

---

JULIE BISLAND:

All right. Good morning, good afternoon, and good evening, and welcome to the Latin Script Diacritics PDP call taking place on Wednesday, 25 March 2026. For today's call, we have apologies from Anil Jain and Prudence Malinki. Statements of interest must be kept up to date. Does anyone have any updates to share? If so, please raise your hand or speak up now. Okay, seeing no one. If you need assistance updating your statement of interest, please email the GNSO Secretariat.

All documentation and information can be found on the Latin Script Diacritics 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. And with that, I will turn it over to our chair, Michael Bauland. Please begin, Michael.

MICHAEL BAULAND:

Thanks, and welcome, everybody, back from Mumbai. I hope you all had a good trip back and a great time in Mumbai if you were there. Next slide, please. So we will basically continue our discussions from the two Mumbai meetings. We will review the public comments and see how we have to respond to them. Next slide, please. And yeah, so recap. Next slide, please.

---

*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.*

We had two sessions during ICANN. Again, on the first day, it seems to be that we are always kind of the first to start. And we went through the public comments. We started with the non-substantive ones first, and we started updating using the public comment review tool. We agreed to update the language as requested by ICANN due to some inconsistency in the notions we used. The majority of the working group members also agreed to bring back a missing clause explanation of how the primary and variant labels are considered the same. This similar clause is from the IDN EPDP, and we changed it accordingly to explanation of how the ASCII gTLD and Latin diacritic gTLD labels are considered the same. For some reason, we omitted that, and but it should actually be there.

Recommendation 6, 8, and 50, those we looked at, but we decided to discuss them further with the GDS liaison who is here now if I see correctly. Yeah. So because there were some issues which should better be discussed with the person there. Yeah, and then that is basically what we have been doing during ICANN. Next slide, please. And we had the four action items to update the public review tool. We have done that. The revisit of recommendations 6, 8, and 50, which we will do now. Staff followed up with the GDS liaisons for clarity on 6 and 11, which we have done, and we ask all working group members to review the public comment review tool again with particular focus on the preliminary recommendations 1 and 39.

Next slide, please. And with that, we come to the preliminary recommendation 6. And it says after submission of an application, the applicant is allowed to withdraw and apply for a gTLD label from that application but is not allowed to add any other gTLD labels that were

---

---

not originally applied for in that application in accordance with final recommendation 325 of the IDN PDP phase one final report. A base ASCII gTLD must exist and comply with the rules set under preliminary recommendation 1. Only an applicant for .brand TLD string who has applied for ASCII gTLD string and/or Latin diacritic gTLD string and is placed in a contention set is allowed to change the applied-for ASCII gTLD string and its Latin diacritic labels under the conditions set out in SubPro PDP Recommendation 28.

I just wonder whether it should be Latin diacritic gTLD string and in brackets S in the third last line, since those could also be more than one, but that's a small thing. Then ICANN... We received a comment from ICANN with respect to the sentence "the applicant is allowed to withdraw and apply for a gTLD label from that application." ICANN Org suggests revising the language to clarify that only the Latin diacritic gTLD string may be withdrawn as follows: "The applicant is allowed to withdraw and apply for Latin diacritic gTLD label from that application." So, sorry. Just looking. The next slide is to the topic, right, Saewon?

Okay, yeah. So, the rationale we provided there was in accordance with IDN PDP phase one final recommendation 325. The IDN 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 IDN 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 gTLD sets as outlined in preliminary recommendation 1.

Clearly, the withdrawal of an applied-for ASCII gTLD label that results in a single IDN 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. And I see Bill's hand. Please go ahead.

BILL JOURIS:

I wonder if for clarity we might want to say you cannot add another one to the set while that initial application is being applied for. So if you have a round in 2025 or 2026 and you do not apply for everything, if there is a subsequent round, you could then add new Latin diacritics to the group ones you already have. You just cannot add it during the current round, assuming that is what we meant.

MICHAEL BAULAND:

That is what we meant. It is just that once you submitted the application, you cannot add anything to that application. But in a new round, you can again add labels to that set. But it is the application where it is not possible to add anything after the window for the application closed.

BILL JOURIS:

I think it would be useful to say explicitly you can add something in a future round.

MICHAEL BAULAND: Okay. I think we have that in other locations, but if this causes or could cause confusion here, maybe we can still mention it here too.

BILL JOURIS: Thank you.

MICHAEL BAULAND: You are welcome. So, any thoughts about this comment? Could we go one slide back, please, so we see the comment again? Do we only want to allow the removal of IDN labels, so Latin diacritic gTLD labels? Or, we know that if the set exists, then the ASCII label also must exist. The question would be, what if the second-to-last label were to be removed? So, should this be possible at all to somehow dissolve this set and just have a single remaining label? And if that is the case, could the single remaining label be an ASCII label or gTLD label or LD TLD label? Or does it have to be the ASCII label? Any thoughts, comments? Bill, please.

BILL JOURIS: I feel like I am monopolizing the discussion. I do not see any particular reason why you should not be able to keep any member of the set. Why it matters whether it is the ASCII label or a diacritic label. Thank you.

MICHAEL BAULAND: Okay. Thanks. Amadeu, please.

---

AMADEU ABRIL: I think I am unmuted? For variants, there is a reason in which there is the base and the variants, but here what we are doing is relatively artificial in defining these bundles of ASCII and IDN. So I do not see an absolute pressing reason for saying that only one type of them must remain.

MICHAEL BAULAND: Okay. This is also, I see in the chat, Tapani also agrees with Bill and you. Sarmad, please.

SARMAD HUSSAIN: Thank you. This is Sarmad. So, I guess it is almost like two different recommendations or two different cases. One where you have multiple diacritized versions and an ASCII, and I think the understanding is that the ASCII must exist for the multiple diacritized versions to exist in the same set. And if that is the case, then the ASCII may not be removed, but one of the diacritized versions could be removed. So that is sort of part one or one version.

And then the second part or second recommendation is that if there is a situation where one is left with only a single diacritized version and a single ASCII version, that is the case we are talking about here where then either one potentially could be removed, leaving only one TLD. So it may be useful perhaps to consider two different parts of the recommendation or two different recommendations segregating those two versions. Thank you.

MICHAEL BAULAND: Okay, thanks. That is a good point. Maybe we should have two sections of this preliminary recommendation 6, which differentiates between the removal while still keeping the set intact, i.e., at least two elements remaining, and so to say, destroying the set by removing the second-to-last one. And in the first one, then it is only possible to remove LD TLDs, and in the last step, it might also be possible to remove the ASCII and keep one LD TLD remaining. Ariel, please.

ARIEL LIANG: Thanks, Michael. Just want to check whether my audio is okay. Yes, I agree with what Sarmad said and what you summarized, and I want to mention maybe in the ICANN Org comment we missed to point out what we said here is that the applicant is allowed to withdraw and apply for a Latin diacritic gTLD label from that application if that application is still intended as an ASCII Latin diacritic gTLD set. That's I think the part we kind of missed.

And also just for context, this is really a non-substantive comment, at least from my perspective. It's mainly to help future applicants to understand what they can or cannot do post-applying for a string, so that we can communicate to the applicant clearly what the limitation is with withdrawing a string. But I completely agree with what other people said, and if there is just one ASCII and one diacritic string in that set, then for sure the applicant could withdraw the ASCII string as well, and then that would just become an IDN application. So yeah, that is a little background on this comment.

MICHAEL BAULAND: Okay, thanks. Yeah. So, I think all who voiced their opinion are of the same opinion that it should be possible to remove also the ASCII TLD if only a single Latin diacritic TLD remains. We should probably just clear up the recommendations to make this more clear in which cases you can remove which label, and the suggestion by Sarmad seems a sensible one to have two cases here, with one being when there are at least two TLDs remaining, and the other one is dissolving the set with just a single TLD remaining. We can work on the wording offline and present a new version next week then. Looking at slide one now for some help and guidance. So the next one was this five, but that was only as a support for this six, right?

ARIEL LIANG: Yes.

MICHAEL BAULAND: Okay. I do not think we need that then since we all pretty much clarified this here. So the next one is this IG 8, where we state with respect to the evaluation of the information submitted per preliminary recommendation 7 in accordance with IG 3.6 from the IDN PDP phase one final report. There are now five bullet points so to say. First, the evaluation panel must include evaluators with Latin script expertise. The evaluation panel should apply criteria based on the general standard of reasonableness, and the criteria must be established during implementation. Third, consistent with recommendation 27.2 of the

---

SubPro PDP final report, evaluation scores on the question should be limited to a pass/fail scale, zero or one points only.

And fourth, the applicant must pass each element to enable the applied-for ASCII gTLD string and/or Latin diacritic gTLD labels to proceed to the next stage of the application process. And fifth, the evaluation outcome of any one applied-for Latin diacritic gTLD label should not impact the evaluation outcome of any other Latin diacritic gTLD labels in the application, including the ASCII gTLD string. Just as a side note, I was stumbling because it's called ASCII gTLD string and Latin diacritic gTLD labels. Do we have any reason why we use string and label and not the same word for these TLD labels? Maybe we can also discuss this offline. Ariel, please.

ARIEL LIANG:

I think I remember this. The string is used when we refer to the... again, this is from the IDN PDP. So the string was used when we refer to the primary string, and the label was used when we refer to the variant label. So that's the way the IDN PDP tried to differentiate these two things. But then again, when we get to the actual application implementation for these, we just call them strings. Like applicants will apply for a primary string and variant strings. So when you get to the next round or 2026 round implementation, it's all strings. But then in the IDN PDP context, it was string versus labels.

MICHAEL BAULAND:

Okay. So maybe we change it also. Also "string" is the way forward, but yeah, that is not the main point here. The question here is we have

---

coming from ICANN Org, and first is there is a need to address the impact of an evaluation outcome if an applied-for ASCII gTLD string in a set fails. Compare PR 1.4 and PR 36 address ASCII gTLDs only in the post-delegation scenarios. And two, there is no LDPDP output corresponding to EPDP IDN's final recommendation 3.1, which states an application for an allocatable variant label cannot precede an application for that variant label's primary gTLD string.

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. Okay. Do you, Ariel, want to supply some background information for that or anybody else has some thoughts or comments?

ARIEL LIANG:

Sure. So, I think the comment itself is pretty clear, I guess, but maybe just a quick context from the IDN EPDP recommendation. I think there is one very important principle for the IDN EPDP is that the primary is like the key to hold the set together. I think it is called the integrity of the set. And then if the primary string does not pass evaluation, then the entire applied-for set will fail as well. So that is pretty clear when we implement this recommendation, like we know in the new gTLD program how to deal with that.

But then for the ASCII and Latin diacritic set, it's not very clear to at least ICANN Org if the ASCII string fails evaluation, what happens next? Even if we do understand there is this kind of a guiding recommendation that the set will cease to exist, but then one of the diacritic marks can still go

ahead and be applied for as the IDN string. So we just want to make sure we have a clear understanding in terms of next steps in the evaluation phase. Like, if the ASCII string for some reason fails any part of the evaluation and cannot proceed, what to do for the applicant? Do they have to apply, like file an application change request to basically remove everything in the set and just propose one diacritic string to proceed with the evaluation, or what else? We just want to seek some clarification on that.

And also, sorry, I just wanted to quickly mention, I think this comment is related to some other comments we made, especially in the context of objection, if I recall clearly. So for example if the ASCII string was being objected to and the objector prevails, then what happens? For IDN EPDP, they do have a recommendation on that regard, in a similar kind of context. But then for the Latin Diacritics working group, the PDP does not have an equivalent recommendation. So that would be helpful to have a clearer understanding on this.

MICHAEL BAULAND:

Okay, great. Thanks. That helps a lot and sounds reasonable. So the question is basically we do not really say clearly what happens in the case the ASCII gTLD cannot go forward for several reasons, either because it fails a validation or there's an objection or whatever, and the question is how to deal with the remainders of the set. Any comments or thoughts? Amadeu, please go ahead.

---

AMADEU ABRIL: Oh, sorry. I was waiting. Okay. I understand what you are trying to do here, but really, can anybody imagine a case in which this may happen? That one of the strings, let's say the ASCII string, faces a successful objection for any reason, be that public order, similarity, whatever, and the same thing with the diacritic does not? I simply cannot imagine that.

MICHAEL BAULAND: You have to work on your imagination, Amadeu. But yeah, I agree that it's quite an unlikely case, but still we should have rules that are also clear in corner cases. So for that reason, I tend to agree with Ariel that we need to have a decision if that case exists.

AMADEU ABRIL: Then let me submit humbly that if somebody has applied for this as a set, and the set falls apart because there is one unavoidable impediment on one of the strings, perhaps the whole set should stop just in case. Because probably if only one of the strings has faced that impediment but not the others, probably it is because of a mistake of the objection or the evaluator. So we would be safer on that side. Even if I repeat, I cannot imagine a real case.

MICHAEL BAULAND: So just to be clear, you say that if any of the strings does not pass the evaluations, the whole application should stop.

AMADEU ABRIL: Yeah.

MICHAEL BAULAND: Because that would be in contrast to what is currently the case with variants and what we are currently saying if a variant string is failing, then the rest of the application can go forward. And we also say that if an LD string fails, the rest can go forward. Are you proposing to change--

AMADEU ABRIL: Well then, I withdraw my proposal, as I repeat, I do not think this is seriously realistic. Probably variants will be different because the kind of reasons why things are variants is different from what we are discussing here for Latin diacritics. But one way or the other, this won't happen. But I agree, it's good that we have a solution.

MICHAEL BAULAND: Okay, thanks.

AMADEU ABRIL: Whichever. I do not care. It won't happen.

MICHAEL BAULAND: Okay. Good point. Don't care about things that are so unlikely to happen that they do not bother you. Sarmad, please.

SARMAD HUSSAIN: Yeah. So this is Sarmad. Though in this case it becomes more consistent with the case with variants where if for some reason the primary cannot

---

move forward, the variants cannot move forward either. But then it does in some way start conflicting with the case we just discussed where we said that if there is one ASCII and one diacritized version and the ASCII is withdrawn, the diacritized version can move forward. But in this case we are saying that it cannot. So I think that is something to consider, the previous discussion and this discussion to see whether we do think that those are still consistent to different versions, or if they are not, then how do we resolve that?

MICHAEL BAULAND:

Yeah. Thanks. Good point. Any thoughts, comments?

SARMAD HUSSAIN:

Yeah. Just a follow-up on that. The reason this inconsistency is coming in this case but not in the variant case is because in the variant case, the variant can never move forward without the primary, but in this case, we are potentially saying that the LD version in some cases can move forward without the ASCII version. So that is probably causing some challenges. Thank you.

MICHAEL BAULAND:

Thanks. Yeah. That is the main difference because the variants, they depend on the primary. But our LD strings, they are kind of independent and only as a requirement we have that there must be exactly one ASCII version in a set. So one solution forward would be to try to be consistent with the previous one we looked at and say that if a diacritic

---

version is found invalid, then this can just be dropped and the rest of the set can remain.

But in case the ASCII is found invalid, then the applicant, if they still have at least two diacritic versions, they have to choose one which would be able to go forward as a standalone application. So this would then make it similar to the previous version where we say you can only drop LD TLDs as long as the set still remains and the only chance to drop an ASCII TLD is by dissolving the set and keeping a single LD TLD. Would that be a way forward here too? Any thoughts? If not, then any other suggestions how to handle this? Or go with what I said and don't care about that case because it is unlikely to happen and let someone else deal with that. Or is my internet still working? Is anybody hearing me? Oh, yeah.

JULIE BISLAND: We hear you.

SARMAD HUSSAIN: I think on your last comment on whether it is unlikely, maybe it could be unlikely, but I do not think we have clearly indicated that it is not a possibility. So let us still perhaps consider it or otherwise we need to discuss that other aspect and close that. Thank you.

MICHAEL BAULAND: Thanks. Yeah. I agree.

---

AMADEU ABRIL:                      Sorry, Michael. May I say a last thing on this one?

MICHAEL BAULAND:                Yes. Sure.

AMADEU ABRIL:                      Okay. Just to respond briefly to Sarmad's argument on the coherence between these and the ability for the applicant to withdraw one of them. I think it is very different. One thing is, can the applicant change their mind and consider that they do not need one of them? The answer is probably yes. The other question here is there is a *raison d'être*, some kind of public order within ICANN terms, reason not to proceed with one of the strings. Be that GAC advice, reserve list, similarity objection, whatever, failure to evaluation in any part.

And what we are saying is, well, despite this, can the applicant still go somehow around with something that by definition is very similar and has defined it as equivalent in the application itself? We want this because they are equivalent, and here I am a little bit more reluctant. But having said that, if we want to be simple, probably we should adopt the same thing as the IDN PDP even if the situation is different. I completely agree that it's different, and I said that initially, but let's be simple. I do not think we should waste ten years in discussing something that simply won't happen.

MICHAEL BAULAND:                Okay. Understood. So, your suggestion would be to be fully consistent with the IDN PDP and say that if an LD TLD is found invalid, that then

---

this can simply be dropped, but if the ASCII TLD is found invalid, then the whole application has to be dropped and cannot go forward. Right?

AMADEU ABRIL:

Yes. If the... regarding the ASCII, the difference between IDN and ASCII, if the IDN is that for technical reasons, yes. If it's for any other kind of reasons, GAC advice, whatever, objection, similarity objection, legal rights objection, then I think that the full set should stop. Because otherwise, we are leaving a door open to make these decisions moot.

MICHAEL BAULAND:

Okay. Understood. Bill, please.

BILL JOURIS:

I am not sure if I am understanding Amadeu correctly, I am not sure I agree. Bear in mind, I am not totally awake. I can imagine a situation where you have an ASCII and several diacritics in a set that someone has applied for and the similarity review panel says, "Okay. This one particular diacritic version is too similar to something in another script," for instance. Does that mean the entire application gets rejected, or does it mean it just loses that one diacritic version? My initial take is they should be able to lose just one of the diacritic versions without losing everything, but maybe I am overthinking this. Thank you.

MICHAEL BAULAND:

No, that's correct, and I think that's really what we state here in 8.5, which says the evaluation outcome of any applied-for Latin diacritic

---

gTLD label should not impact the evaluation outcome of any other Latin diacritic gTLD labels in the application, including the ASCII gTLD string. So, I think that case is covered here, that if an LD TLD fails the evaluation, the rest can continue. I think--

BILL JOURIS: Sorry for my misunderstanding.

MICHAEL BAULAND: That is all right. I think the open issue, and that's what ICANN and Ariel mentioned here, is the case if not the LD TLD fails the evaluation, but the case if the ASCII TLD fails the evaluation. What should happen?

BILL JOURIS: Well, in that case, again, the whole application has to fail because it won't have an ASCII version to anchor to.

MICHAEL BAULAND: Okay. And that would be then in line with the variant case where we say that if the primary label fails the evaluation, the whole application fails. So we want to go this way here too and say if the ASCII fails, everything fails and the whole application has to be dropped.

BILL JOURIS: So perhaps we want an 8.6 to say that explicitly.

---

MICHAEL BAULAND: Okay. Any agreements, disagreements? I see Tapani's comment. He asked, "If there are exactly two TLDs in the application, ASCII one, LD one, and only the ASCII one fails, can the LD one proceed?" According to what we just discussed and what Bill suggested, then no. The LD cannot proceed because the whole application fails if the ASCII fails. Tapani and then Saewon, please.

TAPANI TARVAINEN: So why is this case different from the one where there is one ASCII and one diacritic version and the ASCII one is pulled? In that case, the diacritic one can proceed alone, but in this case it cannot. Did I get that correctly? And if so, why?

MICHAEL BAULAND: Yeah. Good question because couldn't they just voluntarily drop the ASCII one at the moment it fails and then stay on with the LD one? Saewon?

SAEWON LEE: Sorry, Michael, I did not get that question.

MICHAEL BAULAND: If I understood Tapani correctly, he is asking why we want to make the whole set fail if the ASCII validation fails, because in the previous version, we had the option that the applicant can voluntarily drop the ASCII version and keep the diacritic one.

SAEWON LEE:

So I do keep on referring back to preliminary recommendation 5 in how we treated the bundling of the set when we applied. Just to show you the slide. If you see the diagram in the left, at the time, the working group had agreed that when we apply as a set, basically as this is an exception process, we're bundling the set and we cannot unbundle in the middle or we are not allowing the unbundling of the set where we allow the applicant to change course in the middle. And obviously this is a little bit different to the evaluation process, but basically that is what we had agreed upon.

So, the working group had agreed that this would not be allowed where, based on our exception process, based on the unbundling of the set, either of the strings would be allowed to move forward. And that is what we had agreed upon for preliminary recommendation 5. So that's kind of what I could answer to that at this moment. But just related to the reason that I had put my hand up was just to go back to what Bill was mentioning. I kind of lost my train of thought on that, but basically...

Oh, so if you go back to this slide, based on adding another point on implementation guidance 8, that is basically what ICANN Org was raising in point two, where we do not have a corresponding recommendation to 3.1 about how we treat or how we allow the ASCII without the presence of an ASCII label that a Latin diacritic label can proceed. So if you see in the bottom the final recommendation 3.1, if we do want to add, if we do agree that we're basically treating the LD set as how it was referred to in EPDP IDNs, the ICANN Org was asking that we go back to

---

that recommendation to make a corresponding recommendation. So either we can add it to this implementation guidance 8 as 8.6, or we can make a new recommendation where we set this in the beginning so that this is clear as to how we treat the ASCII and LD label during the application process and evaluation impact. I hope that adds to the context.

MICHAEL BAULAND:

Okay. Thanks. Yeah. Maybe go back to the slide with PR 5. So I think maybe we might need to revisit what we meant by this. So we said that if you do apply for TLDs, you have to decide at the beginning whether you want to use a standard process, i.e. apply for each TLD separately and then go into string similarity review and potentially contention with only one string being able to go forward, or whether you want to use the exception process and apply for the TLD strings in a bundle, in a set, and then have them evaluated as a set.

And we said that it is not possible to switch between those two processes. The question is now, does this just mean that once you started with a set, you cannot separate them into two separate standard process applications and vice versa? Once you started with two separate standard process applications, you cannot combine them into a set, or does this also cover the case when there is only a single TLD left in the set? Then this kind of also becomes a standard process, single TLD application, but it could also be viewed as a set with just one element application, although no, not really. So did we want to prohibit this too or... Sarmad, please.

SARMAD HUSSAIN:

Yeah. So in the previous case then we were discussing where we had potentially considering a separate clause for the case where there is one Latin diacritic string left along with the ASCII, and I think the group was discussing that then either one could be dropped. Perhaps if we are very strictly following this criteria where we are using the set formation and not independent string formation, and we're saying that once you've chosen this, then you must stay on this path as an applicant.

Then in that case, perhaps the choice should not be there that they could drop ASCII or Latin diacritic, but it could be that they could only drop Latin diacritic or withdraw the application, but they can only possibly proceed with ASCII, but not with the Latin diacritic version. And I think then that would make it probably more consistent with the current discussion as well. So that is another option. Thank you.

MICHAEL BAULAND:

Yeah. Thanks. That's exactly the big question, so to say, which would then also solve the problems we have been discussing with the preliminary recommendation 8 and the implementation guidance 8.6. The preliminary recommendation 6 and the implementation guidance 8, this all falls together to this question. Should it be possible to drop an ASCII string and just proceed with a single LD string? We could go back to the different approaches, the next slide, I think. Yeah, that one. Thanks. So any thoughts?

Basically the two solutions. Solution one is we say that you can drop any string differently. So as long as a set exists, you have to have one ASCII

---

gTLD. But should it be possible to dissolve the set and keep a Latin diacritic TLD and continue with that single TLD onwards, or should that be disallowed and once you start with the exception process, you have to keep the ASCII one because otherwise the set... Amadeu?

AMADEU ABRIL:

Look, in the process, you cannot change from a standard to community or from community to standard or from standard to brand or back. Indeed, you can fail the brand evaluation and you become a standard. You can request an exception to the Code of Conduct and fail and go back to standard. But you cannot change the type during the evaluation. It is not forever. Afterwards, you can do everything, even give up your TLD or turn it into community or change from community to standard. We have had examples of that. But after the process of evaluation and when you get to contracting.

So I do not think that during the evaluation process, which I think is what we are discussing here, that is, the period for the application when this is still application, I do not think this should be changed. Being congruent with the other kinds of types and situations that cannot be changed either.

MICHAEL BAULAND:

Okay. Understood. Thanks. So your suggestion is to have the policy that once you start with the LD exception process, the only TLD labels you are allowed to drop are Latin diacritic labels, but you can never drop the ASCII label because that always has to exist. And so it is not possible to

---

drop the ASCII label and continue with a single LD label. That will make it consistent with the variant case.

To Tapani's question, yeah, that would be possible. If you drop all of the LD TLDs, you could still continue with the ASCII one, because then you have a set with just a single ASCII TLD in it, and that would be okay. Yep. Set of one is still a set. Okay. So everybody agrees with that, that once you start with the exception process that you cannot switch to a standard process, and as a consequence you have no ability to drop the ASCII TLD, just like in the variant case you have no ability to drop the primary string. Any objection or any agreement?

Since this is a very important decision, I think we really should make up our minds here. Maybe have a quick thumbs up or green or red thumb if that is possible. So, everybody in favor of staying with the set and not being able to drop the ASCII one, please raise your hand or put a green thumb up or whatever. Okay. That's you in favor. Yeah, and okay, just see Saewon's comment whether a single TLD can be considered a set, because that is in contradiction to preliminary recommendation 1, where we said a set consists of at least one ASCII and one diacritic version, I think. Bill, please.

BILL JOURIS:

I was just putting my hand up because when I had put my thumb up, it disappeared. So, I was trying to make it easier for you to count. Nothing else.

---

MICHAEL BAULAND: Oh, okay. Understood. Thanks. So, Asteway, is this a raised hand or just...

ASTEWAY NEGASH: Yeah, that's a raised hand. Thanks, Michael. I am just wondering if there is a situation in reality where the ASCII gTLD might be found inappropriate or it might be rejected by the evaluation panel or something, because in the first case I would understand if the applicant actually pulls out of the application, then the ASCII cannot be pulled out, so the ASCII sustains the process. But in this case, I think we might need to rule out this situation where the evaluation panel can actually drop the ASCII itself in contrast to any LD TLD domain. Thanks.

MICHAEL BAULAND: Yeah. Good point. So, seeing that we only have fifteen minutes left, maybe we revisit this next week and discuss this internally again, and have some examples and some graphics available that make it easier to see what the options are and what the consequences are. And maybe in the last fifteen minutes, we can still visit the open issue discussed on the mailing list where Tapani interjected that possibly our definition of what is a diacritic might be too strict. Should we maybe switch to slide 28?

Thanks. So up to now we had this definition of a diacritic based on the Unicode. Go to next slide, please. And where we said that we only consider those characters which in Unicode can be decomposed into an ASCII character and a combination of one or more diacritic characters. So on the right side you see Latin small letter C with cedilla. This can be decomposed into the 0063 C ASCII letter and 0327, the combining mark,

---

cedilla. And that's a diacritic but, for example, the O slash, which is not decomposable but still called Latin small letter O with a stroke, that is not in scope.

And one slide back, please. And Tapani argued that this is maybe too strict based on the definition of diacritics. The scope is too restrictive by limiting the characters to those that are only decomposable and possibly add those which have the name like Latin small letter ASCII with something. Bill, please.

BILL JOURIS:

Just so I am clear, when you say the non-decomposable ones are out of scope, is that the scope as we have defined it, or is that the scope as we were handed by the GNSO Council? In other words, do we have flexibility to change that, or are we in a position where the only thing we can do is go back and ask if our scope can be adjusted?

MICHAEL BAULAND:

Good question. No, we actually decided that, and we can revisit this. So, the problem is we have the requirement to only look at diacritics, but it's difficult to find a unique or universal definition of what a diacritic is. And that's why we looked at the Unicode, and Unicode has certain code points which are called diacritics, and we said that only those code points which are a composition of these Unicode diacritics with ASCII letters, these are in scope. And others which are not decomposable to such a diacritic letter are out of scope. But that was not handed down from the GNSO, that was our definition of diacritics. Do you have a follow-up, Bill?

**BILL JOURIS:** Yes, I do, actually. It seems to me that we could equally well say if the name of the character is "Latin small letter whatever with," then it's a diacritic for our purposes. And in fact, I think that would be easier for users to understand because they do not have to go digging around to find out what combining diacritics are. They can just look at the name and say, "Right, that says 'with'. I can include that." So I think that would actually be easier for the users to understand as well. So my inclination is to go with the expansion. Thank you.

**MICHAEL BAULAND:** Yeah. That is what Tapani suggested, to look at the label "with" or...  
Yeah. Mark, please.

**MARK DATYSGELD:** Thank you, Michael. So on the way back from Mumbai, I had a 24-hour flight back to Brazil, and I decided what better to do than to write a 10-page document describing our trajectory and character choices without internet. So I mean, that tells the kind of people you guys have chosen to lead this group. Right? So in that document, any of you who had the doubtful pleasure of reading it, I go over why we decided to take this interpretation, this Unicode-based interpretation. And the essence of it is that it's not that we do not think there is a good option, it's that the problem is keeping it testable and keeping it bounded.

So how do we define those boundaries without necessarily relying on the one convention that we established? So whatever we decide to do

---

moving forward needs to still be testable. So on the list, Tapani raised a point and I-- as I was saying, I was coming up with something in the background here to share. Sharing is not turned on. Could you turn it on for me, any of the staff, so that I can share my screen? Perfect. So I came up with the list of what would be the additional characters using this theory of "Latin Small Letter With." So we would be adding with that theory twenty new characters or glyphs. I also sent it to the mailing list.

I do not know if the mailing list delivered it yet, but as you can see, it would add some of the characters that we expect and possibly some that we do not. But still, there is a clean implementation path. Were we to go with that, Latin Small Letter With is viable within the scope. And I updated R2 to be able to test different theses that we might have so that we do not have downtime between things that we might want to consider. So just to recap, this keeps the thesis that we had initially, which is a testable, character-based ASCII plus diacritic.

And this does not exactly fit the model "Latin Small Letter With," just to follow up on Bill, because this is different. That really is a composition. We're really looking at composed characters. Meanwhile, here we are choosing an arbitrary subset that is named "Latin Small Letter With." And that does exist, but it is a subset. So what we could do, were we to go that route, is add both one and the other, or choose a subset of these. For example, I think that the least controversial ones are "with stroke." And "with hook," we create some weird situations like the F that we use for some mathematical equations. That one might be on the weird side.

So maybe we decide, okay, we're going for Latin Small Letter with Stroke as our additional subset and call it a day. Well, all of that depends on the group. I just want to make it viable and easier for us to discuss these things. So for now, that's what I have to offer to the group in terms of how do we move this forward without having gigantic downtime, just trying to give options for us to look into this in a more programmatic way. Thank you.

MICHAEL BAULAND:

Thanks, Mark, and thanks for this quick turnaround and, of course, the time spent on the flight. Very valuable. Just a small comment which Tapani also made. The restriction would be that a small letter must be an ASCII letter. So the case where the small letter is an EZH, that could not be included, even though it's a combination. The first one must be an ASCII because otherwise there is no ASCII reference for the main label. Sarmad, please.

SARMAD HUSSAIN:

Thank you. This is Sarmad. So normally when we are processing strings, we are using Unicode data files, and we are using generally the general category properties for characters which can label them as letters or marks and so on. It's an interesting idea to parse the name of the character with the string as you were suggesting. But I am just wondering whether the existing property structure of Unicode characters could be utilized in this case as well from a programmatic point of view. Sure, we can certainly parse the names of the characters, but generally that is not the practice. Thank you.

---

MICHAEL BAULAND: Okay. Thanks, Sarmad. And with just a few minutes left, just a final comment. The F with the hook, as Pitinan mentioned, would not be actually an issue because that is already a variant of the regular F, and therefore it is out of scope of our PDP anyway, so it would not cause any issue. But yeah, I would like everybody to think about this, about the suggestion by Tapani and the compiled list by Mark and also the comment by Sarmad to possibly stick with Unicode properties rather than parsing character names. And we should continue this discussion next week and with that, I hand over to Saewon.

SAEWON LEE: Thank you, Michael. I wasn't expecting to finish off. Actually, there's nothing really to highlight for next steps, but I do want to focus the working group's attention to the link that I am sharing in the chat. So basically, staff and leadership in the background will be updating the draft language as the working group continues with the discussions. So I checked that there were no input or comments, which I am guessing that it was all fine. But if you haven't had the time to check the draft language from last ICANN 85's discussion, please do review this and provide your input if you disagree. It is also in your mailbox. So, also please do go and review this. Other than that, we will just see you next week.

MICHAEL BAULAND: Thanks, and sorry for being slightly over time, but yeah. Talk to you next week. Continue the discussion. Thanks and bye.

---

JOHN EMERY: Thank you.

JULIE BISLAND: Thank you, Michael. Thanks everyone for joining. We will go ahead and stop the recording now. Have a good rest of your day.

**[END OF TRANSCRIPTION]**