stages or statuses which a particular label can have. A label can be withheld for the same entity, which is actually an allocatable variant but often already a delegated TLD. So it is actually reserved for the same entity. And potentially, on a later request, reallocated. You could actually have a blocked label but if there is a change in Root Zone LGR, it can potentially become allocatable and therefore, potentially, withheld for same entity.

And an allocated label, of course, can be delegated. A delegated label can potentially be allocated. That's not likely but there may be some rare cases possible. And then a rejected can also be

withheld for same entity in case there are some changes in the Root Zone LGR.

DONNA AUSTIN: Sarmad, sorry. Justine has her hand up.

SARMAD HUSSAIN: Okay. Sorry. Please.

DONNA AUSTIN: Justine, go ahead.

JUSTINE CHEW: Thank you, Donna. Could you go back one slide, please? I had a question regarding the slide before. Yes. So in relation to the second main bullet point you have on slide number 27, can I just ask, is it your opinion, Sarmad, that the validation process using Root Zone Label Generation Rules—that process is actually outside of the DNS stability review that takes place as part of the [new] evaluation?

SARMAD HUSSAIN: The Root Zone LGR evaluation is certainly a part of the DNS stability review but it may not be a complete DNS stability review. It is up to the working group to discuss, and of course, decide what additional aspects may actually be looked at. The current bullet point is based on, of course, the existing process from the

2012 round, where the string went through two reviews—the DNS stability review and the string similarity review.

So I'm not suggesting that but I guess that is something which the working group will need to discuss and decide, whether there is a human or manual inspection needed for strings clearing the Root Zone LGR. The Root Zone LGR certainly may not address all the different aspects of DNS stability review.

JUSTINE CHEW: I see. So if I may ask a supplementary question, Donna.

DONNA AUSTIN: Yes, you may, Justine.

JUSTINE CHEW: Okay. Great. Thank you. So one of the questions that the ALAC team posed to Sarmad, I guess, is we were trying to understand how the introduction of the Root Zone Label Generation validation process was going to effect the DNS stability review. So based on your answer, Sarmad, my understanding is there will be consequences in terms of change to DNS stability review process or evaluation but we do not know to what extent. And you're saying that it's up this particular EPDP to determine that. Is that right?

SARMAD HUSSAIN: Actually, this and SubPro was well because, of course, SubPro deals with it so there's probably going to be some discussions

within SubPro. So yes. This says eventually, the community needs to guide us on how to integrate Root Zone LGR into the larger DNS stability review process.

JUSTINE CHEW: Okay. Thank you.

DONNA AUSTIN: Thanks, Justine. We've got a couple more hands here, Sarmad. So, Hadia?

HADIA ELMINIAWI: Thank you, Donna. Thank you, Sarmad for this valuable presentation. Just to follow up on Justine's question. Definitely, the DNS stability review will include other elements, other than the Root Zone Label Generation Rules. And my understanding that for the Root Zone Label Generation Rules, this goes through an algorithm. And once it passes the review and it is deemed valid from that aspect, then it has to go through a DNS stability review in order to check the other elements that would say that it was fine.

My question, maybe it's not related that much to you. But what are the other elements related to the DNS stability that require to be checked? And what are the means of doing that? Thank you.

SARMAD HUSSAIN: I, unfortunately, am not able to answer that right away. I think there is need for a discussion with the technical community to

encapsulate a complete process and then the role of Root Zone LGR in that process. So again, sorry, but I'm not able to answer that question completely at this time.

DONNA AUSTIN: Hadia, what I'm going to ask Ariel or Steve to do, if we could take your question and get a response. And we'll put that on the e-mail list. I think the question is what happens in the DNS stability review and is there any …? What's the relationship with the Root Zone LGR? I think Ariel or Steve, if you could take that, or Emily. Just take that question. We'll see if we can get an answer to that offline. Jeff, go ahead.

JEFF NEUMAN: Yeah. Thanks. And on that question, I think we need to be careful because SubPro did go over the DNS stability review. I have to go back to the recommendations. I remember some discussions where it was assumed that, for the purpose of variants, that this type of review would be included in the DNS stability review as one element, I guess. But I want to be careful that we do not start delving into other areas of the DNS stability review because, again, that was SubPro. They did that. They recommended no changes. So we just need to be careful.

Separately, one thing I realized as Sarmad was going through this—and I put it into the chat—but if an existing TLD is an applicant for a TLD that is a variant … Sorry. If an applicant applies for a variant of an existing TLD and the applicant is the same as the registry operator, then it would not go through a

string similarity review because if we determine that the only party that's entitled to get a variant is the same entity, then it would be a vicious circle, where you would put it through a string similarity review, it would come up as similar to the existing TLD. But of course, we knew that because that's why it passed the LGR validation process.

So again, I think that's something we need to think about putting into the guidebook, if you will. Thanks.

DONNA AUSTIN: Thanks, Jeff. I have a follow-up question. If the TLD that was applied for is a variant but has no relationship and was applied for in its own right, not knowing it was a variant, if it goes through the Root Zone LGR validation process, then it would be picked up as a variant. And if it's not the same entity … Well, yeah. It will depend on how we define same entity, assuming that same entity is where we head. But if it's not the same entity, then it goes through the Root Zone LGR process. And because it's a variant, the string can't proceed.

JEFF NEUMAN: If we took this process that Sarmad has proposed, which says that if a label is applied as a variant of a delegated TLD, the tool uses a modified definition of availability. So therefore, the applicant would fail that second sub-bullet because it's not the same entity. So then it wouldn't have to go through a string similarity review because it's already failed.

DONNA AUSTIN: Right. So not withstanding that we have policy questions around that but okay.

JEFF NEUMAN: Yeah.

DONNA AUSTIN: Yeah. Okay. Michael?

MICHAEL BAULAND: Thanks. I'm not sure about Jeff's comment. Of course, we don't need to check whether a variant is similar to the already-delegated TLD. Most likely, it will be. But what about if the variant is similar to another already-delegated TLD. So we have the case where we have delegated TLDs A and B and they have obviously not been considered to be similar. But then there's a variant of A, say C, and that's also considered similar to B. In that case, probably C will not be allowed to be allocated, even though it's a valid variant of A, right?

JEFF NEUMAN: Yeah. I guess I hadn't thought about that. I don't know how that would happen but I guess, in theory, it could. So yeah. Maybe that's true. It would only go for string similarity for that purpose but it wouldn't go through for the purpose of seeing whether it was similar to other applied-for strings.

DONNA AUSTIN: Okay. We're highlighting that this isn't necessarily easy and there are a number of things that we need to think about. So I think if we're good for Sarmad to move on, we'll move on. And thanks, Michael. Jeff, your hand is still up.

SARMAD HUSSAIN: Okay. Thank you, everyone, for a great discussion. Let's move on. Another aspect of variants and delegation of variant TLDs is the number of variant TLDs which are delegated. Even though the Root Zone LGR is designed to minimize allocatable variant labels, there is still a possibility, as we've seen, that some strings may generate multiple variant labels and they can actually be, in rare cases, numerous.

So SSAC, in SAC060 has advice that a large number of variant strings can present challenges for management of variant domains at registry, registrar, and registrant levels. So it recommends that the number of activated variants is as small as possible.

So then there is obviously some discussion which needs to be done by the working group to see if this is an advice which, indeed, needs to be taken on board how to address it. So even though the integration panel and generation panels have to reduce allocatable variants, there can still be many. And therefore, some policy would need to then be decided to contain the actual allocatable variants which are allocated to address the SSAC advice.

Moving on, there was also some discussion on … Sorry. There's a hand up so let me stop here. Tomslin, please.

TOMSLIN SAMME-NLAR: Thanks, Sarmad. I just wanted to check about the allocatable and blocked variants—the criteria for picking which ones were blocked.

SARMAD HUSSAIN: That's determined by the script community which is designing the Root Zone LGR. So, for example, Arabic script community decided which variants or particular strings should be allocatable and which should be blocked, based on how Arabic script works. Similarly, Chinese script community decided their own. So it is really a decision which is done by the script community.

They actually described their motivation in the proposal documents they have published and that motivation can actually change from script to script. So I would suggest that if you have a question about a particular script, the motivation really is available in the proposal to review. But it is really done by the script community. Okay. So I hope that answers your question so we'll move on.

This is also, I think, relevant. What can trigger the update to an existing script in Root Zone LGR? There are multiple triggers which are possible. There could be evidence that an additional existing code point is needed for one of the languages considered. There is an additional language being considered, not considered

before. So as shared, for some scripts, there are hundreds of languages which are written.

And the way it was done by the generation panels was then shortlisted based on the EGIDS value. But over time, sometimes languages change their EGIDS value because, for example, a particular community actively took up their language and made it more—redeveloped the language or its use and now instead of EGIDS level six, it's gone to EGIDS level four. And therefore, it should be considered and added into the Root Zone LGR. So that kind of change is slow to happen but it can happen over time.

Also a constraint on label in a script can be relaxed without issues to accommodate a particular language. So there are all these rules which have been devised by the script communities. And over time, if a script community realizes that some rule could be relaxed—for example the security concern around that rule due to, for example, rendering issues is now resolved, those kind of decisions can be made but the community.

And then, of course, Unicode itself updates the standard, adding more code points and scripts. So over time, that can also trigger an update in an existing script LGR. But please do note that there is no automatic periodic revision of Root Zone LGR. It is actually triggered by an external event, by either input from a generation panel or a community member. We will talk about that as well. So it's not that Root Zone LGR is updated every year or every two years. It will be updated when we receive a proposal from a generation panel. Sorry. Yes. Please, Donna.

DONNA AUSTIN: Sorry, Sarmad. I was just going to ask has that process ever been triggered?

SARMAD HUSSAIN: There was, I think, a minor change, which was done by, I think, Malayalam generation panel, if I am not mistaken. But currently, I think, from my understanding is that every script, of course, has been doing their first proposals. So it's not really been triggered. Everybody's just been able to do the first version of their proposal. But over time, of course, we have a process where anybody can bring in a revised proposal and we'll obviously take it through the process which we follow.

DONNA AUSTIN: Okay. Thanks. Hadia has her hand up.

SARMAD HUSSAIN: Yeah.

DONNA AUSTIN: Hadia?

HADIA ELMINIAWI: Thank you, Donna. If you could elaborate a little bit about number three, "A constraint on labels in a script can be relaxed without issues to accommodate a particular language." How is this determined or triggered? How does this work? Thank you.

SARMAD HUSSAIN: Right. So if, for example, at some point there were issues with a particular code point because it was … Just to take a hypothetical example here, if there is a code point which is very recently encoded by Unicode, it takes about, sometimes, maybe months before that code point eventually finds its way into the support, into fonts and rendering engines.

So there may be code points which are not stably rendered and they, therefore, can potentially create security problems. And therefore, they may actually not have been considered at some point by a generation panel. And they may actually have put some rules or constraints for blocking such cases.

Over time, when rendering, for example, or other reasons the code point is stable or the language community and the script community decides that the rule which was restricting its use can be relaxed to allow it in certain contexts, they can obviously trigger that change.

So you could actually constrain a particular code point in the context in multiple ways. You could actually totally disallow it. Or you could potentially allow it in certain contexts but now allow it in other contexts. That is largely applicable for complex languages which are used in Southeast Asia, not something which would, for example, be used in alphabetic scripts. We'll talk about and give you an example of the rule in the demo.

DONNA AUSTIN: Jeff?

JEFF NEUMAN: Yeah. Thanks. I'm trying to think of the best way to ask this question and see if it makes sense. But when a group gets together from a script community and they decide on the LGR labels and it goes through the process, is that a group that exists for life or is that a group that submits its report and once it's done, it's disbanded? The reason I'm asking is when someone does submit a proposal to make a change, is it the same group that's constituted or does it have to be reconstituted? How would that work in terms of timing and everything else?

SARMAD HUSSAIN: It doesn't have to be the same group as long as it meets the criteria for a generation panel. We've seen, at least so far, that many of the groups which were formed are still generally available and around. We still have their mailing lists active and we can reach out to them, at least now, because Root Zone LGR, at least the first complete version, is still under development.

But over time, even if the original group gets disbanded, the new group can be formed. It doesn't have to be the same people as long as the generation panel … I shared with you the requirements for a generation panel, where it has to have at least five to six members covering different kinds of expertise. So that's what we're really looking for. It could be the same group, or a new group, or a mix. Thanks.

JEFF NEUMAN: Okay. I'm, again, just trying to think of in, again, the very off chance that we have an applicant that applies for a TLD string that maybe contains something that was not a valid code point and then they want to have their issue decided, it may take a while to actually constitute a team that meets those requirements. Is that a fair assumption or do you think it would be much quicker?

SARMAD HUSSAIN: Sorry. I didn't get your last part. May I request you to repeat the last part?

JEFF NEUMAN: Sure. And it's hard so I'm not sure I'm saying it right. But if we have an applicant for a string and it doesn't pass the DNS stability review because there's a code point that doesn't exist, or is not valid, or something like that, and that applicant wants to have that considered by the applicable LGR group, we'd have to consider the length of time it might take for the group to get together and be able to consider that and how that maps to the new gTLD process. Does that make sense or did I actually do a worse job?

SARMAD HUSSAIN: Yes. It's certainly dependent on a volunteer group outside of ICANN. It's a really community-driven process by the script community. I think there is next slide in which I will talk about that. But to quickly answer your question, it will be dependent on the … So obviously, we would go out to that community and try to formulate the group. But it is eventually dependent on whether the

group is able to form, whether they're interested in looking at the problem. So there is some external dependency there. Thank you.

DONNA AUSTIN: So just conscious of time, I think there's a lot of interest in having the demo conducted by Pitinan. So what I'd like to do is, Satish, if you can ask your question and then we'll give Sarmad the opportunity to finish his presentation here. And if there's any questions, please put them in the list and we'll have them answered over e-mail. But I think we really want to get to the demo. So, Satish go ahead.

SATISH BABU: Thanks very much, Donna, and thanks to Sarmad. My question follows on from Jeff's question, relating to the last but one bullet. This says that the proposal by the GP, the GP is actually dismantled after the last round. There is no GP. So what [group does] this proposal by the GP?

SARMAD HUSSAIN: Eventually, the script community is the GP. So what is suggested here is that the script community is cognizant of what's going on as far as the language changes are concerned; requests in the community, either from applicants or other places, is concerned; updates in the Unicode is concerned. And the script community is, in a way, reacting to those changes and then maybe, as they formulated earlier, they can reformulate themselves into a generation panel and suggest any changes if they think that needs

to be done against the existing proposal which was submitted earlier.

SATISH BABU: Thank you very much.

SARMAD HUSSAIN: Yeah. Maybe this can also help answer some questions. Eventually, the external trigger—a GP, an applicant, anyone in the community—anyone in the community can request a change at any time. So no one is actually stopped from suggesting a change in root zone at any time.

The change request, with description of evidence, can go directly to the generation panel or come to ICANN Org. If it comes to us, we will actually try to reach out to the script community and raise it with them. The script community—the GP is really the script community—evaluates the change and decides if it will go on to consider it or not. So if the changes, they think, is not correct or not needed, they may not even actually reform themselves.

But if they decide to reform themselves, then they would actually, obviously, do a complete analysis, update their proposal, and submit that formally back to the integration panel. And then it will follow the same steps. So it will develop the proposal, will take the proposal through public comment process, and then when the public comment process is done, will hand it over to the integration panel. The integration panel reviews and integrates if it accepts it. And then the next version of Root Zone LGR is published.

So that's all of the process which is followed. And as I said, it can actually be triggered by anyone at any time. So this is really completely open for anyone, including an applicant. But the process is that the script community should decide whether that change is actually warranted.

Let me just very quickly conclude and hand it over to Pitinan for the demo. So as far as next steps is concerned, we are finishing public … We want to integrate pending scripts. There are a couple of public comments open currently, for Latin and Japanese. Please do take a look at those. We are also going to remain open to the community maintaining and updating Root Zone LGR, moving forward.

And we're also going to remain open to new communities coming in, suggesting solutions for the Root Zone LGR. We're still waiting for Thaana and Tibetan has already announced. And of course, we'll support any additional scripts which may quality for Root Zone LGR. So let me stop here. Questions? Or I'll stop sharing my screen and hand it over to Pitinan to take us through a quick demo.

PITINAN KOOARMORNPATANA: Thank you, Sarmad and everyone. I think you can see my screen now. I need to show the whole screen so you can see multiple screens as one. I will take you through the demo today. What we are going to use is the LGR tool, which the link is here, lgrtool.icann.org.

Just a disclaimer a little bit is this tool, currently, is designed for managing an LGR—like create, or edit, or merge—used by the GPs when they work on the Root Zone LGR. Later on, we'll also add the IDN table review function for Contracted Parties. So it's not designed exactly for the SubPro application yet. Once we go to the implementation phase, we will need to redesign this. But the core mechanism of how to use the Root Zone LGR to get passed into the tool and then use it to validate will be used.

So on the right is the tool. We have these multiple modes. Today, we will use this advanced LGR tool to work on because we will have a lot of examples in Latin. Latin, it hasn't been integrated to the Root Zone LGR yet. It is now in the public comment, as Sarmad said. So this is the version being published for the public comment.

So for the tool, basically, first you need to upload the Root Zone LGR, which is the rules for a particular script first. And when you're going to use it to validate something, you select the relevant script and then it will use that.

Okay. So for the cases that we will go through today, I have three main things. First is just to show what happened if the original label is valid. Case number two is what will happen when the original is invalid. Basically, the first tool is about validating the original labels. And when it's invalid, it can be because of multiple reasons so we'll go through that. And then case number three, it will be when the original label is already valid. Then we will generate the allocatable variants for them. And this can be in multiple scenarios as well.

So let's go through to case one. So case one, validating the labels. So for this, I put the link here as well. We used the Latin script Root Zone LGR that is being published for public comment. So the XML and HTML is here. But if you want to know more about explanation, you can go to public comment page and then they will have supporting document to explain the background of the solution as well.

All right. Let's come back here and select "Latin," and then just take you through. So in the LGR itself, it has the XML form. So it looks like this for the machine to read. But the tool can also generate the HTML for it so we can read better. Basically, the LGR will have three main parts. So after this description, which captured the main point from the supporting document. Then it will go to the repertoire—the list of allowed code points to be used for registration, and then the variant set for them, and then any rules if applicable.

I'll just quickly go through to have the idea of the Latin LGR. We have 286 code points. So all these in this section are for repertoire. And then for each one, the GP also mentioned—like, for example, this code point is included because it's used in all these languages, with EGIDS scale information at the end so we can trace back why this be included. The work is based on the inclusion principle, meaning if you don't have a good reason why including this one, then it won't be included in the repertoire.

So let's see. For example, let's go to the variant set. Like a, we see set one. So this is the variant of a. There are five members. This is the result from integration with the related scripts. So we have seen here some code point from Cyrillic, from Greek as well.

So this is the variant mapping area. And then, for the rules, there's not much rules for Latin but there is the rules to try to minimize the number of allocatable variants. So let's talk about it when we reach there.

Okay. So now let's validate a label. Here, on this, you have this button here, "Validate label." And I will just go with the first example first, first is just an ASCII, bbt. This one is a dedicated TLD as well. Okay. So this is valid. And the tool will also try to generate the variant labels as well but for this case, let's look at the valid or invalid first. So, this one is valid. Everything is in the repertoire, b, b, and t. Let's move—

SARMAD HUSSAIN: You have a question—hand up by Jeff.

PITINAN KOOARMORNPATANA: Oh yes. Sorry. I don't see. Please go ahead, Jeff.

JEFF NEUMAN: Yeah. Sorry. So the version of the tool that you're using, that's not available to everyone, right? That's just your internal one? Because I'm not seeing those things when I got to the page.

PITINAN KOOARMORNPATANA: Ah, okay. Sorry. We can switch the mode here. This is external. If you go to lgrtool.icann.org, it will bring you to the last page you were. But if you see … Sorry. Okay. Can you see this one yet?

JEFF NEUMAN: Yes. Thanks.

PITINAN KOOARMORNPATANA: Okay. Thank you. Yeah. you just need to come to this mode, advanced LGR tool.

JEFF NEUMAN: Yeah. I don't have the same page.

PITINAN KOOARMORNPATANA: Oh. Maybe because of this—what is it called?—interactive, responsive display. Maybe it's smaller. Maybe it looks different.

JEFF NEUMAN: No. I don't have … I guess because I didn't import an existing file. But then it asks for an e-mail login eventually. Anyway, go on. I'm sorry.

PITINAN KOOARMORNPATANA: Sorry. Actually let me just do a quick demo on how to import. So if you are coming in, this one you won't see because it's by session. This can be used in multiple people at the same time. So if you are in your working space, you don't see any LGR yet. You come to "import" and then you select the XML that you downloaded from the page. Let me try Hebrew. So before, we didn't have Hebrew here. And then this is validating if it's … You

can see the msr-4 for now. And then Hebrew has come in. And if you go back to home, then you see Hebrew will show in here.

JEFF NEUMAN: Great. Okay. Thank you. Sorry to distract. Sorry.

PITINAN KOOARMORNPATANA: Okay. All right. So let's move on. Then I come back to Latin again and then I'll go to the second one. These starts to have the non-ASCII. We have this é. So right now, it says "valid" and it generates some number of variants. So this is the valid case. Let's move on first and we'll talk about variant later. I captured the screenshot for this as well because you would see the PowerPoint later.

All right. Then case number two, still at the validating the label but for this case, it's invalid. There are multiple scenarios that it can be invalid. For example, they used some code point which is not allowed for a TLD. In the top-level domain, only letters are allowed, not the numbers or hyphen. So if we have this, 1, it will say "invalid" because 1 is not in the repertoire—not in the LGR. Okay.

And let's move on. The label can also be invalid because it contains an excluded code point and that code point be excluded for the security reason. For example, this mark which is the Latin mark, apostrophe. It's not the single quotation mark you type on the keyboard as well. This has been excluded by the Latin GP. You can find an explanation in the proposal. Let's see if I can find it quickly. Sorry. Maybe I closed it. But for the code point, it looks

like punctuation marks is not allowed by the IAB recommendation, so this being excluded as well. So this label is not valid. And then let's move on, for the interest of time.

The label can be also invalid because it contains a code point which has been excluded from the repertoire because there is no language support. So, for example, this one code point, 018C, is in the Latin script code chart. It's in the Maximum Starting Point. But the Latin GP couldn't find a supporting language, whether this code point be used in any of the languages that fall into the criteria of EGIDS one, to, four, and five—this condition. So for that, this is not included. So these are the three cases for invalid, due to the code points is not in the repertoire.

Then we also have this example from the complex script. This is for Thai. So for this one, the original is invalid because it's failed the Whole Label Evaluation rules. This is the rules required for complex scripts, which are used in the South, some East Asian languages as well.

So, for example, when I will try to validate this label, I need to go back and select Thai LGR first. So I'll go back, select Thai script. So the first one is like the base character. And then, for us, we have tone marks. So this one, 0E49, is used to change a little of the tone of the word but it's not expected to be more than one on a consonant because otherwise, basically, you cannot pronounce and it's not how the script works. So it cannot display consistently across the system because all the software or devices, the will just expect only one tone mark in this position.

So if I have only one tone mark, first it will say valid. 0E49 can follow a consonant, which is okay. But if I type more than one—let me type three more so now we have four—it will say this is invalid because of this one has to follow some consonant or some vowel. So it cannot follow the tone mark itself. And these rules will be encoded, along with the code point. So when it encounters this code point, it will check whether it satisfies the rule. And if it's not, it will be invalid.

This is just to show that it's not really a spelling thing. It's really, systematically, how you encode this. So let me type many, many things and let me show you how it works. I'll just paste it here in PowerPoint. You see? The tone marks just go on forever, which is differently if you paste it in the URL bar. It's also something else. So in this case, it's what the GP tried to prevent, to prevent an unstable rendering when it comes to the complex scripts.

Okay. So this is the case 2.4. The original label is invalid because of the rules. Then let's move on. I pasted the screenshot here for all of them—bb1, bb, and then the apostrophe-like, and then for this one not using any language, and then for this one it's just not appropriate use of the script.

Okay. We have just a few minutes left. Let's move on to allocate variant labels. So if the original label is valid, then the next thing is to calculate the variant labels of it. And for allocatable variant is what we focus on. It can be multiple scenarios. First, this label doesn't have any allocatable variants at all. Let me go back to Latin LGR.

DONNA AUSTIN: Folks, I see that we're three minutes from time. Pitinan, two questions. How much time do you need and could you stay on for 10 minutes, if necessary, just to complete the presentation?

PITINAN KOOARMORNPATANA: I think I can. I have back-to-back meetings but my agenda might come later. So I can stay on for 10 minutes and I think that will be sufficient.

DONNA AUSTIN: Okay. I appreciate folks have to drop but we'll continue this for 10 minutes and then the recording will be available. So apologies for this going over but let's see if we can finish it up. Thanks, Pitinan.

PITINAN KOOARMORNPATANA: Thank you. All right. So this is the case where the variant generated but all are blocked—so no allocatable variants. For this, the Latin GP defined that the s has a variant with the s lookalike in Cyrillic and this one is from Malayalam, the Indian script. And this is the Latin s.

Then let's go to the next one. This one can generate allocatable variants because the ß here has an allocatable variant relationship with ss. Let me try this. Because as I mentioned earlier, the tool is not really designed for taking much load yet so sometimes it can be quite slow. And if it starts to generate too many variants, it will give that in CSV and we have to download it. So I already have it downloaded so let me open.

Okay. So this is the result of the straße, or a German word. In the Latin LGR, they have the definition of variant mapping between ß and ss. So if we replace the ß with ss, then this is the allocatable one. And then also, they have other permutations of other things because ß can also have the relationship with the Cyrillic lookalike s and Malayalam that we see in the previous case. But all will be blocked.

So for this, we have one allocatable. And then we also have this multiple—like too many allocatable variants, like the shabaka that Sarmad displayed as well. We have eight of them. They generate 24 in total and eight of them are allocatable. So if we start to have TLD for eight and then maybe the second level, we also have another eight, then when registrant gets a domain name, it can be 64 quite easily. That's why we need to have some mechanism to reduce the number of allocatable variants.

So I think let me move forward. Then this is also important concept on the variants as well—so I'd like to put it here—is the index variant label. This is the mechanism, how to identify which labels have the same variant in that same variant set. So the index variant label is the variant label which has the lowest code point value in all positions when it's in the permutation of replacing with the variants.

So if you go back to the dés example, the first two, d and é, doesn't have the variants but the s has three members in the set. So 0073, 0455, and 0D1F, they are variant to each other. So when they replace this with three possibilities, it generates a set of three. But the one that be the representative, per se, we call index

variant label, would be the lowest one. So the first one is index variant label.

And this index variant label is used to check whether these are in the same set. So it's used in two places—to check if the original labels collide with any existing TLDs and also their variants. So if the new labels come in, you want to also know whether this one collides with the existing TLDs, also as the variant of existing TLDs. So moving forward, each TLD will have the variant index information associated to it and then that's where we need to check.

The second point to use the index variant is when the label is already validated and the requested variant labels come from the application. And we would like to know whether this requesting variant label is the allocatable of the original one. Then these two should have the same index variant.

And I think that's the last slide. So just to recap where it is happening. Sorry, Edmon. Let me read.

DONNA AUSTIN: Sorry, Pitinan. Edmon is responding to a question from Jeff.

PITINAN KOOARMORNPATANA: Okay. All right. So this is the flowchart that Sarmad presented as well. So first, to see if the label is valid, we use the LGR. We call it the element LGR. And then we calculate the index variant of this original label. We use the common LGR, which is basically the same thing but every disposition, every time will be

blocked. So if the index variant is unique, meaning that this original doesn't collide with anything in the existing TLDs, then it is accepted.

Then the next step would be see if the variant labels it's requesting is the allocatable variant of this one. And if it's yes, then okay. This is the allocatable variant. If it's no, it's rejected, technically. But of course, this also, the policy needs to decide a final after this calculation is done as well.

All right. So that's all I have. And happy to take any questions, if time allows. Thank you.

DONNA AUSTIN: Justine?

JUSTINE CHEW: Thanks, Donna. I'm trying to think about how to frame my question to make it clear. Pitinan, thank you for the demo. I think it's been very helpful. Can I just say, is it correct to say that assuming that you use a current LGR, Root Zone Label Generation Rules, whatever script community or whatever GP LGR has been put into the framework, if an anyone puts in the same label to check for validity, the result would always be the same, right?

PITINAN KOOARMORNPATANA: Yes.

JUSTINE CHEW: And it's subject to what XML files you upload. But that XML file is determined by the GP.

PITINAN KOOARMORNPATANA: Yes.

JUSTINE CHEW: I know. I'm just trying to keep it as simple as possible. So there is no way that the same label can come up as invalid or valid unless you change the input, which is the Label Generation Rule.

PITINAN KOOARMORNPATANA: Correct.

JUSTINE CHEW: Okay. Great. Thank you very much.

PITINAN KOOARMORNPATANA: Thank you. Any other question or you want to run some other example? I have Chinese uploaded but because it's quite large, I didn't put it on. Let me see. Perhaps if you're still on here, if you have a few minutes, let's see here. Just would like to show. This is the effort from the community to try to reduce the number of allocatable variants as well.

This is the Chinese LGR and they have the specific type to manage the number of allocatable variants. Obviously, many of you might know Chinese has the traditional version and then the

simplified version. So this is one example. This set has two members. And on the top is the traditional. Below is the simplified version. So these two can be used interchangeably, defined by the LGR.

So if you start to have four of them—that takes a while—then it can be two by two, by two, by two. So it's 16 possible variants in related, right? And everything will be blocked except the one which is all simplified and all traditional. So the one with the mix, the Chinese GP decided the rules, in the way that it won't result in the allocatable version. So instead of 16, then it's down to two so it's more predictable.

Okay. I think if there is no more question, I'll … I'm sorry, Jeff. Please go ahead.

DONNA AUSTIN: Very quickly Jeff so we can [inaudible]. Thanks.

JEFF NEUMAN: Yeah. Sorry. Hebrew and some other scripts read right to left. Does the generation tool account for that?

PITINAN KOOARMORNPATANA: Right. Let me show you. I'm not sure if I can find it quite quickly. So basically, the first code point that you type in, it will be the first code point as well, which means it will display on the rightmost for this script. But in the matching, it will just see the

first, and second, and so on. I don't know if I have this open. Yeah. But the tool can manage that. Let me try to show this, D0 and D1.

JEFF NEUMAN: But if I were to enter a label into the tool to check, do I have to then do it backwards if I was checking a Hebrew string?

PITINAN KOOARMORNPATANA: If you are a Hebrew user, you can just type in your keyboard or locale system and then it will just right to left. But actually, the tool can also take the code point form as well. So, for example, if I choose the code point here … So you can see that the rightmost is the first one.

JEFF NEUMAN: Got you. Okay. Thanks.

PITINAN KOOARMORNPATANA: All right. I'll turn the floor back to you, Donna. Thank you so much.

DONNA AUSTIN: Thanks very much, Pitinan. And thanks to everybody for staying on for the extra 11 minutes, as it is. We had a couple of other items on the agenda but we'll just take those to the list or we can do them for the next call.

One thing I do want people to respond to is I think we can go ahead with a working group call during ICANN 72, based on the Doodle poll that we did. But I do note that we don't have ALAC members that can attend. We do have ALAC participants that can attend. So I'm really looking to the ALAC team as to whether they're okay to go ahead with the working group call for ICANN 72. I appreciate folks have to drop so we will take this to the list and I think we can close out this call. Thanks, Terri.

TERRI AGNEW: Thank you, everyone. I will stop the recordings and disconnect all remaining lines since the meeting has been adjourned. Stay well.

[END OF TRANSCRIPTION]