TERRI AGNEW: The recording has started, and this is Terri Agnew. Good morning, good afternoon, and good evening, and welcome to the Latin Script Diacritics PDP call, taking place on Wednesday, the 1st of April, 2026. We have no listed apologies for today's call; however, Anil Jain did let us know he will be joining a little late.

Statements of interest must be kept to date. If anyone has any updates to share, please raise your hand or speak up now. Seeing or hearing no one, 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 shortly after the end of the call. Please remember to state your name before speaking for the transcript.

All chat sessions are being archived, and as a reminder, participation in ICANN, including this session, is governed by the ICANN Expected Standards of Behavior, the ICANN Community Anti-Harassment Policy, and the ICANN Community Participant Code of Conduct. With this, I'll turn it back over to our Chair, Michael Bauland. Please begin.

MICHAEL BAULAND: Thanks, Terri. Welcome everybody to April Fool's Day, Latin diacritics. I was thinking of what April Fool's joke I should make in the beginning, but then I thought, "No, diacritics is a serious topic; don't make any jokes now." Someone is still laughing. It didn't work then. We have a schedule which looks a bit similar to last week since we still have similar topics to cover. Next slide, please.

We did the welcome, and with that, the recap. We've been looking at PR 6 and IG 8, which are somehow related. We did some discussion there and decided that we need two scenarios for the recommendation PR 6: one in the case where there is more than one LD TLD available, and the other in case it's just one LD TLD and the ASCII one. We'll take a look at that later on. For the IG 8, we also had some discussion last week, and we have now prepared some graphics to make the cases which can occur a bit more visual, so the decision how to deal with that should be a bit easier. Next slide, please.

The action items from last week were that we review the proposed draft from the ICANN meeting and last week's meeting and provide any input if necessary. We divide the PR 6, as I just said, into two scenarios, which we will be taking a look at on the next slides. IG 8 will provide the examples to better discuss this. Fourth, we will summarize what's been going on on the mailing list regarding the topic of what is a diacritic in general, what is a diacritic in our work, and what should be included here and what should be considered out of scope. Next slide.

Just a quick reminder because there has also been some confusion last week: we are currently looking at PR 6 and IG 8. Both of those are in the application process, which is marked here in red with an arrow and blinking, so hopefully most of you can see that. The other topics, like PR 15 to 28, are in the allocation process, which is after the application. Then the PR 29 to 38 recommendations are during the delegation process. All that we are discussing here right now refers to the application process, and anything that's in delegation or allocation is not really important in this context. Next slide, please.

We have been working on the PR 6 quite a bit. You can see on the red and strikethroughs, underlines, and yellows that this is still work in progress. I won't read it now, but we'll take a look at the new version, which we suggest covers all the important discussion points from last week in a minute. This is just as a reminder of what we've been looking at. Next slide, please.

Regarding Bill's comment, I would say that the other two, to the extent that the definition of diacritics matters, would be covered by the definition we come up with here, provided we reference it. I am missing the context. Bill, do you want to raise your hand and say something? I was too fast with the slides. Should we return to that? No need. Sorry if I'm going too fast. If you have questions or comments, just stop me.

For the PR 6, we wrote down a rationale, and I will read an excerpt from that rationale for PR 6 now. In accordance with EPDP IDN's Phase 1 Final Recommendation 325, the LD PDP working group agreed that withdrawal of an applied-for Latin diacritic gTLD label should be allowed after the application has been submitted, but adding a new Latin diacritic gTLD label to that submitted application is prohibited. As mentioned in the rationale for Preliminary Recommendation 5, primary and variant strings do not exist in the context of LD PDP.

However, the base ASCII gTLD label will be treated like a base label, and it must exist for the delegation and operation of the said ASCII and Latin diacritic gTLDs, as outlined in Preliminary Recommendation 1. Clearly, the withdrawal of an applied-for ASCII gTLD label that results in a single IDN gTLD, a Latin script gTLD, will no longer be considered part of an ASCII Latin diacritic gTLD set. To ensure clarity, "application" here refers

to the application of an ASCII Latin diacritic gTLD set. That is what we decided as a suitable rationale for our Recommendation 6. In particular, it states that LD TLDs can be removed, but also an ASCII TLD can be removed. Then we don't have an ASCII Latin gTLD set anymore, but just a single application. Next slide, please. What this means... Bill, please.

BILL JOURIS: I'm just way too slow today. On the previous slide, I think we should be clear that you can remove one Latin diacritic gTLD and still have a set as long as it isn't the last Latin diacritic gTLD. If you see the distinction, if you've got five and you remove one, you've still got a set.

MICHAEL BAULAND: You're actually faster than you think, because this is exactly what we'll be covering on the next slide. You're not too slow; you're too fast. It's a good point exactly. These are the two cases we will be covering for Recommendation 6. We just wanted to first provide some background information on how we derive this and then show what we think the original decision of the working group had been. It possibly hasn't been expressed very accurately, and that's why it caused some confusion. That's why we suggested changing the wording of Recommendation 6, but not the meaning. Next slide, please.

As a reminder here, we have two kinds of processes for applications. We have the standard process. If you start with two TLDs, ASCII and LD, then these will be considered completely separate applications. They will be compared due to string similarity review, and if they are found similar, there will be a contention set and only one of those TLDs can

proceed. If they are considered not to be similar, then both strings are eligible to proceed, and they can coexist without any restrictions on how they are handled. The other option is what we are doing here in the LD PDP, creating new policies for, which is that people can apply for those strings at the same time as a set.

In that case, the two will be part of a string similarity review, but they will not be compared to each other, just to everything else, so they can't be in contention with each other. We also decided that applicants have to choose which process they use, but they can't switch between the processes. In particular, we said applicants may choose which process to use, but the working group determined that switching between the standard and exception processes will not be permitted. Applications submitted under the exception process as a single set cannot later be unbundled.

This unbundling applies solely to the fact that you have two or more TLDs in a single application as a set and then want to unbundle them and have them each as their own application. That's not possible. But it is not a problem to drop all but one TLD and then continue with a single TLD. This is not unbundling; this would dissolve the set and therefore is okay. I see Bill's hand again. Please.

BILL JOURIS: I am really busy this morning. Suppose you have a set and you, as per the previous recommendation, remove one item from the set. Can you then separately decide you want to apply for that one as an entirely separate application, or is that automatically covered by unbundling? If

it is covered by unbundling, can someone else then apply for it? I'm just a little confused on how those two interact. Thank you.

MICHAEL BAULAND: Good questions, thanks. This is all assuming that the application window has closed. Unless the application window is still open, you can adjust your application and everything is possible. You can apply for a new one or drop your application. That's all at the moment when the application window is closed and the evaluation starts. You cannot unbundle this to take one of those applications from the set and continue with that as a standard process. You can only dissolve the set, so to say, removing all but one, and then this will automatically be a standard process TLD. It's not an unbundling; it's just falling back to a single one.

Regarding your question whether someone else could apply for that, yes, but not right now because the application window already closed. In the next application cycle, you or somebody else can try to apply for such a TLD again, but with the likelihood that it will be considered confusingly similar with the set that was already allocated. I hope that answers the question. I see Tapani's comment: "If one already has an ASCII TLD and applies for several LD TLD version of it, what happens, or is that covered somewhere else?" Good question. I'm not 100% sure we have covered everything with this process of adding TLDs to existing ones.

At least intentionally it's included in PR 6, but maybe we indeed need to be a bit more explicit about that. Let's just first go to PR 6 and then see. I see someone put something in the chat. Preliminary Recommendation

3, consistent with Final Recommendation 3.2 of the Phase 1 Final Report of the IDN EPDP: A future gTLD registry operator wishing to apply for a corresponding ASCII gTLD or Latin diacritic gTLDs of their existing gTLD must submit an application during an application round. Yes, but the question is still if they apply for two LD TLDs to be added to their existing ASCII TLD, can they drop one of the LD TLDs?

Yes, why not? It's a standard rule that you can drop an LD from the set. Could they also drop the one remaining LD TLD? Yes, because that would simply mean they stop the application process, so they wouldn't apply for anything. I think it should be okay, but if anybody thinks we should have some explicit wording for that, we can also look into this. But let's first take a look at the PR 6, how we think it was meant to be, and then we can see about that one. Next slide, please.

The LD PDP working group agreement, as far as we understood it, is that both applied-for ASCII and LD TLDs are considered a set from the outset. Switching between the standard and exception process is not permitted, and the set cannot be unbundled, i.e., transformed into two or more standard applications. When an applied-for ASCII and an LD TLD remain that were treated as a set from the outset, the set remains intact. Switching between the process is not permitted, and it cannot be unbundled.

It is possible to withdraw either of the applied-for gTLD strings and proceed with one single standalone TLD. However, it will no longer be treated as a set and cannot proceed through the exception process, but only the standard process. That we think was the meaning and intention what we decided with this PR 6. To make that a bit more clear in the

way how PR 6 is written down, we used Sarmad's suggestion to have two cases in PR 6. Next slide, please.

The proposed language for the new version of the Preliminary Recommendation 6 will read as follows: Consistent with Final Recommendation 325 of the EPDP IDN's Phase 1 Final Report, after submission of an application, the applicant is allowed to withdraw an applied-for gTLD string from that application, but is not allowed to add any other gTLD string that was not originally applied for in that application. This is exactly what's also in the IDN EPDP. You can remove strings, but you're not allowed to add strings, with the exception that we don't have a primary label here, so we can be a bit more lenient in that sense.

The recommendation then reads on. This can be divided into the following scenarios. 6.1: If the application contains a single ASCII gTLD string and multiple Latin diacritic gTLD strings, and one applied-for Latin diacritic gTLD string is withdrawn from that application, the application will continue to be treated as a set under the exception process. As long as you have at least two Latin diacritic gTLD strings, you can always remove one of those Latin diacritic gTLD strings, and the set continues to exist, and you can continue.

The other case is: If the application contains a single ASCII gTLD string and a single Latin diacritic gTLD string, and either of the applied-for gTLD strings is withdrawn, the set will cease to exist, and the application of the single gTLD string may proceed only through the standard process. Then we have the addition about the brand's peculiarities and that they have a replacement string, but I won't go into this because

there was no contention about that. That's just a standard brand exception process. Any thoughts about the new wording? Does it cover the meaning we intended here, or does anybody still have problems with it? Is there anything unclear? Satish, please go ahead.

SATISH BABU: Thanks, Michael. Satish for the record. Can we go back to the slide? Thanks. For clarity, the third line says, "This can be divided into the following scenarios." Can we modify that to, "This can be divided into one of the following scenarios"? Otherwise, there are two, and after 6.1, there is no "or" after the semicolon. Sometimes you put an "or" just to ensure that only one of them applies at any point. Since that is not there, can we have "divided into one of the following scenarios" so that it's very clear that only one applies? Thanks.

MICHAEL BAULAND: Sounds good to me. It's just a small change and avoids confusion that both of them can apply at the same time. I think it's a good suggestion. Any other thoughts or comments? I see Tapani's comment: "I was just wondering if we'd need explicit language for the removal of some TLDs from such an add-on application." That's what I understood in the case that you already have two cases to consider. Either you already have an ASCII TLD and you want to add one or more LD TLDs to that ASCII TLD, which is possible with removal of such TLDs. I think it's quite obvious that in that case you can remove any or all of the LD TLDs.

If you remove the last one, it means you go out of the application process. A bit more complex would be the situation if you start with a

gTLD and you want to add an ASCII and LD TLDs and/or LD TLDs to that. A similar situation would occur here. In general, you would need to count the number of gTLDs both in your application plus the already existing one. If you have an existing one and apply for an ASCII and a gTLD, an ASCII and an LD TLD, it's not possible to remove the ASCII while still trying to add an LD TLD to that existing one because then you would end up with a set of only LD TLDs without ASCII.

From my view, "obvious" is too much to say, but it's clear that the intention is the same. I agree with Tapani that maybe we need to have either a new recommendation to cover that case or add some words into this one. I am not sure how this would work out. Bill, please.

BILL JOURIS: I would agree with the suggestion that we add the additional words because at this point, "obvious" tends to mean it is obvious to us. Whether it will be obvious to someone who hasn't spent months churning through this is a whole different question, and I think we're probably better off going for what seems to us excessive wording just to avoid that issue. Thank you.

MICHAEL BAULAND: Good point, thanks. I guess we'll discuss this in the leadership team to decide whether we add some wording to this recommendation or whether we create a new recommendation to cover this. But at least for now, is everybody okay with the meaning here? Sarmad, please.

SARMAD HUSSAIN: In the case where there's an existing gTLD, ASCII or Latin diacritic, and more are being added in an application round, and then through that application round some of those could be dropped as per 6.1 and 6.2, I was wondering whether in that particular case the recommendations would be a bit more restrictive where the existing gTLD maybe could not be dropped. I think that's slightly different from scenarios 6.1 and 6.2. Thank you.

MICHAEL BAULAND: Thanks, Sarmad. That's a good point. The existing one shouldn't be able to be dropped because that's out of scope for this application round. It's an existing TLD which can't be simply dropped. There are maybe registrations under it, and that would have to go to the standard retirement process, if any. Dropping would only be possible for the newly applied-for strings. Any other comments regarding this version where we currently just look at applications without an existing one?

As said, we will come up with some wording that will create the same rules if there is already an existing one but makes it a bit more clear than just having this. I see some agreements here from Tapani, Anil, and Satish. I am waiting a bit longer if there's anybody objecting to that or has some comments why this is not a good approach or why this is not what we originally intended last year. I see none, so okay, great. I think we are done with Preliminary Recommendation 6, at least for the case with only new ones, and the other will just be with consistent wording; we will create the other one. Next slide, please.

With that, we come to Implementation Guidance 8, which is somehow related. It is the implementation guidance requiring, with respect to the evaluation of the information submitted per Preliminary Recommendation 7, in accordance with Implementation Guidance 3.6 from EPDP IDN's Phase 1 Final Report. This is talking about the evaluation, what criteria to put there, and how this evaluation is to be carried out. The comment from ICANN was that first there is a need to address the impact of an evaluation outcome if an applied-for ASCII gTLD string in a set fails evaluation. We will be looking at this in a second.

Second, there is no LD PDP output corresponding to EPDP IDN's Final Recommendation 3.1. ICANN org suggests that the working group revisit and assess whether a corresponding output could be developed to clarify the treatment of ASCII gTLD strings in the context of application and evaluation process. Final Recommendation 3.1, which is referenced there, reads, "An application for an allocatable variant label cannot precede an application for that variant label's primary gTLD string." Let's maybe first take a look at this second point, where in the IDN case, we have the rule that the primary label needs to be first and only then the variant labels can be.

As far as I remember, for the LD PDP LD set, we decided we do not need this because there is no primary label as such. We have an ASCII label which needs to exist, but there is no real relationship like for the variants. The variants depend on the primary label; they can't exist without it. But for us, the LD TLDs are not really dependent on the ASCII other than that we say the set must have an ASCII. A standalone LD TLD is totally fine, and for that reason, we decided we do not need such a

corresponding recommendation. Does anybody have different opinions or remember differently or see a problem with that? I see none. Unmute, Louis.

I'm just checking the chat here with Sarmad. Sorry, one second. I think we agreed for point two that we don't need that, and we can then take a look at point one, which says what happens if the ASCII gTLD string set fails this evaluation. For that, we have prepared some graphics that show the possibilities. Next slide, please. I guess next slide again, then. This is just... yeah. Here we see first we have a scenario with just two applied-for TLDs, and after that, we'll look at the scenario with three TLDs.

We have an ASCII TLD applied for and one LD TLD applied for as a set. Scenario 1 is both applications pass the evaluation; then there's of course no problem. Both can proceed. Scenario 2 is the ASCII TLD passes the evaluation, but the LD TLD fails the evaluation. Then the set will be dissolved as already covered in PR 6, and the application can proceed as a single standalone application. Similarly, Scenario 3, if the LD TLD passes and the ASCII TLD fails, again, this would be possible to answer ICANN's question. That would then again mean the set is dissolved, but the LD TLD can proceed as a standalone single application.

For completeness sakes, we also have Scenario 4, in which case both the ASCII and the LD TLD fail the evaluation. Then of course nothing can continue because both TLDs are deemed invalid and cannot proceed. Any questions or comments regarding this scenario before I continue with the one where we have two LD TLDs? Seeing none, next slide, please. Satish, sorry I was too fast.

SATISH BABU: Sorry, Michael. I did not understand Scenario 3. Here it says that there is no ASCII, but just a standalone LD TLD. Aren't we saying elsewhere that there has to be an ASCII anchor for every LD TLD?

MICHAEL BAULAND: Thanks. That's a good question. It basically falls back to the PR 6 in the sense that if there are just two TLDs remaining, the applicant is free to drop one of those TLDs, and as a consequence, the set would be dissolved and the remaining single TLD would just continue as a standalone application. Since the set is dissolved, also the requirement to have an ASCII TLD is dropped because we just have one single LD TLD.

SATISH BABU: Oh, okay. I get it. Thanks.

MICHAEL BAULAND: That's the difference. As long as the set still exists, we have the requirement that we need to have one ASCII TLD, exactly one. As soon as the set dissolves because there's only one TLD left, then also the requirement for ASCII is dropped because we don't have an LD set anymore and we just continue with a single application. Then the a bit more complex case of having two or more LD TLDs. Next slide, please.

We have here eight scenarios because those are the combinations two to the power of three if we have three TLDs. This will be similar if we have more than two TLDs, but this case should cover all the corner cases

that can occur if we have two or more LD TLDs. For Scenario 1, all three TLDs pass the evaluation, the ASCII TLD and the two LD TLDs, so there's nothing to worry about. All three can continue and the set is intact. For Scenario 2, the first LD TLD would fail the evaluation, but the ASCII and the second LD TLD will pass. Again, this doesn't cause a problem because it would just mean that one of the LD TLDs can't continue, but the set is still intact and we can continue with the other two.

Scenario 3 is basically the same, just that the other LD TLD fails the application. Again, the set remains intact. We continue with just two TLDs and the failed TLD would have to be dropped. Scenario 4 is that the ASCII TLD passes the evaluation, but both of the LD TLDs would fail the evaluation. This is also similar to the previous slide scenario. This would mean that the set is dissolved and we just continue with that single ASCII TLD as a standalone process. I will postpone Scenario 5 for a second.

Scenario 6 is the one where the ASCII TLD and the first LD TLD fail and only the second LD TLD passes. This is similar to Scenario 4, and as mentioned in PR 6, this means that the set is dissolved and we just continue with a single LD TLD. Scenario 7 is just the other way around. Again, ASCII and one LD TLD fail the evaluation, but this time it's the first LD TLD that passes the evaluation. In that case, again, the set is dissolved and we can continue with the remaining LD TLD as a standalone application.

Now back to Scenario 5. This is the only one here that causes some issues or needs a bit more time to look at. In Scenario 5, the ASCII TLD fails, but both LD TLDs 1 and 2 would pass the evaluation. This again

refers to ICANN's question: what would happen in that case? Here the problem is that if we were to continue here, we would have two LD TLDs but no ASCII TLD. So the basic requirement that one ASCII TLD always exists in the set is not met.

For that reason, if the ASCII TLD has to be removed, it means that the applicant will need to choose one of the LD gTLD strings and drop the other. If they do and just end up with a single LD TLD, then they continue similarly to Scenario 6 and 7, where they just continue with one standalone TLD. The big problem is if the ASCII TLD fails, they can't continue with the two remaining LD TLDs because that would mean we have a set without an ASCII TLD. Anil, please go ahead.

ANIL JAIN: Thank you, Michael. Anil for the record. Scenario 5: it is understood that the moment the ASCII TLD fails evaluation, the set between ASCII and LD TLD is dissolved. That is understood. But if in similar other cases we were continuing to evaluate without a set but as an independent LD TLD, I want to understand if I have correctly understood that we don't allow two LD TLDs which have cleared the evaluation criteria to go through, but one of the LD TLDs has to be dropped or has to be submitted back by the applicant. Why? I am not able to understand. Why can't both LD TLDs go through as independent LD TLDs without the set with ASCII? Thank you.

MICHAEL BAULAND: Good question. If we could go back two slides, please, to Slide 14. Thanks. The reason why this is not possible is that we disallowed

switching lanes, so to say. You have to decide whether you want to start with an LD PDP process, and if you do that, there is no unbundling. So you're not allowed to then switch to the situation where you have two or more TLDs unbundled and go into the standard process.

If we were to allow people to continue with both of the LD TLDs in case the ASCII failed, this would essentially mean we are unbundling the set, and this is something we decided would not be allowed. You are allowed to dissolve the set and just continue with a single TLD, but you're not allowed to unbundle the set and continue with two or more standard applications. Does that answer your question?

ANIL JAIN: Thank you, Michael. To some extent, yes, I understood. But my second question, if you permit me. In case somebody drops one of the LD TLDs in this scenario and proceeds with the delegation, and after some time once the delegation is complete, that dropped LD TLD is again applied for by the same applicant, is it possible?

MICHAEL BAULAND: As I said earlier, this is just the application process. What happens afterwards is out of scope of this recommendation. But in general, if you have already either a single TLD or an LD TLD set, in a later application round you can apply for all not-yet-applied-for TLDs or even for the applied-for, but that will automatically be rejected. So sure, you or anybody else can apply for any further LD TLDs. They can either do this in the standard process, which means that they want their TLD to coexist with the existing set, but that could cause string similarity

rejection due to the already existing ones. Or, if they are the same entity that already owns the LD TLD sets, they could add to that set again one or more LD TLDs.

ANIL JAIN: Thank you, Michael. Thank you for the clarification. Thank you.

MICHAEL BAULAND: Okay. Back to Slide 16. Thanks. Yes, Tapani said there is no good reason to disallow an LD set without an ASCII, but that's out of the question here because in the charter it says that an ASCII TLD must be there. We already discussed this last year: due to the narrow scope here, we don't want to change the charter at this point in time. Once we finish this, we can always see whether there are some further additions, other PDPs that would make more choices possible, like adding different code points that are not diacritics or being more lenient in the sense that an ASCII TLD does not exist. But for our PDP, that's out of the question. We require the existence of an ASCII TLD. Any questions or comments here? Satish, please.

SATISH BABU: Thanks, Michael. First of all, this presentation and this slide are very useful. For me, it enables me to understand the logic very clearly, so thanks for the presentation of this particular slide. Now when you say "set dissolved," that means basically it goes out of scope for this PDP. Is that right?

MICHAEL BAULAND: Yes. Okay. Any other questions or comments here? If not, I think we have an answer to ICANN's question about what happens if the ASCII validation fails, and we can write that answer down. With that, we come to the fun part. Oh, no, no fun on April Fool's Day; sorry. About what is a diacritic and what is the scope of our work.

There's been some discussion kicked off by Tapani with his argument that possibly our definition of diacritics is a bit too restrictive in the sense that it's only saying we include characters that are decomposable in Unicode or which are already decomposed in the LGR. His suggestion was to be a bit broader, and due to what is defined as a diacritic also in Wikipedia and other sources, decomposability is not a requirement for a diacritic. For that, he suggests to include cases where the character is called something like "Latin small O with a stroke" or something similar.

As a counter to that, we have what we have been dealing with now, which Mark quite well summarized: namely that we are just looking at a narrow-scoped PDP and we are just looking at technical decomposability. I don't know if either of you wants to provide some further input here at this point or whether we should continue. In the next slide, we will cover what has been discussed in the mailing lists in the hope that all this information makes it easier for us to decide how to deal with this case. Seeing no hands, next slide. This summarizes the scenarios. Again, Anil, please.

ANIL JAIN: Thank you, Michael. Interesting observation by Tapani, and Mark has very carefully answered as per the charter of this particular PDP. But the

question here is that what Tapani is talking about, those scripts are diacritics which cannot be decomposed. They are existing in the system, and as per the charter, we are not supposed to deal with them.

But suppose we don't deal with them; then the work which we are doing in the PDP remains unfinished until another PDP is formed to handle the non-decomposed Latin diacritic cases. So there are two possibilities: one, the Board considers getting another PDP to handle such cases; or we as a PDP can refer this particular scenario to the ICANN Board and ask for clarification whether this PDP's scope can be expanded to complete the job of the LD PDP. This is just a thought process, but Michael or Mark might have some answer to this. Thank you.

MICHAEL BAULAND: Thanks, Anil, for that comment, but I think it's even a bit easier because the charter does not explicitly state that we have to only consider diacritics which are decomposable. That was a decision of the working group because that was found to be the technically easy and straightforward rule to have a demarcation of what is a diacritic and what is not a diacritic. The charter just says we have to deal with diacritics, and we had the discussion in the working group about what is a diacritic because that was not clearly defined, or different sources define this differently.

We came up with the rule that we go with decomposable Unicode. But if we think the argument by Tapani is valid and brings in new information that has been overlooked before, then without going to the

ICANN Board or changing the charter, we can include this here. Mark, please.

MARK DATYSGELD: Thank you. I would just like to clarify this because I was at the Council when this was being drafted. The recommendations are pretty clear and non-negotiable. Just to reiterate, we can't make changes to the RZ-LGR. That's not up for debate. So if we start from that notion, then what that means is we have to create an artificial form of validation of what we want to accomplish on top of the RZ-LGR.

The problem is we don't have that, Unicode doesn't have that, and ICANN org does not have that. Therefore, we are creating it as we go. The first solution we got to is one that makes sense in terms of this reality. Why? Because it is one thing for us to make the policy happen, and then it's another thing for when a database is handling a user, that their database actually folds correctly the TLD as a variant. It's one thing for us to do this at the DNS level, and this is something that has been a problem of IDN forever.

We create these rules that look really good, and then when it comes to implementation, they need a lot of work. Ideally, if we want to serve the communities we want to serve, the implementation of the rules has to be undeniable. It has to be something that we can point to and say, "Hey, here's the rule," and call it a day. That's what ultimately we're looking for, not to get this just working at the DNS level. At the DNS level, we can do whatever. We can make our own ad hoc charter with 600 characters, and if we convince the Board, that's game.

My concern is: what do we do once this lands in the hand of a project manager and somebody says, "Hey, we need to update this correlation"? What's the difference when they look at it and, on the one hand, you have this one-paragraph thing that says, "This corresponds to this," and what happens when the alternative to that is there's this ad hoc list, or they ship a library or a bundle of scripts that we need to run for diagnosis? In the 10 years I've been doing UA, Sarmad has been doing it for longer, and a lot of people in this group have been doing it for longer; we know that doesn't work.

The real question is how to keep—or whatever we decide, it doesn't matter what we decide—how do we keep it implementable and scopable? That's why I've been very keen on looking to Tapani's proposal because it follows in the spirit of being verifiable, repeatable, testable as we move towards this new idea that Michael basically summarized as, well, this can be essentially "small Latin letter with." That's pretty good. That's pretty implementable. When ICANN goes knocking on people's doors saying, "Hey, this is the new rule," it's very hard for them to come up with an excuse not to implement this.

So that's why I personally see this as healthy, but it's important that the group is on this page. I think we are technically arriving towards a certain direction, but it's important for the group to understand and lay out the arguments for whether this is a good direction, whether this is a good fix, whether this is something desirable, and whether it addresses the revisions that we were asked to do. In my personal opinion, we are headed that way, but interested members of the group should probably intervene as to whether this is a good compromise. Are we moving the

needle forward and not ignoring the problem while still keeping it implementable and testable? Thank you.

MICHAEL BAULAND: Thanks, Mark. Good points here. To summarize what Tapani suggested or what we have right now: at the moment, we are considering those diacritics which are either decomposable or are already decomposed in the root zone LGR. We already found out that this is a technically easy, single straightforward rule. Mark already implemented this and provided the examples which are generated by this rule.

Tapani argued—and please correct me if I say something wrong, Tapani—that while these are all diacritic characters, there are more diacritic characters, namely those which have a name called "small Latin letter with something." Because even though they may not be decomposable, they are still—as a basic definition of a diacritic says—you have something and then you put something around it, and this is exactly what is stated in the name of those characters. As Mark said, while this is not what we had before, it's still technically easy to implement this.

It's only a bit more complicated because we need additional rules. There are also two things to consider. If we just use "Latin small letter ASCII with something," this would leave out the letters that have two or more diacritics. So we would need to have something like "Latin small letter ASCII with diacritic" and possibly another diacritic. We also need to include those characters which are not described as this but which are a combination of an ASCII character and a diacritic. Tapani, please.

TAPANI TARVAINEN: Actually, it seems to me that all those cases where you have two diacritics are decomposable. So they will be covered if we have those two rules: include the decomposables and those "Latin small letter with diacritic." We get them all. You might check that, but I think they are there. Otherwise, you already said most of it. I made the case on the mailing list that basically with these two rules, we are closer to the charter in that we are covering more diacritics, and we would be supporting more languages and more letters, and it would be technically easy enough. So it seems like an easy solution to me, but others may disagree.

MICHAEL BAULAND: Okay, thanks. With that, next slide, please. There's also some interjection by Pitinan, who sent to the list that of course all strings must be valid as per the Latin root zone LGR, so we can't include any characters which are not in the Latin root zone LGR. That's an important point. Then she says that there are basically two types of characters. Type 1 is where the base character and the diacritic are encoded as a single code point, like the Latin small letter A with a grave.

Type 2 is where the base character and the diacritics are encoded as two separate code points, and this is because there is no single code point just for such a character. Examples here are a combination of 0061 and 0331, which is a Latin small letter A plus a combining macron below. These are the two types of diacritics to consider. As Tapani mentioned in the Type 1 example where we have composed characters,

there is a difference between those which are decomposable—like the ones which are put here, which we included previously—and the ones which are not decomposable but still have a name, "Latin small letter something with something."

He suggests to include these two. Next slide, just to have a border as to how far we should go. There was the third possibility: to also add letters that aren't real diacritics but that could still be handled in the same way. These would need to be evaluated individually, and automatic processing would have to be based on tables. It would, however, cover even more languages. Here the problem is twofold. If they aren't real diacritics, then they're out of scope because we are restricted to diacritics in the charter.

There's also a more practical problem. If you take for example the Latin small letter thorn, which is mentioned here, this looks like a combination of a P and a B, but what would be the base character here? Would it be P or B? This is also one of the problems here. For some it's obvious, but not for all. So basically, if we go... Anil, please.

ANIL JAIN: Thank you. Michael, I have heard the argument of Mark on not changing the root zone LGR and proceeding further on this. That is understood, and it is a simple and practical solution to be implemented by this PDP. But in the report of this PDP, are we going to mention somewhere that we have observed these kinds of diacritics which cannot be decomposed?

Since they cannot be decomposed, we need to change the root zone LGR so they can be handled at a later stage. Are we going to say somewhere in this report, or are we going to be silent on this? As far as the third aspect here which you have shown, I fully agree with you that they are not diacritics, so they are out of the scope of this particular PDP. However, there are some additional issues also. Thank you.

MICHAEL BAULAND: Just a quick clarification: there is no need to change the root zone LGR if we want to include the not-decomposable characters. If we could please go back to Slide 18, I guess that is the best to follow with the discussion. Mark, please.

MARK DATYSGELD: Thank you. To follow up on that, to make very clear where we really hit a wall: as a linguist, I would love to be able to, for example, include AE. AE is super important in Danish, in Norwegian, and in Icelandic; that would be a great thing. But then we start entering this AE and OE territory, which is multiple-to-one, and that becomes completely outside of the scope. We were not tasked with doing multiple-to-one at all.

That needs to be a separate PDP by design. It's not even extendable to our charter. This is just outside the scope of this PDP, for example. There are incredibly valid discussions to be had about other characters, but they clearly need the scope of a different PDP. I don't think there's anything wrong in our final report in outlining where we think the next

PDP should go and handing it over directly to the GNSO Council so that they can start scoping that as soon as we're done.

Some things are just outside our ability to act, and I don't think they're very extendable. Even supposing we went and said, "Oh, I want a charter rewrite," at that point we are changing our mandate period with some cases, and those are better put in a recommendation section of, "Hey, we found these things. We discovered our limits. We want to account for these things, and we urge the GNSO Council to get started as fast as possible on scoping this so that most of the people in this group can reconvene and we can get started on working on many-to-one and working on other issues that clearly we want to tackle but were not handed over to us." Thank you.

MICHAEL BAULAND: Wait. I'm losing you. Is it my internet connection or is it yours? Can anybody still hear me?

TERRI AGNEW: I can still hear you, Michael.

MICHAEL BAULAND: Okay. We have just one minute left. Satish, very quickly and then we finish.

SATISH BABU: I'd rather discuss it at the next meeting. I don't think I have time to discuss what I want to say now, so I can take it over to the next meeting. Thanks.

MICHAEL BAULAND: Okay, great. Then let's keep that for the next meeting. I think we have laid out all of the options here, and for the next meeting we will put them together in a more succinct way and then we can discuss which way we want to go forward. With that, we are at half past, or at least in our time zone; in the Indian time zone it's the top of the hour. With that, I close the meeting. Thanks everybody for attending, and I hope to find a solution for that next week. Thanks all, and goodbye. Terri, you can stop the recording.

[END OF TRANSCRIPTION]