UNICODE®

Internationalization & Unicode® Conference 41

CONFERENCE PROGRAM

Monday, October 16, 2017

09:00-10:30	SESSION 1 TUTORIALS

Presenter:

Track 1: Unicode / Writing - Part 1

Addison Phillips
Principal SDE, Internationalization Architect, Amazon

This tutorial helps you understand the unique characteristics of non-Latin writing systems that impinge on the implementation of Unicode-based applications. It doesn't provide detailed coding advice, but focuses on essential requirements you must understand to deploy Unicode-based solutions, and does so across a representative range of all the world's scripts (including Chinese, Japanese, Korean, Arabic, Hebrew, Thai, Hindi/Tamil, Russian and Greek). It also provides memorable examples to help you understand the buzzwords used in the rest of the conference and your future work with Unicode. The tutorial starts with basic character encoding principles, but goes much further, covering things such as input of ideographs, combining characters, context-dependent shape variation, text direction, vowel signs, ligatures, punctuation, wrapping and editing, font issues, sorting and indexing, keyboards, and more. It has a proven track record as an orientation for newcomers to the conference, but also appeals to people at intermediate and advanced levels, due to the breadth of concepts discussed and the way they are related to real-world script usage. No prior knowledge is needed.

Presenters:

Track 2: Introduction to Unicode and Beyond

Craig Cummings
Staff Consultant/ Evangelist, VMware

This tutorial will give you the knowledge for correct implementation for using Unicode to process text in any language. Unicode is the text encoding standard covering every major language on the planet. Taught by software internationalization experts, this tutorial will introduce you to the key principles of Unicode, its design and architecture, and provide you with examples of real world implementation. Attendees will come away with a basic knowledge of Unicode and how to be more effective at processing, handling, and debugging multilingual

Mike McKenna
Globalization Strategist

Tex Texin
Chief Globalization Architect, Xencraft

text content. The modules of the tutorial will cover: • Why is the Unicode standard necessary? What problems does it solve? • How computers work with text: Introduction to glyphs, character sets, and encodings. • Unicode Standard Specification and Related Data and Content • Principles of Unicode's Design • Components of the Unicode standard • Encoding forms, behavior, technical reports, database • How to use the Unicode Standard • Related standards - Integration with RFCs, IETF, W3C, and others • Unicode Implementation Details and Recommendations • Attributes, compatibility, non-spacing characters, directionality, normalization, graphemes, complex scripts, surrogates, collation, regular expressions and more • Unicode and the Real World - Support for Unicode in software platforms • International Components for Unicode (ICU) • Unicode in web servers, application servers, browsers, content management systems, and operating systems • Programming languages JavaScript, Node.js, C/C++, Java, PHP, SQL • How Unicode is evolving • Adding minority and other scripts, languages, and improving linguistic processing.

Presenter:

Track 3: Creating Fonts for the Universal Shaping Engine

Andrew Glass
Senior Program Manager, Microsoft

Windows 10 introduced a new shaping engine driven by Unicode data. As a result, Windows is able to support most complex scripts in Unicode and will add shaping support for newly encoded more rapidly than in years gone by. Now it's time for font developers to bring their talents to supporting the worlds complex scripts. This tutorial walks through the process of building an OpenType font for a complex script. It covers the end-to-end process of font development at a high level and pays particular focus to developing OpenType layout rules to work with the Universal Shaping Engine.

10:30-11:00 - Morning Refreshments

11:00-12:30 **SESSION 2 TUTORIALS**

Presenter:

Track 1: Unicode / Writing - Part 2

Addison Phillips
Principal SDE, Internationalization Architect, Amazon

This tutorial helps you understand the unique characteristics of non-Latin writing systems that impinge on the implementation of Unicode-based applications. It doesn't provide detailed coding advice, but focuses on essential requirements you must understand to deploy Unicode-based solutions, and does so across a representative range of all the world's scripts (including Chinese, Japanese, Korean, Arabic, Hebrew, Thai, Hindi/Tamil, Russian and Greek). It also provides memorable examples to help you understand the buzzwords used in the rest of the conference and your future work with Unicode. The tutorial starts with basic character encoding principles, but goes much further, covering things such as input of ideographs, combining characters, context-dependent shape variation, text direction, vowel signs, ligatures, punctuation, wrapping and editing, font issues, sorting and indexing, keyboards, and more. It has a proven track record as an orientation for newcomers to the conference, but also appeals to people at intermediate and advanced levels, due to the breadth of concepts discussed and the way they are related to real-world script usage. No prior knowledge is needed.

Presenters:

Craig Cummings
Staff Consultant/ Evangelist, VMware

Mike McKenna
Globalization Strategist

Tex Texin
Chief Globalization Architect, Xencraft

Track 2: Unicode in Action

The Unicode in Action tutorial is a 90 minute session that demonstrates programming with Unicode and related best practices. This tutorial will build a simple application and demonstrate the code and resulting behavior as internationalization functions are added. Attendees will be able to relate these prototype examples to the requirements of their own applications and reference them to code solutions. The program will show sorting of different strengths, regular expressions, Unicode normalization, bidirectional languages, and other features of the Unicode standard. The tutorial will highlight why each of these functions are needed so you can determine when to use them in your applications. This tutorial is updated for IUC41.

Presenters:

Norbert Lindenberg
*Internationalization Solutions Developer and C*O, Lindenberg Software LLC*

Muthu Nedumuran
Founder and CEO, Murasu Systems Sdn Bhd

Track 3: Creating Fonts for Brahmic Scripts with OpenType and Apple Advanced Typography

Brahmic scripts such as Devanagari, Tamil, Thai, and Burmese have complex requirements for mapping characters into correct arrangements of glyphs, including glyph reordering, conjunct formation, mark stacking, and mark positioning. OpenType, the font technology supported in all major operating systems, and Apple Advanced Typography, the technology in Apple’s operating systems, use different approaches to support the creation of fonts that meet these requirements: OpenType provides specialized shaping engines for several scripts as well as the universal engine for numerous others, while AAT provides a flexible generic shaping language. This tutorial discusses the requirements of Brahmic scripts and the approaches used by the two technologies, and provides practical guidelines for creating fonts.

12:30-13:30 - LUNCH

13:30-15:00 SESSION 3 TUTORIALS

Presenter:

Addison Phillips
Principal SDE, Internationalization Architect, Amazon

Track 1 - Internationalization: An Introduction

What is internationalization? What are localization and globalization? How does internationalization enable software teams, large and small, to ship software that delights a global audience? How do you incorporate internationalization into the design, implementation, and delivery of a software product? This tutorial presents the basic concepts, with a focus on real world examples, so you can understand how to analyze a product for internationalization issues, develop a design or approach, and deliver a global -ready solution.

Presenter:

Tex Texin
Chief Globalization Architect, Xencraft

Track 2 - Web Internationalization

This tutorial, updated in 2017, is an introduction to internationalization on the World Wide Web. The audience will learn about the standards that enable global interoperability and come away with an understanding of how to work with multilingual data on the Web. Character representation and the Unicode-based Reference Processing Model are described in detail. HTML, including HTML5, XHTML, XML (eXtensible Markup Language; for general markup), and CSS (Cascading Style Sheets; for styling information) are given particular emphasis.

The tutorial addresses language identification and selection, character encoding models and negotiation, text presentation features, and more. The design and implementation of multilingual Web sites and localization considerations are also introduced.

Presenter:

Steven Loomis
Software Engineer, IBM

Track 3 - Putting ICU to Work

This tutorial gives attendees everything they need to know to get started with working with text in computer systems: character encoding systems, character sets, Unicode, and text processing, using the International Components for Unicode library (ICU). ICU is a very popular internationalization software solution, and is now hosted by Unicode itself. However, while it vastly simplifies the internationalization of products, there is a learning curve. The goal of this tutorial is to help new users of ICU install and use the library. Topics include: Installation (C++ libraries, Java .jar files, Java SPI for JDK integration), verification of installation, introduction and detailed usage analysis of ICU's frameworks (normalization, formatting, calendars, collation, transliteration). The tutorial will walk through code snippets and examples to illustrate the common usage models, followed by demonstration applications and discussion of core features and conventions, advanced techniques and how to obtain further information. It is helpful if participants are familiar with Java, C and C++ programming. Issues relating to ICU4C/C++ as well as ICU4J (Java) will be discussed. After the tutorial, participants should be able to install and use ICU for solving their internationalization problems. Topics covered will include packaging of ICU data, integrating ICU into an applications development process, and how to get involved in the ICU development community.

15:00-15:30 - Afternoon Refreshments

15:30-17:00	SESSION 4 TUTORIALS

Presenter:

Jim DeLaHunt
Principal, Jim DeLaHunt & Associates

Track 1 - Email addresses and domain names are NON-latin! Now what?

Email addresses, and domain names, are no longer limited to ASCII Latin script. They can now be http://普遍接受-测试。世界 or مانيش @ أشوك. الهند or données@fußballplatz.technology. Software, frameworks, and workflows will need to change to accommodate. What are Internationalized Domain Names (IDN) and Email Address Internationalization (EAI)? What do you need to know? What do you do next? This tutorial brings you up to speed. It explains IDN and EAI. It shows you the implications. It connects you to sources of information. It helps you understand what this will mean for you. Suitable for software developers, QA, marketers, system administrators, and management.

Presenter:

Mihai Nita
Software Engineer, Google, Inc.

Track 2 - Android Internationalization and Localization

A tour of Android’s internationalization and localization features, including a tutorial for developing an internationalized Android app from scratch (localizability, formatting, bidi, etc.). New internationalization-related features of Android N will also be discussed, especially the new support for multilingual users.

Presenter:

Martin J. Dürst
Professor, Aoyama Gakuin University

Track 3 - Character Equivalences, Mappings, and Normalization

The wealth of characters in Unicode means that there are many ways in which characters or strings can be equivalent, similar, or otherwise related. In this tutorial, you will learn about all these relationships, in order to be able to better work with Unicode data and programs handling Unicode data. Character relationships and similarities in Unicode range from linguistic and semantic similarities at one end of the spectrum to the same character being represented in different character encodings or Unicode encoding forms at the other end of the spectrum. In the middle, there are numerical and case equivalences, compatibility and canonical equivalences, graphic similarities, and so on. This sometimes bewildering wealth of characters, equivalences, and relationships is due to the rich history of human writing as well as to the realities of character encoding policies and decisions. The tutorial will give some guidance to help users navigate equivalences and differences for their use cases and applications. Each of these many equivalences or relationships can or should be ignored in some processing contexts, but may be crucial in others. Contexts may range from use as identifiers (e.g. user ids and passwords, with security consequences) to searching and sorting. For most of the equivalences, data is available in the Unicode Standard and its associated data files, or is provided by other standards such as IDNA and PRECIS. But the use of this data and the functions provided by various libraries requires understanding the background of the equivalences. When testing for equivalence of two strings, the general strategy is to map or normalize both strings to a form that eliminates accidental (in the given context) differences, and then compare the strings on a binary level. The tutorial will not only look at officially defined equivalences, but will also discuss variants that may be necessary in practice to cover specialized needs. We will also discuss the relationships between various classes of equivalences, necessary to avoid pitfalls when combining them, and the stability of the equivalences over time and under various operations such as string concatenation. The tutorial assumes that participants have a basic understanding of the scope and breadth of Unicode, possibly from attending tutorials earlier in the day.

Tuesday, October 17, 2017

09:00-09:15	***WELCOME & OPENING REMARKS***
09:15-10:00	**KEYNOTE PRESENTATION - Can We Escape Alphabetic Order? Chinese I.T. Before and After Unicode**

Presenter:

Thomas S. Mullaney
Associate Professor of Chinese History, Stanford University

Long before Unicode, China’s search for a stable text encoding system dates back nearly two centuries. In the age of electric telegraphy, China’s non-alphabetic writing system confounded Morse Code. In the age of word processing, the script’s 70,000-plus characters overwhelmed ASCII. And even today, in the era of Unicode, a polyphony of Chinese text encoding systems continue to compete for dominance, with new ones being invented each year.

Drawing upon more than a decade of research in the fields of Chinese and Non-Western information technology, Stanford historian Tom Mullaney maps out the parallel and still-uncharted worlds of Chinese IT. Worlds where text encoding systems have never been hidden from the average user’s view, but instead where they are directly manipulated, command-line-style, by hundreds of millions of code-conscious users. Worlds where the mythology of “plaintext” was never allowed to take hold, and where everyone knows that ‘WYS’ is not ‘WYG’.

With vivid examples pulled from the archives of telegraphy, computing, machine translation, digital typography, and more, he will give a guided tour of China’s 200-year-old quest for a stable information order, posing a fundamental question along the way: Why has the Chinese script proven so difficult to encode, and what does this tell us about the fundamental inequalities that are still baked into our modern-day information order?

10:00-10:30 - Morning Refreshments

10:30-11:20 **SESSION 1**

Presenter:

Track 1 - An Artist’s View of All Writing Systems Ever

Kaitlin “Ducky” Sherwood
Artist

Sherwood's glyph-themed artworks, including _Rugzetta_ (a mashup of Persian rugs and the Unicode standard) turned into a roadmap for a researching a broad history of writing systems.

Through visual images, Sherwood will discuss interesting things she found by looking at a broad scope of writing systems. How many times were writing sytems developed by illiterates? How often did a culture change its writing system wholesale and why? What are some interesting alternate technologies for making human communication persistent? What is the second life of Unicode character encoding proposals as cultural raw material?

Presenters:

Track 2 - I18n & Machine Learning for Mobile Input: Smart Keyboards, Speech Recognition

Daan van Esch
Technical Program Manager, Google, Inc.

Elnaz Sarbar
Program Manager, Google, Inc.

Smartphone users want to be able to use mobile input tools such as keyboards and speech recognition systems in their own native language. Simple keyboards exist for many languages, but smart keyboards with auto-correction, next-word prediction, and glide typing require machine learning models, which are harder to internationalize. Speech recognition requires even more machine learning components.

In this talk, we'll describe our best practices for scaling machine learning systems to dozens or even hundreds of languages, like building one-button data-driven pipelines, relying on linguists to inject human knowledge efficiently, and more. Using these approaches, we've been able to extend Google's speech recognition system to support 80+ language varieties. We've also brought 140+ fully supported languages to our smart Gboard keyboard app.

Presenters:

Track 3 - Salesforce Grammar Engine

Pu Chen
Lead Engineer, Salesforce

Bo Yang
Senior Manager, Salesforce

Steven Tamm
CTO, Salesforce

Has your Localization team or translator complained about sentences like "Edit {0}" or View open "{1}"? Imagine having to enable customization for customer business jargons? Maintaining grammatical correctness has always been a problem in UI localization. Salesforce grammar engine is a Java library that enables programmatic renaming of nouns while maintaining grammatical correctness, including genders, pluralization and starts with. In this technical session, we will explain and demonstrate how Salesforce grammar engine solves the problems to encode the article, noun, and adjective declensions properly for 65+ languages and support programmatic use of nouns.

The session will also provide a look at how the feature called "Rename Tabs & Labels" in Salesforce implemented the grammar engine which enables administrators to change the name of standard Salesforce objects.

11:30-12:20

SESSION 2

Presenters:

Claudia Galvan
Early Stage Innovation, Technical Advisor

Adam Asnes
CEO, Lingoport

Track 1 - Practical Approach to Internationalization for Startups

All startups are certainly not created equal in terms of funding, strategy maturity and stage of development. They are interesting in terms of cultural understanding of globalization and balancing the creation of technical debt in pursuit of immediate goals that may cost down the road. There isn't necessarily a right and wrong answer in many cases and business situations rule. We have anecdotal experience with early stage firms that either baked globalization into planning once past the prototype MVP stage, and those who were forced to address it for a sale. In this talk, we will review creating a global strategy, addressing technical debt, defining international requirements and i18n and l10n best practices to aid pivoting successfully an early stage startup into a global player. Take-aways:

- Moving from tactical to strategic product and development planning
- Addressing technical debt
- Starting from scratch
- Legacy code
- Developing requirements for MVP through common user scenarios
- Continuous systems and practices for supporting ongoing i18n and L10n

Presenter:

Suresh Prasad
Senior Engineer, Uber Technologies

Track 2 - Dynamic Localization in Mobile Applications

This session presents the design and implementation of a dynamic localization model for mobile applications at Uber. It was built to address the issues with a static localization model, where localized strings in mobile applications cannot be changed after a mobile application has been released to the app store. Uber is available in hundreds of cities across the world and needs to support localized versions of their mobile applications across nearly 50 languages. At Uber, strings and their localizations are inserted into a mobile app bundle which is then released to the App/Play store on a weekly basis. This results in a static localization model; once strings and their localizations reach the App/Play store, they are not updated. This has three main limitations: 1) new mobile application releases cannot be released until localizations for all strings are available, 2) incorrect/missing localizations have to wait until "the next release" before being corrected, and 3) A/B testing localized copy on mobile clients is not well supported. Dynamic localization at Uber has two main components: 1) A backend service to construct and store snapshots of localization updates and 2) a change to mobile applications to download updated localizations on app restarts. The backend service is implemented in Golang and uses a CDN to serve JSON files containing the localization update snapshots, which are then downloaded and applied by mobile applications. This session discusses the motivation and technical components of this project. In particular, cursor-based update fetching, Redis-backed distributed locks, concurrency via goroutines in Golang and dynamic refreshing of translations by mobile clients.

Presenter:

Track 3 - Localized Number Formatting in ICU and Beyond

Shane F. Carr
Software Engineer, Internationalization, Google, Inc.

How many ways can you display a number like "1234" in different locales around the world? Did you know that Unicode has digits in over two dozen non-Latin scripts? What's the difference between Arabic digits and the Arabic numbering system? When parsing a string such as "987.789", is the period a decimal separator or a grouping separator? How do you display currencies that have different rounding rules? This presentation will have two parts. The first part will provide background, answering the above questions and discussing how numerals are encoded in the Unicode standard. The second part will discuss formatting localized numbers in ICU, with a focus on new features added in ICU 60. You will learn how ICU formats numbers under the hood, how to write a decimal formatting skeleton string, and best practices on using the new number formatting API.

12:30-13:30 - LUNCH

13:30-14:20 **SESSION 3**

Presenter:

Track 1 - The History and Evolution of Typing Systems

Daniela Semeco
President, Polyglotte Inc.

From Ancient Chinese linotype to the Gutenberg printing press, this tutorial will cover the history of typing systems and their evolution. Inputting systems define the way we live and interface with the world. There is so much we take for granted when it comes to keyboards, but like words, they each have their own story. This tutorial will present examples of some of the world's most popular typing systems, from ancient times to today: • First, we will look at various typing systems, their components and how they work. • Next, we will cover their origins and what made them popular. • Lastly, we will analyze how the various systems have influenced current trends and where the industry is going. Some solutions have been mandated by governments, others born out of necessity. Every attempt of making a Chinese typewriter has failed; with some designers spending their whole lives architecting intricate solutions. Even now, as virtual keyboards are able to do more than physical keyboards, we've become addicted. Emojis have become relevant in keyboards as well as pop art. Whichever way you wish to look at it, designing a keyboard is not a trivial affair. It can profoundly impact the way we experience the world, whether involving new scripts that are being integrated into Unicode for the first time, or an innovative way of interfacing with a century-old layout.

Presenters:

Track 2 - Pseudolocales in Android and CLDR – Normalization Efforts

Katsuhiko Momoi
Staff Test Engineer, Google, Inc.

Igor Viarheichyk
Software Engineer, Google, Inc.

In a recent CLDR release 31, we have added 2 pseudolocales, en-XA and ar-XB, through a tool that can generate them. This represents a culmination of our multi-year efforts to "normalize" pseudolocales. In this presentation we delve into the details of how pseudolocales are generated for CLDR and Android platform and how we went about popularizing their uses at Google. They key concept is "normalization" of pseudolocales in the development environment. Our efforts to "normalize" pseudolocales have now brought us to a state where in all respects adding script-generated pseudolocales to an app is no different than adding human generated/translated locales such as French and Russian. In a large organized development environment, such normalization is critical for adoption. Pseudolocales are very familiar tools in internationalization and localization context -- there had also been several talks that touched on them at past Unicode conferences. At Google some forms of pseudolocales existed as early as 2006. There were a number of problems that prevented them from being widely deployed and used by product teams. For example, there were no fixed names for pseudolocales and different project used their own ad hoc names. Another problem was the locales names needed to be parsed by different platforms such as Java, C++, Javascript, etc. but many of the names used were not BCP47 compliant and failed in parsing. Yet another problem was consistency in what pseudolocales meant -- some projects applied one set of pseudolocalizing rules to the original data and another project applied a different set of rules to pseudolocalize the original. This led to inconsistent appearance of the locales depending on the project and mastering the use of pseudolocales in one project did not make it easier to use them in another if the second project deployed different pseudolocales. These problems illustrate the essential problem of this haphazard approach -- no consistency in what it means to have a pseudolocale and impossibility to establish it as a common tool for i18n testing/debugging. Our efforts to rectify these problems started with standardizing the locale naming and settling on 2 specific implementation of pseudolocales. Once we agreed on the naming, we then worked on standardizing pseudolocalizing methods. Then we followed up with auto-generating these data for all of our projects and made them available in the source tree with updates occurring whenever the master English data were updated. The next step was to integrate the generation of pseudolocales for apps into the common BUILD environment. In this presentation we will show how build packaging option works on Android platform and how you can use them also in Android Studio. The latest phase of our efforts was to address how we can provide realistic locale formats under the pseudolocales when projects use ICU to provide localized formats. Until recently we were leaving this part alone, but now with the addition of the 2 pseudolocales to CLDR, ICU itself can offer localized formats for en-XA and ! ar-XB. Standardizing on a specific set of rules and implementation options means “fixing” how the pseudolocales look and behave. These could be considered “limitations” to this type of approach. However, this ”normalizing” also offers great benefits. For example, products integrating other products or components can have the same pseudolocale appearance in all the UI parts. And this also allows us an easy way to build the standard pseudolocales with a build option flag.

Presenters:

Track 3 - New in ICU

Markus Scherer
Software Engineer, Google, Inc.

Steven Loomis
Software Engineer, IBM

The International Components for Unicode library, or ICU, provides a full range of services for Unicode enablement, and is the globalization foundation used by many software packages and operating systems, from mobile phones like Android or iPhone all the way up to mainframes and cloud server farms. Freely available as open source code from Unicode, it provides cross-platform C/C++, and Java APIs. This presentation will provide an overview of ICU, with emphasis on the recent updates in ICU 59 & 60, including the latest support for Unicode 10.0, Emoji 5.0 and CLDR 31/32, line-breaking improvements, new number formatting code, migration to C++11, and other changes (see http://site.icu-project.org/download). The presentation will also touch on ICU’s planned direction for future releases.

14:30-15:20 **SESSION 4**

Presenters:

Track 1 - Panel: Making Cuneiform & Old Italic Great Again: Dealing with Variants in Historic Scripts

Deborah Anderson
Researcher, SEI, Dept. of Linguistics, UC Berkeley

Adam Anderson
Mellon Postdoctoral Fellow in the Digital Humanities,

Even after historic scripts have been published in Unicode, successfully using them in text is often a challenge. A fundamental requirement for encoded scripts is, of course, availability of keyboards and fonts. Users of many historic scripts often also need access to glyph variants, since Unicode may have unified a variety of shapes that vary depending upon language, time period, and/or geographical location. There are, however, different ways to handle glyph variants, and certain approaches may fit specific scripts better than others. Also, some options may not be well supported across all software. This panel will include two brief case studies outlining the problems in cuneiform and Old Italic and will be followed by a seasoned typographer who will present the

UC Berkeley

Kamal Mansour
Linguistic Typographer, Monotype

technical options available, along with their pros and cons.

This panel will include:

- Glyph variation found in cuneiform
- The case of Old Italic, which unified different historic alphabets
- The range of choices facing those working with historic scripts: separate fonts, TrueType collections, Variation Selectors, and OpenType stylistic options

The goal of this panel is to highlight the problems and issues, and invite suggestions from audience members so guidelines on glyph variants can be developed for tech-savvy users, and font and software developers. The ultimate aim is to make it possible for scholars, students, and others to create and access texts in historic scripts, which will highlight the historical and cultural contributions of these past civilizations.

Presenters:

Rohit Puri
Manager, Netflix Inc.

Andy Swan
Senior IU Engineer, Netflix Inc.

Track 2 - Internationalized Timed Text at Netflix

Timed text assets, such as subtitles and captions, play an essential role in Netflix global user experience. This presentation offers an overview of technical innovations and challenges of two teams at Netflix Content Platform Engineering in this field and focuses on how standards such as Unicode, TTML, and IMF play an important role in these efforts.

Presenter:

Daniel Bruhn
Software Engineer, i18n, PayPal

Track 3 - Extending Libphonenumber for Fun and Profit

Google's libphonenumber library is the de-facto industry standard for processing international phone numbers, providing support for formatting, validating, and normalizing phone numbers in 250+ regions. However, the default phone metadata is quite heavy. Various custom packages have reduced the code & metadata footprint by:

- Simplifying the API and pre-compiling with Closure (grantila/awesome-phonenumber)
- Providing individually compiled code+metadata bundles for each region (leodido/i18n.phonenumbers.js, nathanhammond/libphonenumber)
- Rewriting the entire library without Closure and providing the option to hot load metadata for groups of regions (halt-hammerzeit/libphonenumber-js)

For my use case, I needed:

- The official libphonenumber code (not a pure JS re-write)
- A static code base that doesn't change for different regions
- Hot loadable metadata bundles for individual regions

And so, I implemented yet another custom JS libphonenumber package. In this talk I will present this package and discuss some of the interesting challenges in producing a customized version of libphonenumber.

15:20-15:50 - Afternoon Refreshments

15:50-16:40	SESSION 5

Presenters:

Edward Stratford, Ph.D.
Brigham Young University

Benjamin Mackley
Brigham Young University

Track 1 - Crowdsourcing Cuneiform: Leveraging Human Pattern Matching to Pave the way toward Cuneiform OCR, Accomplishments and Challenges

Optical character recognition has become a vital method for data transformation. However, image processing logic is still dealing with a range of challenges when moving from typed pages to other forms of writing. Cuneiform, the writing system used by the Assyrians and Babylonians, poses a particularly difficult challenge. Unlike typed forms, layouts of the clay tablets often vary, and individual characters can be compressed or overlap. Moreover, cuneiform is polysemous: a single sign can be read in various ways depending on context. This complicates logic to be used for refining propositions about an individual character. At the same time, cuneiform signs are, in principle, more easily reducible than some other forms of writing because they are ultimately configurations of wedges. (In this way, cuneiform OCR shares some problems other types of geometric pattern matching.) Machine learning is a promising method toward refining a process for cuneiform OCR. However, the scale of the problem at present makes the task daunting. Enter tabletninjas.com. Like Captcha, tabletninjas.com aims to draw on human pattern matching capabilities, but augment the users' capabilities with expert-curated guidance. Within the Old Assyrian cuneiform corpus, more than 10,000 tablets have already been read and roughly half have digital images. Tabletninjas.com uses data from the Old Assyrian Research Environment (OARE). OARE, a project within the Online Cultural and Historical Research Environment (OCHRE, ochre.uchicago.edu), uses a data model in which each and every sign for each and every tablet is stored as an individual database item with its own UUID. Using data from OARE, tabletninjas.com presents an average user with a photograph paired with a series of drawn signs, arranged in the same order they appear on an individual line on the tablet. The user then can drag and drop the drawing of a individual sign onto the relevant area of the photo, resize, and accept. Tabletninjas.com will soon provide short tutorials to accustom user to the relation between a drawing of a sign and what it looks like in a photo. But development has already shown a relatively quick adaptive transition for the user. With this system in place, tabletninjas.com aims to go in several directions, including a learning tool for cuneiform. However, the path toward cuneiform OCR leads through handwriting analysis. Because the tagging action in tabletninjas.com is based on a dataset that is contextualized to individual documents, locations on tablets, and writers, complex analysis on the data can proceed by recourse to a number of variables. Through the data provided from OARE, it will be possible to aggregate large numbers of examples of different signs, then analyze them for variability among a range of writers, thus leading to data from which to derive optimization algorithms for character recognition. The talk will discuss the development of the project thus far, and desires to map existing data onto higher resolution versions of the photos offline, and the intricacies of handwriting analysis. Plans to gamify the platform will also be discussed. Ultimately, the project should provide a large data set on which to discuss cuneiform allographs and strategize how such variations can be mapped onto the appropriate Unicode cuneiform character. Once such determinations can be made, it will be possible to unleash the utility of Unicode to a broader set of cuneiform specialists, and eventually index all cuneiform to Unicode, and thus more fully bring into the digital age one of the major datasets from the first half of history.

Presenter:

Track 2 - Intl All the Things: Unicode in Node.js

Steven Loomis
Software Engineer, IBM

Node.js has become a popular platform, using JavaScript on the server, or in other environments outside of its traditional role in web browsers. This presentation will discuss the latest status in enabling and making use of the Intl (EcmaScript-402) module support in Node.js, the status of the library ecosystem, and what's next for JavaScript and Node.js globalization with upcoming features slated for future EcmaScript-402 and EcmaScript-262 editions and discuss techniques and best practices for Unicode and international support in Node.js applications.

Presenter:

Track 3 - Comparative Analysis of W3C Text Layout Requirements

Behnam Esfahbod
Founder, Virgule Typeworks

W3C's Internationalization Activity is documenting text layout and typographic needs of various writing systems and languages such as Arabic, Chinese, Ethiopic, Hebrew, Indic, and Japanese for better support in web technologies and other e-publications. Working Groups are formed by experts from around the world to document these requirements and the existing gaps between the real-world expectation and existing solutions. In this talk, we’ll have a review of the Working Groups activities and comparative analysis of their publications, looking at problems and solutions unique to each writing system and/or language, and how common problems are addressed differently. Major requirement areas covered are: script overview, document compositions, page formats, paragraph and line compositions and adjustments, inline features, and special cases.

16:50-17:40 SESSION 6

Presenter:

Track 1 - Digitizing Ethiopic: Coding for Linguistic Continuity in the Face of Digital Extinction

Isabelle Zaugg
Mellon-Sawyer Seminar Postdoctoral Fellow in “Global Language Justice,” Institute for Comparative Literature & Society, Columbia University

Despite the growing technical sophistication of digital technologies, it appears that they are contributing to language extinction on a par with devastating losses in biodiversity. With language extinction comes loss of identity, inter-generational cohesion, culture, and global wealth of knowledge to address future problems facing humanity. Linguists estimate a 50%-90% loss of language diversity during the twenty-first century, with the lack of digital support for minority languages and their scripts a contributing factor. Over time, digital design has come to support an increasing number of languages, but this process has been largely market-driven, excluding languages of communities that are too small or too poor to represent viable markets. Lack of support for a language in the digital sphere means that language communities begin using other more dominant or “prestigious” languages for digital communication, ultimately resulting in “digital extinction”, or the impossibility of raising youth, who are particularly dependent upon digital communication, fluent in their mother-tongue. This research investigates the role of digital design and governance in including or excluding languages from the digital sphere through the instrumental case study of Ethiopic, a script that supports the national language of Ethiopia, among others at risk of digital extinction. This research investigates late 20th century efforts to include the Ethiopic script in Unicode, the dominant digital standard that allows scripts of the world to be used on software, devices, and websites, as well as the ongoing challenges the script faces for full digital viability in the 21st century. Concluding with policy implications and best practices for digital design and advocacy efforts, this research sheds light on the far-reaching implications for the public good of digital design and standards governance, and their impact on global language diversity. At a moment in which the decisions we make about the nature of the digital age will impact generations to come, this dissertation asks, "Are we coding for the future we want?"

Presenter:

Track 2 - ECMA402 Status Report & New Localization Framework from Mozilla

Zbigniew Braniecki
Platform Engineer, Mozilla

JavaScript ecosystem is behind every web application and increasing number of offline applications these days. The Intl API is designed by the working group called ECMA402 which over last year became very active again. In this presentation I will introduce the new additions in the 4th edition of ECMA402, present the vision for the next year and explain how to join the project and contribute to the standard. Mozilla is one of the biggest open source projects of our time. With hundreds of millions of users, desktop, server and web applications and over 100 volunteer localization communities behind it. Based on almost 20 years of experience Mozilla is introducing a full new localization framework and a new paradigm for software localization called Project Fluent. In the second part of the presentation, I will present the vision behind Fluent, how it integrates into existing standards like CLDR, ICU and ECMA402, the scope of the project and the current status.

Presenter:

Track 3 - An Overview of Variable Fonts in OpenType 1.8

Peter Constable
Senior Program Manager, Microsoft

The recent introduction of OpenType 1.8 makes it possible to create a single font that can show very different appearances based on user choices. One font has multiple "axes" allowing a fluid, continuous mutation of glyph shapes. This makes it possible for one font to have multiple weights and stresses, in a compact representation that can save considerable space compared to separate fonts. But OpenType 1.8 allows more than just this mutation, which has been around for 30 years already in the form of Apple's AAT Variations or Adobe's Multiple Masters. Using OpenType 1.8, it is possible, for instance, to make a font with a "Time" axis, where you can show how the shapes of glyphs have changed over time. Choosing alternates via GSUB substitutions can be given a user interface via this mechanism. This talk will go over OpenType 1.8 and all its capabilities, with live demos showing off some fonts of interest.

18:00-19:00 - CONFERENCE RECEPTION

Wednesday, October 18, 2017

09:00-09:50 **SESSION 7**

Presenters:

Track 1 - Implementing Adlam: What Happens After Unicode Adds the Script?

Abdoulaye Barry
Co-Inventor of ADLaM, Winden Jangen

Ibrahima Barry
Co-inventor of ADLaM, Winden Jangen

Craig Cornelius
Engineer, Google, Inc.

In the early 1990s, two young teenagers created a writing system for Pular/Fulfulde, the language of the Fulani people. Their work became ”Bindi Pulaar” and eventually “Adlam,” the “Alphabet that Will Save a People from Disappearing”. The authors describe how Adlam became widespread across many African countries, first as handwriting, then implemented as a font encoding with Arabic code points. Now Adlam is enabling literacy and is growing in use for commerce, education, and publishing. Learn about the experience of adding the script to Unicode, and hear about the benefits and challenges as the newly standardized script meets the technical infrastructure of the Internet.

Presenter:

Manikandan Ramalingam Kandaswamy
Senior Software Engineer, PayPal

Track 2 - Enhancing Date, Time, Timezone Support in Globalize.js

Is Globalize.js on its way to becoming the ICU of the Javascript world? Let’s find out. Globalize.js has been adopted as the internationalization library of choice in Javascript by organizations ranging from Twitter, PayPal, IBM to many startups. Globalize.js has been key in providing i18n features including date, time, number, message formats, plurals, units and relative time. Now, with the latest enhancements, Globalize.js also supports complex date, time and timezone transformations. The library supports IANA and CLDR standards with a tiny footprint one-fifth the size for code and data compared to similar libraries like moment.js. Come find out more about these enhancements and how you can use Globalize for your i18n needs.

Presenter:

Andrew Glass
Senior Program Manager, Microsoft

Track 3 - At the Limits of Complexity: Experiments in Creating OpenType Fonts for Egyptian Hieroglyphs

OpenType was designed to be able to support the writing systems of the world's major languages. The requirements of ancient writing systems were not taken into account at its inception in the mid 1990s. Since that time, Unicode has progressed to encoding more and more ancient writing systems and the rendering needs of these scripts can, on the whole, be supported by the mechanisms that were put in place for complex scripts such as Arabic and Devanagari. Rendering Egyptian Hieroglyphs with OpenType presents interesting challenges when using these existing mechanisms. Correct presentation of Egyptian Hieroglyphs requires the ability to do two-dimensional layout clusters that exceeds the complexity seen in other stacking scripts. Moreover, the large character set of Egyptian Hieroglyphs presents additional burdens on OpenType mechanisms. This talk will describe efforts to create a font that can faithfully render Egyptian Hieroglyphs in a plain text form, allowing for arbitrary structures for two dimensional clusters for a large character set all while staying within the limits of OpenType.

10:00-10:50 SESSION 8

Presenters:

Herman Lookout
Elder Master Speaker of the Osage Language

Mark Pearson
Web Specialist, Osage Nation Language Department

Craig Cornelius
Software Engineer, Google, Inc.

Track 1 - "I Thought We Were Done!" After Unicode Comes to the Osage Nation

Osage is a Siouan language spoken by the Osage people of Oklahoma. Although the last native speaker of the language died in 2005, the Osage Language Program and second-language speakers have worked to revitalize the language and encourage its everyday use. Historically, a variety of ad-hoc Latin orthographies and transcriptions have been used for Osage over the past 210 years. In 2006 the new writing system of Osage orthography was created, and it was further developed and was submitted to the Unicode Consortium in 2015. Osage orthography was released in Unicode version 9.0 in June 2016. The talk highlights efforts of the Osage Nation Language Team to standardize orthography, create fonts, work with technology platforms and vendors, and adapt existing resources to Unicode for wider use and impact on the Osage language and community. Learn about language revitalization as well as the joys and challenges of developing the Unicode proposal. And after standardization comes the hard work with technology vendors, applications, and IT departments, all part of the journey of the Osage language and people.

Presenters:

Roozbeh Pournader
Internationalization Engineer, Google

Seigo Nonaka
Software Engineer, Google, Inc.

Siyamed Sinir
Software Engineer, Google, Inc.

Track 2 - What’s New in Android Text and Internationalization

We will be presenting new features and improvements in text, fonts, typography, and internationalization in the latest version of the Android platform and its accompanying tools and libraries. We will cover features of Android Nougat in more detail (including multilocale and sublocales), present what's new in Android Oreo (XML fonts, downloadable fonts, variable fonts, advanced hyphenation, ...), and explain new and interesting features of the Android Support Library (including EmojiCompat and downloadable fonts) that are useful in developing text-centric or internationalized software.

Presenter:

Nicholas Doiron
Consultant, GeoReactor

Track 3 - Right-to-Left Language Support in OpenStreetMap

This session covers support for right-to-left scripts Arabic and Dhivehi (Maldives) in OpenStreetMap's editor, tiles, and associated data viewers. Recently contributors closed a long-time issue in OpenStreetMap's iD data editor. We added a right-to-left layout for translations of the editing tools. We also developed a text-shaping algorithm to select Arabic letter-forms to get around an SVG bug in Chrome/Webkit, and make Arabic text connect properly. There are existing implementations but with GPL licensing, so we had to create and test a new library. OpenStreetMap has used image tiles for years, but recently developers have rendered their maps in HTML5 Canvas or WebGL.

This presentation will cover supporting bi-directional text in these new map views.

10:50-11:10 - Morning Refreshments

11:10-12:00 **SESSION 9**

Presenter:

Lorna Evans
Script Technologist, SIL International

Track 1 - Beyond Unicode Proposals: Encoding Characters and Scripts is Not Enough!

The timeline from proposing a complex character or complex script to Unicode until we are able to render those characters in an application can take years. During this session we will look at how much time it can take to get a character or script from a Unicode proposal, through the approval process and finally into a released version of Unicode. Secondly, we will discuss text rendering issues with those new characters and scripts in various applications. We examine the current level of complex script support in modern applications in an effort to encourage industry to update rendering engines to support the most up-to-date version of Unicode. We will also examine an added complexity of using glyph variants. Very often glyph variants are needed for different regions of script use or for specific languages. Glyph variants can be supported in OpenType fonts through Character Variants and Stylistic Sets. We will discuss the few applications that support these OpenType features. Finally, we will also review interim solutions that are being used while we wait for applications to support complex text rendering for current versions of Unicode. Our hope is that this presentation will encourage software developers to support the latest versions of Unicode in text rendering engines. Furthermore, adding OpenType Character Variant and Stylistic Set support will benefit many language communities.

Presenter:

Mark Davis
Chief Internationalization Architect, Google Inc.

Track 2 - What’s up with Emoji?

So what has happened with emoji in the last year? (And maybe some hints as to what will happen next.)

Presenter:

Moriel Schottlender
Software Engineer, Wikimedia Foundation

Track 3 - Flipping the Web: How We Support Right to Left Throughout Wikipedia

People around the world read and edit Wikipedia in more than 350 languages, using multi-lingual interfaces that display bidirectional content. One of the bigger and more confusing internationalization challenges we face is that of Right-to-Left support. This lecture describes the strategies we’ve developed to solve these challenges, from the open-source tools we’ve developed to the ways in which we’ve changed the mindset of our libraries so that they use direction-neutral terminology (like ‘forwards’ and ‘backwards’ instead of ‘right’ and ‘left’). It also explains how we intend to achieve our ultimate goal of making RTL support so seamless and transparent that volunteer developers in any country can submit RTL-ready code without even trying. The Wikimedia Foundation, the nonprofit that operates Wikipedia, has addressed multi-lingual and bi-directionality challenges that few organizations have faced, including how to flip the interface automatically based on the chosen language, how to adjust cursor movements in our editor across browsers, and how to make sure our popups appear in the right places on the screen. Wikipedia’s articles are full of multi-lingual content, often containing segments that are multi-directional and written by multiple users in different countries. Content can be multi-directional and appear in a language whose direction is different than the interface language, both of which must still support proper display of the script, the directions, and whatever interface elements the feature holds - in any given combination of directions given.

12:00-13:00 - LUNCH

13:00-13:50	SESSION 10
Presenter: **Luke Swartz** *Product Manager, Google, Inc.*	**Track 1 - Multilingualism at Google: Better Serving Users in More Than One Language** The world is increasingly multilingual: there are more second-language learners of English than native speakers, and even in the supposedly monolingual US, one in five residents speaks a language other than English at home. Google has always strived to make information more accessible across language boundaries, but recently has invested more effort in serving multilingual users. In this talk, we'll discuss some of the challenges and opportunities that multilingualism provides, and some lessons learned from Google's efforts.
Presenter: **Jim DeLaHunt** *Principal, Jim DeLaHunt & Associates*	**Track 2 - Universal Acceptance of Non-Latin Email Addresses and Domain Names: How Does Your Framework Rate?** The next one billion internet users use a wide variety of languages and scripts. They will demand email addresses, and domain names, in scripts they can easily read. App development frameworks, libraries, and programming languages on all platforms will be called on to meet this challenge. This is Universal Acceptance (UA) of all domain names and email addresses, from http://普遍接受-测试。世界 or مانيش @ أشوكا.الهند or données@fußballplatz.technology. We present technical compliance criteria, a list of problem areas, and ways to evaluate compliance. We give our compliance findings so far. Does your library and platform provide Universal Acceptance?
Presenter: **Shawn Xu** *Internationalization Engineer, Netflix Inc.* **Lee Collins** *Internationalization Architect, Netflix Inc.*	**Track 3 - Bidi: A Tale of Two Directions** "it was the spring of hope, it was the winter of despair", this mixed feeling is all too familiar for those who have taken the ride to add bidirectional (BiDi) support for Right-To-Left languages. Enabling stable BiDi display and editing is challenging. Platform implementation of Unicode BiDi Algorithm varies and the idiosyncrasy of each platform got in the way. We'll share some of the challenges we encountered while adding Arabic support at Netflix, the effective ways identified to troubleshoot and fix the BiDi issues on various platforms, and the solutions that we developed, including the server side transformations that were put in place to solve the issues in a more systematic way.

13:50-14:40	SESSION 11
Presenter: **Alolita Sharma** *Principal Technologist, Amazon Web Services*	**Track 1 - Internationalization at Scale** Large organizations with hundreds of teams building front-end features for web and native apps constantly face the pressure of releasing continuously for global audiences. Scaling up to continuously ship at a fast pace for hundreds of countries can be quite a challenge. In this talk, we will discuss the challenges in shipping global-first products scaled up with internationalization best practices, libraries and standards including CLDR, ICU. This talk will also cover recommended guidelines and examples from Twitter, PayPal and Yahoo for app UI design, core code as well as localization readiness.

Presenter:

Nova Patch
Principal Engineer, Shutterstock

Track 2 - Characters for Humans

A character can mean different things to different people, but the largest disparity is between applications and the humans who use them. Programmers aren't to blame, as our programming languages, libraries, and databases provide little or no support for understanding user-perceived characters. Many systems disagree on the basic units of characters, some use code points, others use code units, and others still operate on individual bytes by default. This frequently leads to products with a poor experience in some users' languages, especially written languages that use grapheme clusters, sequences of code points that compose a single user-perceived character. With the rise in global emoji usage and the rapid evolution of standard emoji sequences, this problem is increasingly experienced by users worldwide, regardless of their language. This session will cover: • Extended grapheme clusters and emoji sequences • Programming with these user-perceived characters • Data input, parsing, analysis, formatting, and output • Setting product requirements for character support • Examples from Shutterstock's platforms for content editing and collaboration

Presenter:

Sharon Correll
Software Engineer, SIL International

Track 3 - Using Graphite to Address Challenges in Nastaliq-style Arabic Script

Nastaliq-style Arabic is one of the most complex forms of writing in the world, and standard font technologies have not been quite adequate to handle its sloping, calligraphic form. For this reason, SIL's smart-font technology, Graphite, has been extended with some special capabilities to address the particular challenges of this form of writing. There are two main challenges that arise in developing a Nastaliq font. One is the sheer volume of glyphs that are required, due to the complexity of the calligraphic shapes. Unlike other forms of Arabic, which require initial, medial, final, and isolate forms for dual-connecting characters, most Nastaliq letters potentially require a separate form to precede every other letter of the alphabet, in both initial and medial contexts. This means that the number of glyphs required in the font is at least O(n^2) on the number of base characters. The sloping nature of Nastaliq creates a second, even greater challenge: glyph collisions. A straightforward, naive layout of base glyphs, nuqtas, and diacritics will inevitably result in a rendering where the glyphs collide, forming ugly and even unreadable text. Fixing these collisions is exacerbated by the large number of glyphs and the complex positioning created by the sloping baseline. Workarounds to current font technologies have been used to create Urdu-specific fonts, but these approaches do not scale well when multiple languages and a variety of diacritics are needed. For this reason, SIL International is developing a font called "Awami Nastaliq," specifically intended to support lesser-known languages of west Asia, and using an extended version of Graphite. To solve the problem of collisions, we have enhanced Graphite with an automatic collision-fixing capability. The Graphite engine makes use of a simplified form of the rendered glyphs to detect collisions, shift nuqtas and diacritics, and add kerning to create nicely laid-out text. Besides fixing collisions, the kerning mechanism can also create diagonal overlaps in the sloping text, as Nastaliq is traditionally written. The behavior of the algorithm can be fine-tuned using parameters defined in GDL, the programming language used for creating a Graphite font. BubbleKern, developed by Toshi Omagari, is an approach that shares some similarities with Graphite's collision algorithm. It may be possible to use the BubbleKern algorithm to handle Nastaliq kerning. This would require applying the algorithm at run time rather than building kerning-pair data into the font. BubbleKern does not handle general collisions, however.

In order to provide adequate support for the wide range of languages that use Nastaliq style, some technology comparable to Graphite is needed in a wider range of applications and platforms than currently support Graphite. For instance, there are several approaches that could be used to integrate Graphite's mechanism into OpenType.

14:50 – 15:10 - Afternoon Refreshments

15:10 - 16:00 **SESSION 12**

Presenter:

Tex Texin
Chief Globalization Architect, Xencraft

Track 1 - Agile Internationalization User Stories

User stories are the way that Agile Methodology describes the functionality of the software being developed. Each story describes an action or need of a user and in so doing defines the functions the software must provide and the requirements it must satisfy.

This session will describe the mapping of an internationalization checklist into a suite of user stories that are used in internationalizing a software project.

Presenter:

Mike McKenna
Globalization Strategist

Track 2 - What’s NOT in a Name?

When conducting financial transactions, the regulations of many countries and international treaties dictate that to help with anti-money laundering (AML) and combating the financing of terrorism (CFT), business relations with anonymous clients or clients using fake names is prohibited. So as part of the due diligence to comply in a global marketplace, applications need to have confidence that a name entered has a high probability of being authentic, and are able to quickly detect if a name may be gibberish, fake, or incomplete. This presentation will explore the multi-regional and multilingual aspects of detecting what is not in a name. There are established natural language processing (NLP) techniques to create probabilities of character sequences for determining the likelihood that a word or phrase is written in a specific language. Names are text, but not all text are names. We will explore - Standard NLP - using probability theory with 2-character Markov chains - Sourcing and using regional census and names databases to train good patterns - Examples of fake, gibberish and incomplete names in English - Why English positive and negative patterns do not apply across languages and regions - Using CLDR keyboard maps and regional honorifics to train gibberish patterns - Negative probability of mixed script names

Presenter:

Mohamed Mohie
IT Specialist, IBM Egypt

Track 3 - Enhanced Arabic Collation Support in CLDR and ICU

Arabic characters come in different shapes based on its location in the word “initial, middle, final“ which considered different characters with the current Unicode code points as each shape is represented with a different character. Besides, some Arabic characters like Arabic letter Hamza is represented using different characters based on its diacritics in the word. This makes Arabic collation based on characters code points has many issues, which affect data integration applications and may cause severe issues for matching different data. Enhanced Arabic collation rules handle this situation to overcome the potential issues, new

Arabic collation rules had been implemented in CLDR and picked up by ICU to be used in different products based on the customer requests to meet their requirements.

This presentation will give an overview of the issue, its solution and a demo of the Arabic collation rules “both old and new rules“ using ICU to show the difference in results.

16:10 - 17:00 **CLOSING SESSION**

Moderators:

Martin Dürst
Professor, Aoyama Gakuin University

Alolita Sharma
Principal Technologist, Amazon Web Services

Lightning Talks

This is the third installment of the very successful Lightning Talks from IUC 39 and IUC 40. This closing session will be a series of lightning talks of 5-10 minutes each, followed by extremely short closing remarks. The talks should be related to internationalization, localization, or any other of the topic areas listed in the Call for Participation. This is the chance for you as a conference attendee to present your latest idea or development, spread the word, or raise awareness about something of importance to you, or talk about a topic that doesn't need a full session, or a conclusion or question you are taking home from the conference. If we have any remaining slots, we will also accept proposals during the conference. Questions on any of the lightning talks will be at the end of the session.