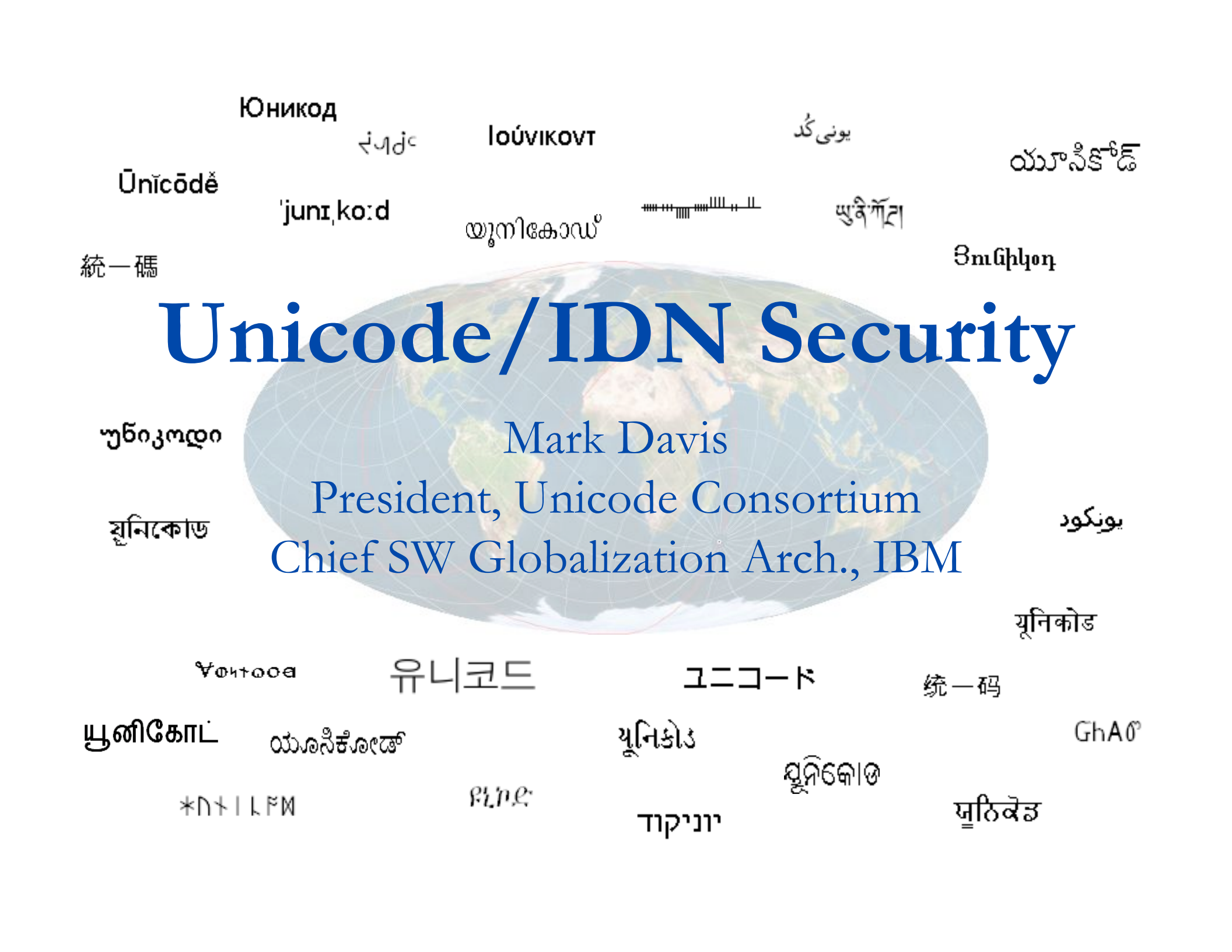

Юникод
Ünĭcōdĕ
'junɪˌko:d
統一碼
Unicode/IDN Security
Mark Davis
President, Unicode Consortium
Chief SW Globalization Arch., IBM
ยูนิโค้ด
유니코드
ユニコード
统一码
GhA0

The Unicode Consortium

- Software globalization standards: define properties and behavior for every character in every script
  - Unicode Standard: a unique code for *every* character
  - Common Locale Data Repository: LDML format plus repository for required locale data
  - Collation, line breaking, regex, charset mapping, …
- Used by every major modern operating system, browser, office software, email client,…
- Core of XML, HTML, Java, C#, C (with ICU), Javascript, …

Security ~ Identity

System A

$X = x$

System B

$X \neq x$

IDN

- You get an email about your paypal.com account, click on the link…
- You carefully examine your browser's address box to make sure that it is actually going to http://paypal.com/ …
- But actually it is going to a spoof site: “paypal.com” with the Cyrillic letter “p”.
- You (System A) think that they are the same
- DNS (System B) thinks they are different

Examples: Letters

- Cross-Script
  - p in Latin vs p in Cyrillic
- In-Script
  - Sequences
    - rn may appear at display sizes like m
    - अ + ा typically looks identical to आ
    - søs looks like søs
  - Rendering Support
    - ä with *two* umlauts may look the same as ä with *one*
    - el□ is actually e + l □

Examples: Numbers

Western	0	1	2	3	4	5	6	7	8	9
Bengali	০	১	২	৩	৪	৫	৬	৭	৮	৯
Oriya	୦	୧	୨	୩	୪	୫	୬	୭	୮	୯

Thus ৪২ = 42

Syntax Spoofing

- http://**example.org**/1234/not.mydomain.com
- http://example.org/1234/not.**mydomain.com**
  - / = **fraction-slash**

Also possible without Unicode:

- http://example.org--long-and-obscure-list-of-characters.**mydomain.com**

UTR #36: Security Recommendations

- General Security Issues (not just IDN)
- V1 approved mid-2005; V2 in progress
  - http://unicode.org/draft/reports/tr36/tr36.html
- Describes the problems, recommends best practices
  - Users
  - Programmers
  - User-Agents (browsers, email, office apps)
  - Registries
  - Registrars

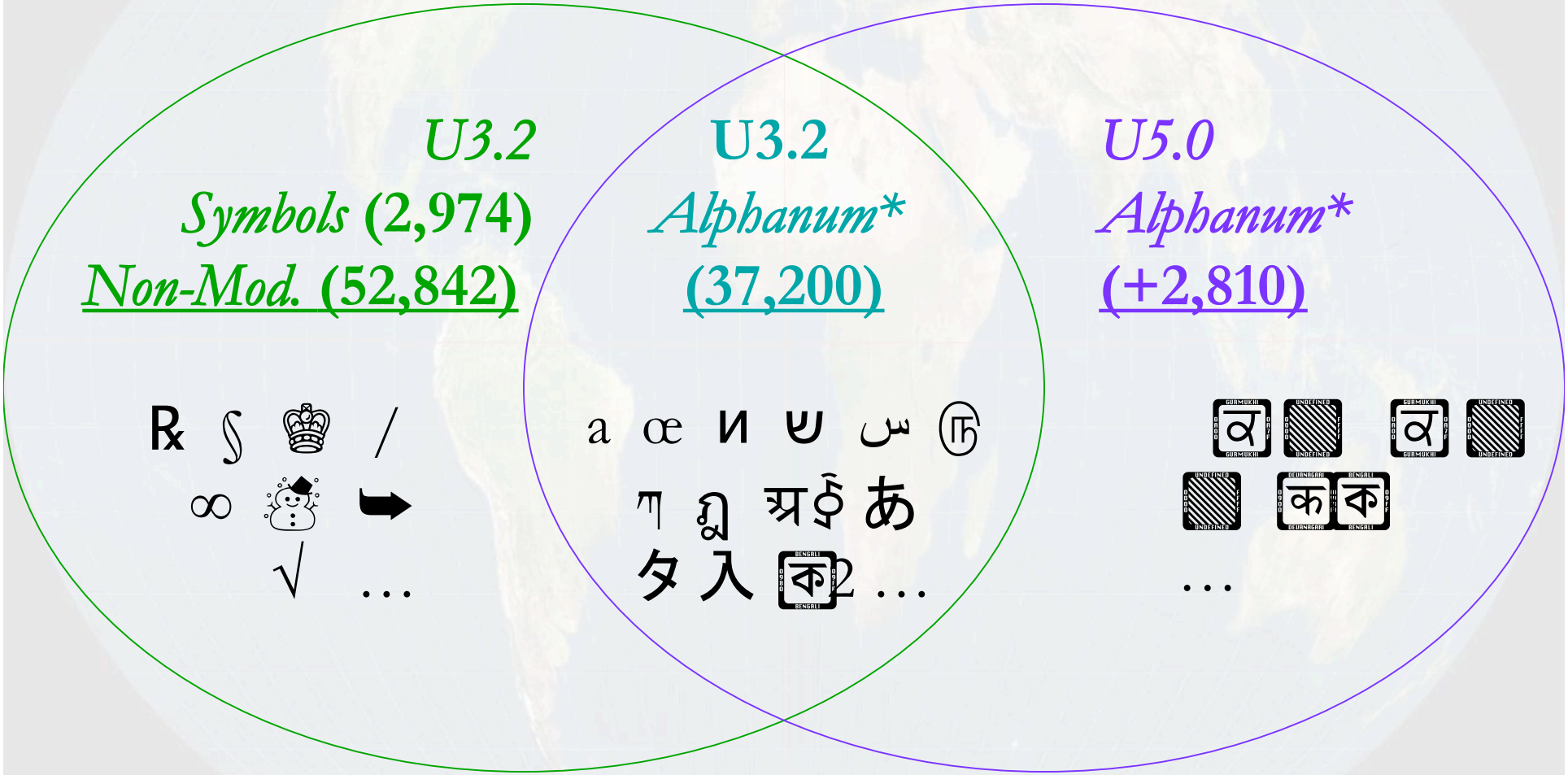
UTS #39: Security Mechanisms

- Supplies data /algorithms for implementations
- Restricted character repertoire:
  - Based on Unicode Identifier Profile
  - Intersect with current NamePrep
- Characters → scripts, confusable characters

Originally in UTR #36 Version 1; split out for clarity

http://www.unicode.org/draft/reports/tr39/tr39.html

Current NamePrep ≠ Unicode Identifiers



http://unicode.org/reports/tr36/idn-chars.html

Restriction Levels*

2. Highly Restrictive

- All characters from a single script, or from limited combinations:
  - *Han* + *Hiragana* + *Katakana*; *Han* + *Bopomofo*; or *Han* + *Hangul*
- No characters in the identifier can be outside of the Identifier Profile
  - includes Letters, Numbers; excludes Symbols, Punctuation,…

3. Moderately Restrictive

- Allow *Latin* with other scripts except *Cyrillic*, *Greek*, *Cherokee:*

  ip-アドレス.co.jp خدمة-rss.eg

4. Minimally Restrictive

- Allow arbitrary mixtures of scripts:

  sony-βίντεο.gr xml-документы.ru

  игро-shop.com …

** Subject also to restrictions on confusables*

ICANN Guidelines v2

http://icann.org/general/idn-guidelines-14nov05.htm

- Improvement on v1,… ***but*** needs new revision:
- *Procedurally*
  - Insufficient time for thorough review
  - The disposition (with rationale) of comments not available
  - Only single cycle of public review
- *Technically*
  - Any specification needs a much clearer structure – the exact implications of a claim to adhere to the guidelines are currently impossible to measure, and useless for security
  - #3 (script/language limitations) has far too many loopholes.
  - #4 (symbols) is too permissive, and not well-defined
  - #5 (registration) should use the post-nameprep'ed form

Guideline 3 (lang./script limitations)

- Associate with script *except* with language and script, or *except* with set of languages, or *except* with "more than one designator"
- Publish set of code points, define variant code points; indicate script/language.
  - **Why language? (too fuzzy to be testable)**
  - **Why script? (derivable from characters)**
- Single script in label, *except* when language requires, *except* with mixed-script confusables, *except* with "policy & table" defined.
  - **Who decides when required?**
  - **Allows single-script confusables.**
- All registry policies documented and publicly available, with table for each set of code points
  - **Machine readable? Discursive description?**

Guideline 4 (disallowed symbols)

- Line symbol-drawing characters (as those in the Unicode Box Drawing block)
  - **One small set of the *many* symbols**
- Symbols and icons that are neither alphanumeric nor ideographic language characters,
  - **Numbers? Combining Marks? Letter modifiers? Kana length mark? Ill-defined, untestable.**
- Characters with well-established functions as protocol elements
  - **/ is confusable with a "protocol element" but isn't one. Ill-defined, untestable.**
- Punctuation marks used solely to indicate the structure of sentences
  - **Em-dash? Who decides? Ill-defined, untestable.**
- Punctuation marks that are used within words *except* "essential to the language" & "associated with explicit prescriptive rules"
  - **Ill-defined, untestable.**
- *Except* "under corresponding conditions, a single specified character may be used as a separator within a label, … by designating a functionally equivalent punctuation mark from within the script."
  - **Ill-defined, untestable.**

Guideline 5 (registration)

- A registry will define an IDN registration in terms of both its Unicode and ASCII-encoded representations.
  - Should use *output* Unicode representation (after mapping and normalization): otherwise many more visually confusable characters are present
  - Should say "ACE", not ASCII.

Unicode Recommendations

Precise Specification, Mechanically Testable:

- **Guideline 3 (script/language limitations) →**
  - Publicly document the Restriction Level being enforced (≤ Level 4)
  - Publicly document the enforcement policy on confusables: whether any two domain names are allowed to be whole-script or mixed script confusables according to [UTR39].
- **Guideline 4 (symbols) →**
  - Only characters in *IDN Security Profiles for Identifiers* [UTR39].
- **Guideline 5 (registration) →**
  - Define an IDN registration in terms of its:
    - **Nameprep-Normalized** Unicode representation (*output* format)
    - ACE representation

Work with IETF to update NamePrep to Unicode 5.0 (+)

Backup Slides

Agenda

- Unicode Background
- Security Issues

Domain Names

	String	UTF-16	Internal - IDNA
1a	a¨t.com	0061 0308 0074 002E 0063 006F 006D	xn--t-zfa.com
1b	ät.com	00E4 0074 002E 0063 006F 006D	xn--t-zfa.com
2a	tOp.com	0074 03BF 0070 002E 0063 006F 006D	xn--tp-jbc.com
2b	top.com	0074 006F 0070 002E 0063 006F 006D	top.com
4a	søs.com	0073 006F 0337 0073 002E 0063 006F 006D	xn--sos-rjc.com
4b	søs.com	0073 00F8 0073 002E 0063 006F 006D	xn--ss-lka.com

Non-Visual Attacks

- Exploiting Expectations
  - Collation:
    - X < Y, so X + H < Y + H *wrong*
  - Casing:
    - len(X) = len(toUpper(X)) *wrong*
  - Encoding:
    - '/' is always represented by $2F_{16}$ *wrong*

UAX #31: Identifier & Pattern Syntax

- For identification of entities (programming variables, resources, domain names, ...
- Appropriate characters -- stable across versions
- ***Not*** all natural language words:
  - *can't*
  - *U.S.A.*
- Provides *Foundation*: specifications can "tailor" it for different environments: adding or removing characters.

“StringPrep” Processing

- Map

  A → a

- Normalize

  c + ¸ → ç

  ㄱ + ㅏ → 가

  ｶ → カ

  ﬁ → f + i

- Prohibit

  & / . , …

UAX #15: Unicode Normalization Forms

- Normalizes most visually confusable sequences to unique form

c + ¸ → ç

ㄱ + ㅏ → 가

ｶ → カ

ﬁ → f + i

- Core part of StringPrep, other Identifier Profiles