
ICANN70 | Virtual Community Forum - Remediating Universal Acceptance Issues
Thursday, March 25, 2021 – 09:00 to 10:00 EST

MOHAMED ELBASHIR: Hello everyone and welcome to the universal acceptance remediation session in ICANN 70. My name is Mohamed Elbashir, ICANN staff supporting the universal acceptance community and Universal Acceptance Steering Group which is overseeing the activities of the community related to universal acceptance.

The session today will focus on the identified technical gaps and issues related to the universal acceptance- of all domain names and e-mail addresses using applications and systems on the Internet.

Myself and my colleague, Sarmad Hussain will be monitoring the chat for your comments and questions, so if you have any comments that you would like to be read out, please write it on the chat and write comment between parentheses. And if you have a question, also please write in the chat and indicate it's a question by putting the word "question" before and "question" at the end of it between parentheses. Thank you very much, and I would like to start and the first speaker will be Dr. Ajay Data, chair of the Universal Acceptance Steering Group, followed by the universal acceptance working group chairs. I hand it over to you, Dr. Data.

AJAY DATA: Thank you, Mohamed, t yes, everyone, to join this very important session by Universal Acceptance Steering Group. Good morning, good

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.

afternoon, good evening. As we all know, Universal Acceptance Steering Group is a very important initiative by the community to break the language barriers. Next slide, please.

this group is focusing only on a very small vision that all domain names and all e-mail addresses work correctly in all software applications. It sounds simple but there's a lot more which goes into which my colleagues are going to take up in the course of this entire session.

The mission is very simple. The mission is to mobilize all application developers, in fact other stakeholders too, to get their products UA ready by providing encouragement, documentation, case studies, test suites, tools and measures to help provide a better user experience to the end user.

We have a structure which my friend Maria is going to take you through to address UA, community and activity engaged and participating in the UA local initiatives and dedicated working groups.

The leadership structure is known as Universal Acceptance Steering Group. We have elections every two years, and community selects the leader and people volunteer to lead that effort globally. The impact is very simple: to promote consumer choice, improve competition and provide broader access to the end user. In a nutshell, domain names and e-mail address choice is provided to the end user and they can pick up a long top-level domain or short top-level domain, or script-based name, whatever they like to. Next slide, please.

There, I would like to invite Maria to come and share this working group structure. Maria is the communication group chair.

MARIA KOLESNIKOVA:

Thank you, Ajay, very much. Yes, I will try to present UASG. Okay, so you know that Universal Acceptance Steering Group is a community-based initiative and it works with support of ICANN staff. So the structure of Universal Acceptance Steering Group-is quite simple. Now it has six working groups with their separate focus and scope of work. They are technology working group, EAI working group, measurements and communication working groups, and also, two working groups focused on local initiatives and work with UA implementation procedures and issues in their own countries. This is local initiatives working group and UA ambassador working group. So you see the scope of their work on the screen. Next slide, please.

The main work of UASG is based on this UA remediation circle. Mainly, we all are looking for what technologies still have gaps in universal acceptance implementation, then look at how to evaluate them and based on this evaluation, develop guidance and recommendations, if necessary, training for their owners and try to solve the issues and to clear these gaps. Next slide, please.

And to better understand what universal acceptance is about, I hope that everyone knows what we are talking about but still, practice makes perfect. So universal acceptance-readiness covers such categories as all domain names that currently exist today. So that means new gTLDs,

it means letting long new gTLDs and internationalized domain names there currently exist, I repeat, and we need to think about this.

And also, of course, e-mail addresses based on these domain names. You know that e-mail addresses have two parts, and the main part is exactly what I'm talking about, TLDs and second-level domain names based on them, and also, e-mail addresses have local parts and this is quite a separate issue because now the local part can contain Unicode symbols and it requires system administrators to think about this and make special settings in e-mail servers and services to correctly support this. Next slide, please.

Here, you can see the technology stack. You see the application services and standards that are connected to the support of domain names and e-mail addresses that process these identifiers. For this reason, Universal Acceptance Steering Group works with all these technologies and tries to improve them to make universal acceptance ready.

So I won't list all of them, but just some such as social media, search engines like operating system and system tools, of course websites and different applications, and standards and best practices. Next slide, please.

So, you see that there are a huge number of different software products that need to be overviewed and made universal acceptance-ready, and now we see that the current level of universal acceptance readiness quite low and there are only about 10% of e-mail service data configured to really support e-mail addresses in local languages and only about 11% of websites of—the evaluation was provided for 1000

global websites, but only 11% of them can support e-mail addresses in local languages. So this is what we also work on together and need to improve. Next slide, please.

To understand how UASG works, you can see some reports and projects that were completed recently, and I hope that the following speakers will cover all these issues and the subject of these reports and projects. And some of these projects are listed here, and they are devoted to evaluation of current level of universal acceptance complaints and EAI complaints in, for example, programming languages, libraries and frameworks and e-mail software and services, and also, they were provided some recommendations for system administrators how to solve issues with naming, e-mail address internationalization, for example, and also some popular questions based on using different programming languages and e-mail tools to develop new applications and software. Next slide, please.

And also, UASG of course will continue to work on all these issues, and for this year, there are planned different kinds of projects, and some of them already started. They're aimed at evaluation of different categories of software such as CMS, content management systems, social media networks, some standards and best practices, and new evaluation of browsers and how do they now support universal acceptance categories.

So all these projects are ongoing or planned for this year, and I think that maybe next ICANN meeting, maybe we'll report on how the process is going on. Next slide, please.

This was quite short overview of Universal Acceptance Steering Group and universal acceptance initiative, and I only want to add that besides these six working groups, the universal acceptance includes more than, I think, 500 participants that are subscribers to our common mailing list, but of course, there are much more people around the world who are involved in this initiative because not all of them are in our mailing list, but they are really involved in these issues and help all of us to move forward this universal acceptance remediation process.

So if you have any questions in chat, Mohamed, I can answer or comment, or our panelists. If not, I will provide the microphone to our next speaker.

MOHAMED ELBASHIR: We have a question in the chat. Sarmad, please.

SARMAD HUSSAIN: Thank you, Mohamed and Maria. We have a question from Keiron Tobin. The question is, has any work been done in relation to Punycode? If so, for accents on domains, etc., is this something that you are also working towards?

MARIA KOLESNIKOVA: I invite maybe Satish to comment on this.

MOHAMED ELBASHIR: Satish, if you're speaking, you may be muted.

SARMAD HUSSAIN: We have a hand raised by Mark. Mark, you want to come in in the context of this particular question?

MARK DATYSGELD: Thank you very much. In relation to that, a lot of work has been put in a document that I linked in the chat. It's called considerations for naming internationalized e-mail boxes. Even though it's very directed at e-mail boxes, at the same time, it deals more holistically with questions of how different characters and different provisions for UA work. So in the context of this question, I would suggest heavily that you have a look at this document. You'll probably find it rather interesting. And we do explore some limitations and recommendations and have a general look at the state of things right now. So that would be a good place to start. Thank you.

MARIA KOLESNIKOVA: Thank you. I hope that the question was answered. So then I will provide the microphone to our working group chairs. The next one is Mark Svancarek, the chair of EAI working group. Mark, please.

MOHAMED ELBASHIR: Maria, it seems Mark is on his way to join us from the EPDP session.

MARK SVANCAREK: I have arrived.

MOHAMED ELBASHIR: Okay, great. Please go ahead.

MARK SVANCAREK: Sorry. I was being pinged and I missed my pings. So, hi, everybody. I'm Mark Svancarek, the chair of the EAI technical working group, and here's a little bit of an update on EAI. So our test methodology is published online. One of the things that we do is we check the acceptance of e-mail addresses in various websites and services. And here's a summary of this.

I think you can see that from 2017 to 2020, there has been a steady increase in support across the board. We've split this into various categories from most complex, being .arabic@arabic—notice that's left to right instead of right to left—then all Chinese all the way down to ASCII e-mail addresses which have either new TLDs or new long TLDs. So that would be the least complex e-mail addresses.

And for those last ones, new TLDs which remain ASCII, you can see the support is pretty decent, above 90% for the short ones, around 80% for the long ones. That's not great, but it is steady improvement. You can see that the most improvement was achieved in the first year, but he improvement continues.

Arabic remains challenging. Surprisingly, Chinese is a little bit challenging too. I'm not really sure what's causing that, but that's what our statistics are telling us. You may notice that just as chinese@ascii.ascii have a little setback when we measured this two

years ago, `ascii@chinese.ascii` has had a little setback this year. We don't know what's causing that. Hopefully it's some sort of an outlier rather than some systemic trend, and when we measure it again, we'll see if that reverts back to the trend and starts to show improvement again. Next slide, please.

When we talk about these tests, you may wonder, what is it we're testing? Isn't an e-mail stack fairly complicated? And it's true, the classic e-mail stack has a lot of components. So when you look at the diagram below, you can see that there's a user agent and there's a user sitting next to the user agent, and then all sorts of other components that the message has to flow through. A user agent, submission agent, transfer agent, delivery agent, service providers.

When you look at our results, you'll see that some of the providers provide an entire stack, or they may provide only portions of the stack, individual components. And that's okay, it just means their test methodology needs to consider how a single component could be tested in an entire stack. So if you read our test methodology, you can understand how that works. Next slide, please.

So here's a summary of top systems and components, starting with Coremail, Outlook.com, Yandex, etc., Courier, Gmail. Some of these are web platforms. Web platforms are not broken into individual components, they're taken as a holistic system, and so that impacts the way that we test some of these things. And then of course, we also have individual servers like Exchange servers or Postfix that have to be measured in isolation. So when you see these scores, the reason that

it's not an entire row of scores is because the server is just providing an MSA/MTA function. Hopefully, this chart makes sense. You can see that there's decent support across the Board. Still a lot of pink columns, still a few red columns, but we have had increasing interest in this over the last year with some things. Most noteworthy would be Apple rejoining UASG after a few years, and you can see that their e-mail solution is now edging towards the sort of compliance that we'd like.

I think that's it for me on e-mail addresses. Take it away.

MOHAMED ELBASHIR:

Thank you very much, Mark, for the update regarding e-mail systems UA readiness. There's a question for you in the chat.

MARK SVANCAREK:

Yes, we also have statistics on less complex expansions, for example, Greek, Cyrillic or non-ASCII Latin glyphs. We don't really consider those to be less complex, so if it's containing Unicode codepoints, it will still fall into that category of new labels in the local part and IDNs in the domain part. So we lump all of those together. We're not making a distinction necessarily between Greek or French or Cyrillic. It's just that Chinese seems to be an outlier and so we're measuring that one separately. For the other ones, we're not maintaining separate statistics, although I suppose if you're interested, we could figure out a way to test those things and get some statistics. As I said, the test methodology is published, so it is possible for someone to duplicate

those results and submit those back to the working group. Does that help?

Also, Mark is pointing out that there's the open dots. Japanese also has an open dot, but we're not seeing it there, really. Chinese is where you'd see it more often.

MARIA KOLESNIKOVA: If possible, I can add something, that based on this evaluation, we provided some separate evaluation for Cyrillic script, and I can confirm that this is more or less the same as Chinese script. So this can be shared for other Unicode symbols and scripts.

MARK SVANCAREK: So it's a good question, and I hadn't really considered it before because I was, in my mind, lumping these things together. I think it would be a fun project for the working group to look at, how that breaks down script by script. And I would hope that it was not too significant a difference from one script to another, Chinese and Arabic being the outliers. But if Cyrillic is looking more like Chinese, you probably should call it out.

MARIA KOLESNIKOVA: It also depends on the country where you provide this test.

MARK SVANCAREK: Yeah.

MARIA KOLESNIKOVA: So, okay, our next speak will be Dennis Tan Tanaka. He is a chair of universal acceptance measurements working group. So Dennis, please, the floor is yours.

DENNIS TAN: Thank you, Maria. So continuing in the measuring and efforts, this is one pertaining to open source code. So this is one of the projects that the measurement working group has done in the past two years almost, and this one in particular is building on previous work, a multi-year effort into trying to understand the successes and deficiencies related to UA when it comes to programming libraries and applications.

So again, I said this is building upon the understanding of what UA readiness exists or does not exist in programming libraries, and that's one part. This work went into open source repositories such as GitHub looking for applications using the Java and Python programming languages and looked how those programming libraries were used.

So we're going from the actual components, whether they're UA ready or not, and then going into how applications are using those components. So the main objective of this project was to evaluate looking for signatures of domain name and e-mail evaluation procedures and see how they stack up against IDNA2008 validations or using other methods that are or aren't suited for UA readiness.

Mark Datysgeld is one of the authors of this project, so I welcome him to chime in whenever necessary here. But I'm just giving you a glimpse

of what this study did here. So basically, the group went into looking for these signatures in all the applications in the GitHub repository, again, Java, Python [inaudible] to keep the scope of the work manageable, and looking for those signatures. And on the next slide, we can see what he found.

So again, I'm showing a simplistic view of the work, but once this is published, I will encourage you to look at the report because it's a very thorough analysis of the repository and the findings and the learnings there. But what they found in Java, that 7% of the projects have an IDN library or validation method. That in itself is not bad. It is not saying that the rest 93% of applications is bad, just that 7% has an IDN library or a domain name validation procedure. Next slide, please.

And this is the most important finding of all. I was trying to convey here, and you see the picture here, of that 7%, the vast majority of validation procedures use a regular expression. In our view, the UASG view, this is not the best way to validate a domain name, especially when it comes to a Unicode domain name, because regex can do so much.

And so that limits really the ability of an application to really do good validation of a domain name, especially when you talk about IDN domain names or new gTLDs in general. So you see this table that is divided into different segments, and the [least] one is the IDNA2008 or IDNA2003 libraries. IDNA2003 is not ideal, but at least getting closer to the right place. But the key takeaway here, exploring or scanning the Java library, is that most of them use regex, so that's something that we need to look at. Next slide, please.

They also look at Python. In Python, the story is a little bit different. 38% of projects contain a validation procedure. Next slide, please. And the story is different from the Java. The Python, the vast majority use an IDN library that relatively works well with the recent standard, and that is IDNA2008. So in the python, it's a better story than what we find in Java, but the takeaway here is that there's still work to be done, and these findings will inform next steps as far as what UASG is doing in terms of how to reach out to those maintainers of the open source repositories in order to encourage or give them education materials and what have you so that UA readiness advance, improve a little bit better in the open source world.

And I believe that's all of my slides. And Mark, do you want to add anything on this one? Please chime in. I know this does not do justice to the whole report, but just offer a glimpse of the findings.

MARK DATYSGELD:

Yeah, thank you, Dennis. Very briefly, this finding in particular about the usage of the IDNA module for Python I think is keys for us, because in previous talks with the developers of the Python language, they had concerns over how much overhead implementing IDNA2008 would add to the programming language, because they thought it would be heavy, because they thought it would be cumbersome, and yet, we find that it's the sixth most used module, exactly does the function of leveraging the default from IDNA 2003-2008.

So if so many users of the language are doing this, then we have a very strong case to make to them that they might as well implement this by

default in the language. So I think this is a very strong finding that we need to follow up on with the language developers and point out, hey, most of your users, a very significant quantity of them are using this already, so why not implement it on the base? So that was my key takeaway.

MARIA KOLESNIKOVA: Thank you, mark. Mohamed, do we have any questions?

MOHAMED ELBASHIR: No, Maria, we don't have any questions [at this time.]

MARIA KOLESNIKOVA: Okay, then we move forward and I provide the microphone to Satish Babu, the chair of technology working group. Satish, please.

SATISH BABU: Thanks, Maria. Hi everybody and good day to everybody. So over the next two slides, we will be looking at the state of the readiness of programming languages and frameworks. As you can imagine, this is the outcome of the work we have done earlier and we are now looking at this and trying to mitigate and remedy what we find here.

So on the slide here, we have C, Csharp, Go and Java for mainstream languages, and the picture can raise, between the greens and the reds, the current status of readiness. Now, if you look at C, the situation is not bad because one is green, one is yellow, so it's reaching there.

Csharp is also very similar. GO is both yellow, we don't have a green one yet. So the difference between yellow and green would be that a programmer who uses these libraries, if it is green, can be very sure about how they can use all the features, but if it is yellow, you may not be able to use all the features as expected.

In Go, you also have one red. It is a library with very low compliance, 19% which is amongst the lowest of all that we've tested. So it is a matter of concern and a key library, SMTP.

Coming to Java, we have the status either very good or very bad. So we have greens and reds, no yellow. So we note that another aspect that the slide of course does not explicitly mention is the kind of projects these are. For instance, Go was from Google which is very different from, say, Commons Validator which is from Apache. And we also have Csharp from Microsoft.

So from a mitigation perspective, this would be quite different in terms of how we can mitigate, because the fact that these are large companies or communities of open source developers or smaller groups, it'll kind of impact the way we plan to mitigate these gaps. Next slide, please.

So here, we have JavaScript, Python and Rust. Rust is a fairly new language but JavaScript is kind of very mainstream, although it's newer than Python. So in JavaScript, we do have one green and two yellows. Python is a complex case. It is heavily used, but kind of a legacy language when you compare it with Rust. But the point is that it is still used by many users, especially applications like some of the CMS,

content management systems, which makes it very important for the end user that we have them, or you have UA readiness in these libraries. So in Rust, we have one library that is very low compliance, but we do have one which is fairly well.

So to summarize, the picture is quite mixed as of now. But as we look forward, we are hopeful that we can see more of greens and less of reds and yellows as we progress with our mitigation and remediation efforts. So, thank you very much. That's all from my side. Back to you, Maria.

MARIA KOLESNIKOVA:

Thank you, Satish, very much, and thank you all. One more time, if you want to go deeper in all of these documents and issues, please visit uasg.tech website, and also, you can be subscribed to and follow us through the social media networks. You can find UASG there as well.

so now we are starting the second part of our session. it'll be a short roundtable discussion with our panelists. So we'll have Dr. Ajay Data, Dennis Tan Tanaka and Satish Babu who will be involved in this discussion.

We have three main questions to be discussed, but I would like to invite our participants to take a little part in our session and please, can you please raise your hands, or maybe use this reaction button in your Zoom and choose some nice picture and say, do you really understand what remediation and mitigation of universal acceptance means? Do you know it? So, what we are talking about, what is remediation of

universal acceptance? So just you can choose some heart or smile or something.

Okay. Don't be shy.

DENNIS TAN:

So Maria, it looks like we have to do a little bit of explaining.

MARIA KOLESNIKOVA:

Yeah. So I think that I want our speakers to—we have not so many reactions. Maybe you are right that our participants are shy, but I would ask our panelists. So, what do you understand under remediation of universal acceptance? So, what does it mean and what we are talking about? Just shortly and in simple words, please.

DENNIS TAN:

I can take that one. Thank you. If we can scroll back to our remediation slide, I think that picture talks 1000 words. So maybe that will clarify what we talk about remediation. Remediation, mitigation, we kind of use interchangeably. But basically, the end goal is to fix UA. But we need to start with knowing what are the areas of improvement. There's an old saying in management, if you can't measure, you can't improve it.

so this is where we'll start. We identify the technology that we want to evaluate, whether it's a web browser, an open source repository or a social media application or what have you, and we define, okay, what do we want to assess here in order to find these deficiencies or successes in terms of universal acceptance?

And when we talk about that, we are really narrowing down to the five actions of UA, acceptance, validation, processing, display and storage. So how do they perform against each of those functions in different applications and features?

Then we go about, we do these projects, whether we do it inhouse or we commission that work outside the UASG in order to expedite the discovery process, we get those results, as you have seen a glimpse of that in today's session, how open source is [inaudible] again, programming libraries, EAI applications and what have you, and then with that knowledge, what's wrong, then we can map out a path to remediate those issues, whether it's to reach out to developers directly, to reach out to software companies, to reach out to local communities and [extend] education and do other—what we need to do in order to fix or narrow the gap in terms of UA.

So that's kind of the cycles that we see here when we talk about remediation here. So let's assess, try to make an action plan how to address it, reach out, and start all over again. So hopefully, that explains when we talk about remediation. Thank you, Maria.

MARIA KOLESNIKOVA:

Thank you, Dennis, very much. I hope that this is important one more time to understand and go deeper in all these issues we are talking about today.

So okay, here is a first question to all our panelists. What is the best strategy to speed up the universal acceptance and e-mail address

internationalization mitigation and remediation efforts? So, what can be the best strategy to speed up? This is a question to our panelists, and if anyone has comments and following questions, please do not hesitate to put them in the chat.

Okay, who will be first? Dennis, can we start with you again?

DENNIS TAN:

Sure, Maria. Happy to. So, what's the best strategy? I'm not sure it's the best, but it's certainly one of the strategies that we could use or we are using to speed up the mitigation and remediation efforts, to speed up the getting to full UA state. And that starts with education, I think, and exposure of the UA concept and the—sorry about that, exposing those that make the—whether it's the standards, whether it's the guidelines, whether it's the actual code, expose them to Unicode and domain names and gTLDs and what have you.

For example, one specific example is in the world of registries, we use EPP, the extensible provisioning protocol in order to offer our registry services. So we know we have standards that are in this IETF forum and we have EPP for provisioning, we have standards for how to handle EAI, IDNs, but those are the standards, papers basically, would say, okay, this is what you can do.

But you need to also educate people how to actually implement the standards. So there is an ongoing effort today, there's an Internet draft, and [Verisign] is one of the authors of that draft, how to implement an operational model to implement EAI in EPP. And so bridging that gap

between a standard and actual operation, that's important, because it puts the knowledge in practice. And I can share that link later, so those in the IETF world, that's done within the REGEXT working group. I think it's been put forward for consideration for the working group to work on that. So I would encourage those that are interested in that work that support that draft and keep it moving.

So I'm going to leave it there. I think it's one of the strategies, educating, reaching out and exposing the technology, the standards and how to really implement those in the real world. Thank you, Maria.

MARIA KOLESNIKOVA:

Thank you, Dennis. Dr. Ajay, please.

AJAY DATA:

Thank you, Maria. Dennis has said a lot around that, but I would like to keep it very simple. That is, finding the gaps which are hurdles, which are bottlenecks, which are not allowing us to get there where we need to, and remediate that problem by providing the knowledge, solution, technical documents and everything else.

And we have to obviously take baby steps here. Not a very big leapfrog step where we think to convert the whole world in one go. We need to see that these small baby steps, like if we can do WordPress as UA ready, can we do a mail server completely UA ready? Can we have STML file, e-mail validation UA ready? Small steps which have a large impact worldwide. So let us say if we are able to successfully find a solution for WordPress to be UA ready and we have this implemented, then we are

looking at millions of websites getting UA readiness there. At least they have there sufficient tools to become UA ready. That's the kind of steps we need to take where we have a focused effort and we have the world getting impacted.

MARIA KOLESNIKOVA: Thank you, Ajay. Satish, can you add something, or you totally agree?

SATISH BABU: Thanks, Maria. First of all, a quick response to the question of what is mitigation and remediation. To me, it is my personal understanding—I might be wrong, so [inaudible] correct me, some of the efforts required to fix this gap are quite invasive. Earlier, Dennis mentioned the regex problem. That is a hard problem to crack. It requires maybe moving out from regex into some other way of doing it, because it is very difficult to cover all the cases in the IDN-based e-mail IDs to bring it under the regex—because regex is a programmer's shorthand and it's not easy to fit everything into that mode, whereas some others are low hanging fruits, maybe we can have a temporary fix, that would be mitigation. So remediation would be a deep process, quite invasive at times, so that would require significant willpower from the developer community, whereas mitigation may be temporary also because something else might solve the problem in time. So we need only a temporary solution or we need a fairly superficial kind of solution. In that case, you would talk about mitigation.

Now, regarding strategies, I have a couple of observations. One is that we are very clear that no strategy is going to work for all the different cases that we need to solve. One interesting thing would be to kind of document success stories. Whether it's a big tech company or a small open source community, they would like to know, "What is the point of doing this for us?" If you can highlight some of the success stories in terms of the numbers of users impacted or even in some cases the kind of new markets opened up, that would be an incentive for people to kind of consider solving these gaps. And of course, we have to have multiple strategies depending on whether it's the big companies or smaller communities. If it's an open source project, we may want to work through and with those communities, and we have already talked about having some kind of ambassadors in these communities from our side. It is not like our local initiative ambassadors but like having our own people working with these communities, highlighting and promoting and advocating for the fixing of these UA gaps. Back to you.

MARIA KOLESNIKOVA:

Thank you, Satish. I think [they three are quite in] agreement with the comments that are in the chat, and you have covered some of them already about low hanging fruit, for example. And there also was a comment that UASG needs to make dialog with the biggest companies that cover around 90% of the market and Internet users. So I think that this correlates with the next question in our queue, because the main question is how to engage these companies in universal acceptance issues and how we can start to work with them better.

And before this, I just want to comment on the [inaudible] that he wrote in the chat, yes, and I will follow Ajay Data that, okay, you are welcome to join and we always invite our community members to join UASG initiative and to make and provide their input there. So this is quite important for all of us and for the whole community, for the whole world, let's say, because we need people from ccTLDs, from gTLDs, from technology companies and so on, from academia community as well. So if you are one of them, please join us, at least follow us and provide your feedback on the work that we are doing.

And so we are going to our next question, and there were also some comments. Okay, there are comments on the [following question.] Then the question is, how to engage these tech companies and developers in all these processes? Do we have any tools, instruments, approach to engage them? Ajay Data, please, we will start with you this time.

AJAY DATA:

Okay. Sure. I think we have some success stories in India, especially about engaging with the tech companies and developers and motivating them enough to start looking at UA as a practice. Rather, some of the companies have started looking at UA as a practice and looking for business opportunities from the [knowledge] perspective.

The best approach for tech companies is always around commercial and getting new consumers, and we have to just let them know that if you are not building UA-ready applications, you are not getting new

consumers who are accepting new TLDs and new domain names with IDNs and EAI. And of course, you are losing those customers.

Once you are able to show that picture very clearly, they are automatically inclined towards hearing you well more. And if they are convinced with that idea that companies like Microsoft, like Google, Apple, have also amended their applications to support EAI and these kind of companies are taking steps, then it becomes a little bit more promising case for them to look at that area and start modifying these applications. This has worked in many cases in India, and I hope that it will work in other parts of the world also. Thank you.

MARIA KOLESNIKOVA: Thank you. Dennis, please.

DENNIS TAN: At the UASG, we have had successes in terms of reaching out to people from companies, and our message resonates with them, and as long as the message resonates with them, they will engage. That's what we have found.

Their involvement ebbs and flows depending on their priorities. What we have found is that we need to start reaching out more. I think now with ICANN Org also aligning with having a UA [ambition] on their strategic goals, that helps a lot because the Global Stakeholder Engagement really does help us reaching out, and that's helpful. I think we can work together with them in order to focus, I guess, the engagement or at least the message that we want to send them and

having that conversation will help us get more buy-in, if you will, from companies to participate in this area. So an idea to consider.

MARIA KOLESNIKOVA: So yes, this is our villain, I think, is we need to develop this strategy, how to reach them out practically, because this is quite an issue. Okay, Satish, please, you can add your point.

SATISH BABU: Thanks, Maria. So I think first point is that the most crucial thing here is to convince companies why they should be making their applications UA ready. That's easier said than done. In the past, we have explored a number of different options for closing these gaps. In some cases, if it's a very small library, we may be able to directly issue an SoW and get the code fixed directly. But that's going to be in very few number of cases.

In some cases, we have tried to use the bug reporting system of the particular community or company to highlight the fact that there's a need for it, so it'll at least bring it to their awareness repeatedly.

In some cases, maybe there is room for commissioning a group to create the code. We've not tried this so far, but if nothing else works, maybe we have to consider something like that.

But the key thing here is advocacy directly with communities, with companies, and although we're not talking about governments at this point, the government is also a key ally that we have to kind of influence the companies, because governments can set policies that provide

incentives and also disincentives for how products should behave. So these are some of my suggestions. Thank you.

MARIA KOLESNIKOVA:

Okay. Thank you, Satish, very much. Yes, and there was a recommendation in chat to contact the internationalization team in such big companies. But probably, this is ... So I think that if you want to comment on that, you are welcome, but probably internationalization of big companies mostly covers the issue with translation of their services and providing the work with local language content. And our issues are more technical and connected with, let's say, Internet identifiers, so probably, I don't know, if this is really good point to start to communicate this. Okay, so thank you for your comment, anyway, Harald.

And also—yes, identifiers, thank you. Our next question, it was already covered shortly that we have talked about how to engage the tech companies, but I think that some people—and there are many thoughts on this that many companies can be engaged at once when we have the government and governmental support for the universal acceptance issues implementation.

And my question to the panelists, do you really think that governments can play a role in generated demand for UA and EAI services and remediation? Please, Satish, you're the first now, you have mentioned this topic.

SATISH BABU:

Thanks for the question, Maria. For me, I think government is a very crucial partner in the journey towards UA readiness, especially since—of course, we do have the support of the ICANN Org and ICANN community, although we are an open community kind of a forum where we expect community to join in in these efforts. So we are counting on the fact that we have ICANN and our community with us, but the external partner that we must influence, and we have I think been fairly successful influencing, is the government, the community of governments.

Now, we realize that in some parts of the world, this is more an acute requirement than in other parts of the world. In India where I'm based, we have an extreme amount of linguistic diversity and therefore recently, there was an attempt by the .in registry to kind of promote IDNs, and there was one day of free registration where they could kind of ... And even I was given a free e-mail box, which is actually quite an interesting thing because now I can communicate with EAI address.

So at the policy level, the governments have a major role in setting a framework of incentives and disincentives that would promote the companies, whichever company wants to work with the government, this is one way to influence them.

So rather than us trying to communicate directly with the companies—I suppose that's also an effective tool in some situations, but not in all situations. But if it's us plus the government, then that makes it a very formidable force to influence the way these companies behave.

As Ajay mentioned a little earlier, the companies also have to see some kind of an incentive for them, financially that is. So it is creating a system of rewards and also constraints to actually get this thing moving. So I would reiterate that the government is the key player in this and we must think of partnering closely with the governments of those countries where IDN-based e-mails, the whole UA is an important factor and the whole country is held back when there is no UA.

So this is something that we can, I think, easily communicate to all affected parties, but it's not only the government, there are also other communities like the student community, the research community, etc. that we can also work with, but the government is a very major member of this. Thank you.

MARIA KOLESNIKOVA: Thank you, Satish. We have quite less of time, so Dennis, please, next you, and Ajay, you will continue, please. Shortly, please. Sorry.

DENNIS TAN: Yeah, absolutely, they have a role. Again, same as developers, it's education so that they set the right policies and frameworks for, one, they can acquire systems that are UA ready for themselves and their own constituencies, or just provide a way of platforms for other companies to also offer their services in a UA fashion. Thank you.

MARIA KOLESNIKOVA: Okay. Thank you. Dr. Ajay Data, please.

AJAY DATA:

Thank you, Maria. This question is very relevant because large impact policies are driven by government, and government is a key stakeholder in UA perspective. The two, three things which government can do and doing to promote UA—and I'm sharing an experience in India—the procurement policy. If government has a procurement policy while adopting software to be UA ready, just like they have a procurement policy for IPv6, if the software are procured with UA readiness, it becomes a very good pull from the market and developers automatically start looking at making software UA ready because they need to have customer as government [inaudible].

Second, to reach out to the people who do not know English and they would like to reach out to the people which is a large number, especially in India, Russia, China, Korea, Japan, where large population communicate in their own language.

And providing the 360-degree content and domain name and e-mail address is the way out for government to reach out to them. And that is an IDN and EAI. So that is also [an adoption] and taking to the people government can do it and should do it very well. We are doing our effort here in government of Rajasthan, the state I belong in, there are 7 million plus Hindi e-mail addresses, EAI. That's the largest deployment in the world for full-fledged mailboxes on EAI.

And this is a case study also available on UASG.tech, and people can look at that. Maria, let me also at the same time respond to Roland who is looking for an answer from leadership. Answer is very simple, Roland,

that as elections do happen every two years, secretariat announce the nominations as per the charter, and that charter allows anybody to nominate and you could have nominated, other people have nominated, and if there are more than three people looking to become vice chair and more than one person looking to become a chair, the elections do happen. And unfortunately, there was not enough people looking to fight for that position and fight voluntary work, so those people who nominated got elected. Please keep an eye on the discuss list. And no right is being taken away. Rather, the transparency is maintained as per the charter. I hope this answers your question. If you still have a question, please feel free to write to me or write at info@uasg.tech. We'd be happy to answer more. Thank you very much. Over to you, Maria.

MARIA KOLESNIKOVA:

Thank you very much. Actually, we're out of time, so I just want to mention some points here and wrap up. First, I want to mention that there are quite the really absolutely different approach in such countries that use language which are not based on the symbols that are similar to Latin one, so like India, like China, like Japan and so on, Arabic world, so on and so on. The situation is quite different and there are many people who are better acquainted with the English language, just sometimes don't really understand this difference. And this is really big issue also when we have our international dialog on universal acceptance.

The second point is please, one more time, join us in social media, in mailing list, on our website. You are more than welcome, and please provide your input in our works also. So I want to say many thanks to the participants who provide comments in the chat. They're valuable, I think, and it makes us feel that we are not alone and there are lots of UA enthusiasts around the world and you support our work as well.

So this is quite nice, and we are very pleased to have these comments. So I think that we will continue our discussion on some other ICANN meetings and between them during our working groups. So one more time, I thank you all, participants and our panelists, for this great discussion and your inputs and your thoughts. I think we need to analyze them and put and get include in our further work.

Thank you so much.

AJAY DATA: Thank you, Maria. Thank you, everyone, for joining in.

MOHAMED ELBASHIR: Thank you.

MARIA KOLESNIKOVA: Thank you.

[END OF TRANSCRIPTION]