ICANN85 Mumbai | CF – ccNSO: Tech Day (4 of 4)
Monday, March 9, 2026 – 16:30 to 17:30 IST

CLAUDIA RUIZ

Hello and welcome to the final session of ccNSO Tech Day. My name is Claudia Ruiz, and I, along with my colleague Susie Johnson, are the participation managers for this session. Please note that this session is being recorded and is governed by the ICANN Community Participant Code of Conduct, the ICANN Expected Standards of Behavior, and the ICANN Community Anti-Harassment Policy. Please observe the following guidelines to participate in this session; they will be posted in the chat for your reference. During this session, questions or comments submitted in the chat will be read aloud if put in the proper form, as noted in the chat. Interpretation for this session will include English, Spanish, and French.

If you would like to speak during this session, please raise your hand in Zoom. When called upon, virtual participants will be given permission to unmute. Onsite participants will use a physical microphone to speak and should leave their Zoom microphones disconnected. Please state your name for the record and the language you will speak if speaking a language other than English. Please speak at a reasonable pace to allow for accurate interpretation. Thank you, and with that, I will now hand the floor over to Regis Massé, moderator for this session. Thank you.

REGIS MASSÉ Thank you, Claudia. Ladies and gentlemen, welcome to the final session of Tech Day. My name is Regis Massé, and I am the CTO of the .fr registry. Today, we have three remote presentations during this session. We will start with the first one. Just before the break, we had an update for FRED, but there is a new platform to operate registry services nowadays. Ilya will talk about a project to automate registry services using this platform named Namingo. Thank you, Ilya, for sharing your lessons learned and your reasonable approach with the community. Ilya, you have the floor.

ILYA BAZLYANKOV Thank you, Regis. My name is Ilya Bazlyankov, and I have been around the community for a while; my first ICANN meeting was nineteen years ago. Recently, I became the core maintainer of an open-source project, Namingo. It is a registry and registrar platform under the MIT license. There are already some users both on the registry and registrar side, and I wanted to share what we learned as a team from all those migrations. Let me share my screen.

What we found out in the process is that some small ccTLDs still operate manually. Some of them directly edit the zone files when they want to add a new domain or remove an expired one. Others have a basic table, Excel, CSV, or other format that they use to keep track of contacts and expiration dates. Some of them process changes by email only, and usually, there is just one person doing

the whole process. If that person takes days of sick leave, then the registry is delayed in this process.

We would like to help, and we have been discussing the possibility of migrating to Namingo with three ccTLDs in the last year. I just learned during this ICANN meeting that there are two more who are exploring the opportunity. What Namingo can offer to them is the basic package, but it is an improvement to what they have for the moment. This includes an app server that is standards-based or RFC-compliant. If they like, they can have custom extensions for data validation or for specific policies that the ccTLD may have. It also includes a full RDAP server that is updated in real time or near real time, depending on the number of domains. There is a zone generation process and DNSSEC with all standard tools.

We keep their existing data and help them contact their registrars. We provide tools from a simple EPP client to EPP modules for the most popular paid and open-source billing platforms. All of these can be installed for a very small ccTLD on one or two virtual servers, including one for the backup. The migration pattern involves the ccTLD sending us their zone file or their CSV file. From there, we extract the data domain by domain and customer by customer. Sometimes it is complicated because data is in a non-standard format, it is mistaken, or it is outdated. We need to have a discussion with the ccTLD about what is current and what can be deleted. We take all of this, import it to a MariaDB database, activate DNS automation, connect to their name servers, and launch the RDAP server. Technically, if the data is okay, this process

can happen in a week. We already have the tools for that. The most complicated part would be if a registry has an internal issue matching contacts with domains, requiring them to manually check one by one. We have one user currently in that process, and we try to help them as much as possible.

Namingo offers an EPP server, an RDAP server, and a modern control panel. It can be used by registry personnel and by registrars. We have domain, contact, and host objects. It can work for ccTLDs and even for gTLDs. It can be supported by one or two administrators because it is written in PHP and uses all standard Linux tools for monitoring and for DNS. We also provide full logging of all changes, so the database has audit capabilities. Whenever a user changes something, it is seen. We have role-based access control; they can give access to their staff or their accountant, even with read-only data. We automatically generate the DNS and send it. We support BIND and other servers as secondary. Backup is done with fully open-source tools and can be done incrementally every hour or daily.

This system is pretty much proven. For infrastructure, we had a ccTLD with two thousand domains that ran all the infrastructure on one virtual server. Four gigabytes of RAM is enough. Of course, the bigger the better, but for their needs, it is okay. There were no issues, and the system is quite light. It uses just PHP and some other monitoring tools. Namingo can be deployed on any standard VPS, and there is no dependence on specific cloud providers. It just needs Ubuntu or Debian as operating systems. It uses the standard

EPP protocol with no mandatory extensions; that is up to the registry. There are no proprietary components. Everything is either Linux components under a GPL license or, in the case of Namingo, MIT. The full source is available on GitHub.

We also offer a registrar platform for small ccTLDs that they can use to test the platform or to launch their own registrar to automate the process of registering and renewing domains or changing name server records, which usually takes a lot of time for smaller entities. It can be installed as a brand new platform, or if they use a platform like WHMCS or the open-source equivalent, FOSSBilling, they can just use the modules and have their own registrar. It can be set for different categories, such as a registrar for local partners or partners abroad under different conditions. They can test all EPP procedures and commands with our client.

We can also offer partial adoption because our components are very much modifiable. The EPP server can connect to an existing database if the registry does not have EPP but does not want to migrate fully. The same is true for RDAP. They just need to write a few database requests, put in a custom driver, and it runs on any database. It can serve as a gradual replacement: first EPP, then RDAP, then the whole system. It can also work for ccTLDs that have a non-standard platform for communication. If they wish one day to offer EPP, they can just plug in our server.

From those few migrations, we learned that the main problem is data cleanup. It can be old, messy, or the ccTLD might not know if

some domains are active. They may have no customer contacts or need to match contacts to domains. This takes the longest. After this is sorted, it is just a matter of a script and import. We offer clear onboarding for registrars, documentation, the client, and EPP documentation. If they already have integration with another plain EPP, they can literally copy-paste and match the logic.

Namingo registry is best for smaller registry sizes and smaller teams with a lower budget who cannot afford to invest in new infrastructure. They prefer to have ICANN-aligned standards and wish to reduce the risk of a single person being on sick leave or holiday. In conclusion, I want to thank the organizers for this session and invite the community to share any issues, suggestions, or comments via email or GitHub. We respond quickly. We are open to helping any ccTLD with testing or installing a test system so they can see how it works. Now, if somebody has any questions, I am open to answer them here or later by email.

REGIS MASSÉ: Thank you, Ilya. A very interesting presentation. I have got a question from the room.

PETER THOMASON: Hello, Peter Thomason. This looks like a pretty cool system. It is a lot of work to develop it. I was wondering how it is funded and what your goal is besides making the software available. Why do you want to make it available, and what is your business model?

ILYA BAZLYANKOV

Thank you for the question, Peter. For the moment, it is funded just by the team. We have people from Bulgaria, Ukraine, and all over the world. They contribute in their free time. The goal is to have an open-source system fully available to anyone to use. As for a business model, if a ccTLD would like custom work or different help than the standard, this is how we make some money. But the main goal is just to make it available for the community, and that is why we do the work free of charge for smaller ccTLDs. If they can provide us with the infrastructure, we install it for them and help them write the migration script if they are willing to share the data. It is like that for now.

REGIS MASSÉ

Thank you, Ilya. Another question from the room.

PAULO TSINGERENDA

Paulo Tsingerenda from Malawi, for the record. We are a small ccTLD with a few domains. Over the past year or two, we have come to understand that there is a need for a variety of domain types to be registered. Some domains are premium and some domains are normal, for example. I do not know if Namingo has thought about this, whether you are treating all domains as the same or if there is an option for variety or diversity. Thank you.

ILYA BAZLYANKOV

Thank you for the question. There is an option to turn on the EPP price extension in the configuration, and you can also define pricing categories. It can be, for example, for premium or super premium domains versus standard domains. You can also have pricing categories for registrars. For example, you can have one price for a registrar inside the country and another for abroad, because this is the policy of some ccTLDs.

REGIS MASSÉ

Another question? I will ask one. Regis for the record. As the Security TLDops chair for the ccTLDs, I saw on the Namingo website that you have started working with small ccTLDs. We have to keep in mind that small ccTLDs are targets for attacks because they do not have all the resources in place to protect themselves. Have you performed some penetration tests to check Namingo's security level?

ILYA BAZLYANKOV

Yes. It is on a best-effort basis for the moment. Whenever we have knowledge from the team using automated tools or manually, we apply it. We also advise smaller ccTLDs to harden their infrastructure as much as possible; we help them with that. We continue testing and checking regularly for any new issues that can appear.

REGIS MASSÉ Okay, thank you for the answer. Thank you, Ilya, for this presentation. You will have a clap from the room remotely.

TIMO VOHMAR Hello, everyone. Can you see the slides?

REGIS MASSÉ Yes, Timo. It is okay.

TIMO VOHMAR Cool. Hi, everyone. I am Timo from the Estonian Internet Foundation, Head of Development. We are the ccTLD manager for Estonia. I am a big fan of data quality. My goal here today is to share a bit of my passion and maybe inspire someone or find other like-minded people among you to cooperate with and make the internet a better place.

The first reason for dealing with data quality, which I have been doing for fifteen years, is knowing that this is the key parameter of a safe zone. If a malicious actor cannot stay anonymous, they tend to find other places to do their bad deeds. That means data quality for us does not just mean technical checks. It also means looking behind the data and identifying the people and players behind domain registrations.

Last week, I had a chat with Steve and Klaus from Project SHAKE, who are here today, because they also have data quality in their focus. During that discussion, Klaus asked if we have considered

having a quality stamp or marking for domains with high-quality data. I could not understand the question at first because it does not play well with our philosophy; we aim at verifying one hundred percent of collected data. We believe that .ee is the stamp for quality. The other reason is, of course, NIS2, the cybersecurity act coming into force in Europe, which addresses the domain industry in this version. All registries and registrars that do business in Europe need to know about it.

We aim to identify and verify everyone before entering into a domain registration contract with them and verify their contact data. That is an ambitious goal. With any such goal, the first thing you should do is think about how to make your own life a bit simpler. For us, the first goal was looking critically at the data we collect and identifying the data that we do not use and therefore do not need. The first data bits to go were postal addresses and fax numbers because we do not write to or fax anyone. It is the internet, after all. We actually planned to drop the phone numbers as well, but the creators of NIS2 had other things in mind, so they are here to stay. This is called the minimal data principle, which is a key to any data quality endeavor.

We run a Registry-Registrar-Registrant model. The registrar is in contact with the registrant, so the task of strong identification falls on the registrars. As a registry, we list accepted methods of doing this, but it is up to the registrar to decide the correct method for a specific registrant. We used to require registrars to provide a full set of documentation collected with every registration, including all

document copies. Because we were the only ones in the world requiring that, it became a big burden for our registrars. A few years ago, we dropped that requirement. However, registrars still need to keep the proof of how they identified the registrants and must be able to provide it to us upon request.

As a registry, we validate everything. We double-check everything the registrars are doing. We do this before letting any bit of data into our database to keep the garbage out. We do regular revalidations of the whole dataset at least once a year. We focus mostly on private contacts. Even though eighty percent of our registrations are for businesses, a business is a virtual thing. A business does not do anything; it is a label for a group of people with the same goal. It is the individual who signs contracts and makes decisions representing that entity. We are very focused on the individual. In the case of business registrations, we require assigning a private admin contact to the registration. This person is the responsible one authorized to represent that company. This is usually the person who applies for the registration.

We usually do identification electronically using electronic IDs or bank identification. With bank identification, we do not only use identity-focused services provided by banks that are popular in the Nordics, but we also use bank transactions and payments. For example, if the payment comes from a bank account with data matching the registration data, we consider it verified.

REGIS MASSÉ Excuse me, Timo. I do not know if you have a slide because we are still on the title slide.

TIMO VOHMAR Sorry. I do not see my slides. Did it change now?

REGIS MASSÉ Yes, we have got the summary and the agenda.

TIMO VOHMAR Let me try to re-share them. What do you see now?

REGIS MASSÉ The 'Why' slide. Cool.

TIMO VOHMAR Thank you for letting me know. I would have gone to the end with the first slide. Electronic identification via bank ID is not available for a big part of the world. For the rest of the world, we have document verification. This is basically a video identification option that is also used by banks and financial institutions. A person presents their face to a phone or laptop camera and shows their document. Magic happens behind the screens to make sure the person is aware, awake, and alive. It compares the face to the picture on the document, reads the data, and checks the document against databases to make sure it looks real and is valid. We are able to cover the whole world with this. We have been running this

for a few years now with no problems. Failed attempts were always for a reason. We have seen attempts to fool the system by presenting copies, using video feeds, or showing screens with recorded material. Until now, none of these attempts have succeeded. We are happy with the system.

Because most registrations are businesses, we also need to verify business data. We do this against the Estonian business registry. This is an open data platform, which makes it easy and quick. The rest of the world is a bit of a Wild West, and we are still looking for a good solution for that. We also have technical validations for the rest of the contact data, such as email and phone numbers.

What if a check fails? The first action is that we deny any registrations with invalid data. To help our registrars, we introduced an ad-hoc verification service a year ago. This is mostly used by foreign registrars and some of our own local registrars that deal with registrants from China or parts of the world with no electronic identification options. The registrar can direct the registrant to our verification service to go through this electronic identification process and identify themselves via the document verification option. As I mentioned, business registry is a good data source for Estonian companies. We have implemented automatic fixes to the data. If a business changes its name or contact data, we update the data in the registry automatically so the registrant does not have to do anything.

If invalid data gets into the registry or becomes invalid over time, we have an administrative deletion process. This denies renewals depending on the severity of the problem. A critical problem would be a non-existent registrant, a deceased private registrant, or a bankrupt or liquidated company. In these cases, we remove the domain from the zone immediately, and the registrant has thirty days to fix the problem. Technical checks for phone numbers check if the area codes are valid, as well as the structure and length. We do not call or send messages to them yet regarding domain registrations, though we have implemented phone checks for our auction system.

With email validations, we have three levels. We have regular expressions to check if the string matches the email format according to RFCs. We do DNS-level checks to see if the TLD is there, if the domain name is actually registered, and if MX, A, or AAAA records are present. We also do SMTP-level checks, requesting the SMTP server to see if the specific mailbox actually exists. This means handling greylisting and other measures taken on the SMTP server side to battle spam.

This is how our verification works: the registrant applies for the registration, the registrar decides to send the registrant to an additional identification process, an email is sent to the registrant with a link, and the user is redirected to our eID service to choose between electronic identification or document verification. The registrar is notified once the process is complete. If successful, we

generate a certificate that includes the requested data retrieved from the identification process.

Looking ahead, we are currently working with DNS and DNSSEC records to complete the circle. Business verification has always been a big problem. There is a European Business Register, which is a central data exchange platform. Sixteen national business registries have joined this platform. It is a paid service, and data details vary from country to country, but it does not have an API. Even if that does not discourage you, it is closing on March 31st of this year. I think the only way forward is cooperation between registries and registrars to provide access to each other's national business registry options where available. That is pretty much what I wanted to say today.

REGIS MASSÉ Timo, if you can conclude, because we are out of time.

TIMO VOHMAR Yes, done as requested.

REGIS MASSÉ Okay. Any questions? We have got time for one. Yes, Calvin.

CALVIN BROWN: Calvin Brown here. How do you think the verification process you have put together affects market share in terms of adoption, and how does it affect the reputation of your ccTLD?

TIMO VOHMAR: It is a tricky question because we have been doing this since the beginning, so we do not know what would happen if we dropped everything. We can expect that numbers would be higher, but the count of problems would rise even quicker. We are not worried about this because we have always had these practices in place. The number of registrars is not huge, which is partly the reason why. Our registrars are familiar with the system and can help their registrants buffer this. We believe this is the reason why we have such a low number of issues in our zone.

REGIS MASSÉ: Thank you very much, Timo, for this presentation. A clap from the room for your presentation, please. [audience clapping] I propose to jump into the last presentation about Domain DNA, which is a technical initiative for domain name analysis and security made by Nic.at. It is an open-source project and can be found on GitHub. Alexander, please tell us about this interesting project.

ALEXANDER MAYRHOFER: Hello, everybody. Good afternoon to those who are in Mumbai and good morning or good lunch break to everyone else. My name is Alexander Mayrhofer. I work for Nic.at as Head of Research and

Development. The work I am presenting today was created together with my colleague, Clemens Moritz. I developed the idea and the visualization, and Clemens did the hardworking data science behind it.

We all know that as a registry operator, whether it is a ccTLD or a gTLD, a domain name has a life cycle. There are events during that life cycle and time intervals between those events. All registered domain names have such a life cycle. It is not just one domain; it is many. This becomes a pattern where transactions sometimes clump together, or there is a long time between them. We looked into whether it makes sense to compress the representation of this into a more compact format.

We came up with a simple scheme where we created a single-letter abbreviation for each transaction. For example, R represents a registration, N represents a name server change, T represents a transfer, D is a deletion, and P is the final purge of the database object after a redemption period. This represents the sequence of transactions on a domain name. We can also compress the time intervals between them into a different notation using a sequence of numbers. The higher the number, the longer the interval. This makes it very easy to differentiate the intervals from the transactions because there is no character overlap. This string represents a very compact representation of the transactions.

We also created some special characters. A slash as the first character represents an initial registration; the domain was never

registered before. An arrow to the right represents a fake transaction for 'now,' which allows us to capture the interval between the last known transaction and the current time. We used a visual separator to make it more human-readable, but this is not part of the actual informational representation. This sequence of characters looks like a meaningful combination, so we called it Domain DNA. We came up with the acronym Domain Doings Notation Abbreviated. 'Doings' is what we call the transactions. This Domain DNA represents the life cycle of the domain.

Intervals are a tricky thing because there is significance in very short sub-second intervals in transaction sequences. On the other hand, you also have very long intervals between transactions. This chart represents a histogram of all the transactions and intervals in our database. We tried to allocate intervals according to sensible time periods. Ten percent are below two hundred milliseconds, and it goes on from there. There is a significant spike at one second. It is important to understand that sometimes we measure database artifacts. The one-second interval exists because, in the past, we did not record milliseconds or microseconds on transactions.

It goes on to two minutes. We allocate time spots so a sensible amount of transactions fall into a certain sequence. If you look to the right at the sixty-thousand interval, there is a huge spike. This reflects our redemption period, which is a little over a month, between the deletion of a domain name and when it becomes available again. We made a compromise at sixty-five days. Two

months would have been nicer, but that spike would have been exactly at the peak. One point two years comfortably includes the typical renewal period. Then it goes on for older and very old domain names, reflected by the plus sign. This might look different for other registries; this is an exercise for the industry if you want to use it on a broader basis.

If a string has a slash at the beginning and an arrow to the right at the end, it represents the complete life cycle of a domain name. You can see here that there was another registration in between, meaning there are two delegations. These probably had different owners. The domain name was deleted with the P and then re-registered. We call an instance of a registration a delegation; this is one isolated registration out of the whole view of the domain name.

Sometimes there are patterns. For example, D:6P:6 represents sixty-five days. This is a delete followed by a purge about two months later, which is our classical redemption cycle. In another example, there is a purge followed by a registration under three seconds. This substring represents a quick drop catch. You can already see a certain pattern of transactions in these strings. Sometimes things happen simultaneously, so we use the ampersand for instantaneous events.

We also contract some transactions. In .at, a transfer implies a registrant handle change, but since this is a natural component of the transaction, we do not list the handle change. We propose not

to exclude R and S. R is a registration and S is a DNSSEC exchange. You might do DNSSEC exchanges at the beginning of the life cycle by registering with DS records, but you might also not. We keep those as separate transactions.

We can present the life cycle as a compact diagram using a metro metaphor. There are circles with identifiers and bars in between. Doings are circles with an identifier like R for registration. The sequence is left to right. Intervals in between represent time periods. We play with colors and intensity. If a domain name is not registered, we use a gray narrow line. When it is registered, we use a thicker line. We can also show whether the domain name is included in the DNS.

The challenge is representing twelve different intervals with different line lengths. The intervals range from sub-seconds to multi-year. In our visualization, we propose overlapping the buttons for instantaneous intervals and using increasingly longer lines for others. Once we go beyond five, we use bars like railway crossings to identify 'five plus something.' This makes it readable, even though seven years might look as long as four. It is a reasonable way to compress intervals.

In a real-world example of a twenty-year-old domain with three registrations, you can see it has a pretty busy life. For more than seven years, the domain was not registered after a deletion. Contrastingly, here is a quiet life with few transactions over twenty years. A fresh domain might have many shady name server changes

in a quick interval. This domain was recently locked for legal reasons.

The visual representation allows for grouping. These registrations from last September are centered around the fresh registration event. One is a drop catch because of the quick interval. Others look like standard registrations. Two domains have the same pattern: registration followed quickly by two name server changes. We can see these visually as anomalies. We can then cluster these fragments into groups using data science. For example, a quick sequence of name server changes or a drop catch faster than five hundred and fifty milliseconds. The biggest cluster is just a registration where nothing else happened, which represents the 'good guys.' We can also see re-registrations after seven years. This string allows us to cluster different domain names. An interesting example shows a registration, name server change, transfer, name server change, and transfer. These domain names often have other properties in common.

We can use machine learning models to create and predict Domain DNA strings. For example, after a slash, the model says R is the most frequent because it is a registration. It then predicts domain name life cycles reasonably, such as registration, name server change, transfer, DNSSEC, and name server change. We can use this to predict the future life cycle of a domain name based on enough previous data.

Our GitHub repository contains the Markdown specification and the visualization code used for these examples. It is SVG generated from JavaScript. We are open to feedback and discussions. We are currently looking at integrating this into our workflow, such as presenting a transaction overview to customer care. I am curious to hear about potential applications from others. This could be used for outlier detection or a model that predicts a domain name life cycle. We wonder if these specifications work for others, considering the .at sixty-five-day cooldown period and implicit auto-renewals. We could extend the schema for more extensive descriptions. Thank you for your time and attention. Any questions?

REGIS MASSÉ: Thank you very much, Alex, and I apologize for the duration. As we are running out of time, I will jump directly to Calvin for the wrap-up.

CALVIN BROWN: I am just going to wrap up, bearing in mind that I am between us and the bar and we are in negative time. I will keep it as brief as possible. We had four sessions: the first chaired by Andre, the second by Abdul, the third by Louis, and the last by Regis. In the first session, Warren Kumari started us off with how to make the root local. That is an implementation of RFC 8806, whereby recursive resolvers can get a copy of the root to speed up queries, enhance privacy, and increase resilience. I learned that a lot of

resolving software has this ability built-in; you just need to activate it. We then moved to something more local with Sagarika Saparamadu talking about their sixty thousand domain names. He took us through the security risks faced by ccTLDs, the potential consequences, and the mitigating steps. ISO 27001 was involved. We then moved to our host presentation, which addressed AI. He spoke about AI-powered detection of malicious .in domain names and their processes.

In the second session, which was a bit more free-flowing, we started with a presentation from NIXI. Samir took us through the IDNs. It was fascinating that he outlined the twenty-two official languages in India with a mix of left-to-right and right-to-left scripts. They have fifteen IDN ccTLDs with about three hundred thousand IDN registrations. We then moved to the .ru guys, who have almost eight hundred thousand Cyrillic domain names. He told us how they analyze their content and web space and how they check for user acceptance. Anant from ICANN gave us a rundown of their universal acceptance programs, where they look at programming languages, web servers, CMSs, and web browsers. Abdul then gave us a breakdown of the interaction of AI and user acceptance and how they can interplay. The panel discussion included contributions from others like Michael Bauland from Knipp and Sami Mohammed Ali from Bahrain.

In the third session, we spoke about Steve Crocker's proposed RDAP data access framework, and he took us through the work he has been doing. Jeff Osborn gave us an interesting talk about the

DNS root system. Anyone who needs a primer on how that works and how the whole ecosystem hangs together should revisit that; it is great if you are new or just want to brush up on your knowledge. We then went into a section about the FRED software and how they met the requirements for a CZ project using FRED and having new access to a particular zone for gov.cz.

This led us into our last session, where Ilya took us through Namingo, an open-source MIT-licensed ccTLD software, along with his processes for migration and onboarding. He mentioned the system passes the ICANN RST and is a full EPP system with zone generation, RDAP, and billing. It is under active development and is probably best suited to small emerging registries. Timo from Estonia showed us the verification model where they aim for one hundred percent verification of contact data and yearly revalidation. Finally, we heard about the Domain DNA project from Nic.at, where they can represent domain life cycle and activity to visualize various anomalies and analyze domain behavior. That is my wrap-up.

REGIS MASSÉ Thank you, Calvin. Ebba, do you want to say something?

CALVIN BROWN Thank you very much for lasting this long. We will see each other again in Sevilla.

REGIS MASSÉ	Thank you, everyone. This ends this ccNSO Tech Day. I wish you a very good ICANN week and see you next time. This ends this session.

[END OF TRANSCRIPTION]