インターネットにおける識別子文字列の国際化について

2015年9月24日

株式会社日本レジストリサービス
米谷嘉朗
<yoshiro.yoneya@jprs.co.jp>

もくじ

- 識別子の国際化とは
- IDN
- EAI
- precis
- lucid
- lager

識別子の国際化とは

識別子とは

- **識別子**（しきべつし、英: identifier）とは、ある実体の集合の中で、特定の元を他の元から曖昧さ無く区別することを可能とする、その実体に関連する属性の集合のこと[1]をいう。ほぼすべての情報処理システムで何らかの識別子が使われており、識別子を利用することで機械的な処理が可能になる

  [1] ISO/IEC 25760-1:2011 3.1.1~3.1.4
  <http://ja.wikipedia.org/wiki/%E8%AD%98%E5%88%A5%E5%AD%90>から引用

- 識別子の例
  - 実社会：電話番号、住所、パスポート番号
  - インターネット：IPアドレス、メールアドレス、URI

インターネットでの識別子

- コミュニケーションする相手を特定するもの
  - 端末、ホスト、サービス、アカウント
- 識別子の形式はプロトコルによって異なる
  - 使用文字種、文字列の構成
  - 例:

    ftp.example.jp

    user@example.jp

    https://example.jp/

文字セットとエンコーディング

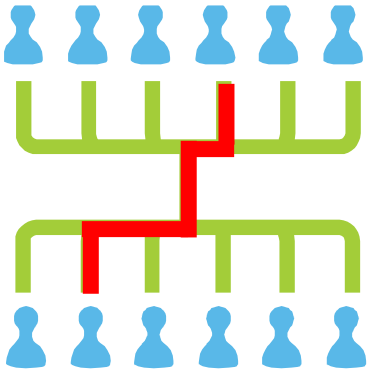
- 文字セット(文字集合)は、コンピュータで扱う文字の種類やその範囲と、各文字の固有番号(文字コード)を定めたもの
  - 文字セットの例

    ASCII
    EBCDIC
    JIS X 0213:2012
    ISO/IEC 8859
    Unicode
- エンコーディングは、文字コードをコンピュータが扱いやすいビット列に変換する符号化方式のこと
  - エンコーディングの例

    ASCII
    Shift_JIS、EUC-JP、iso-2022-jp
    UTF-8、UTF-32、Punycode

プロトコルとは

- 複数の人がコミュニケーションをするときの共通の取り決め

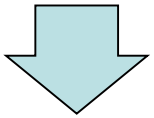


- 取り決めを守らないと情報が正しく伝わらずコミュニケーションが成立しない

「こんにちは」⇔「縺薙s縺ォ縺。縺ッ」

標準化とは

- インターネットでは世界中の不特定多数の人がコミュニケーションする



誰もが共通に守るプロトコルが必要

標準化

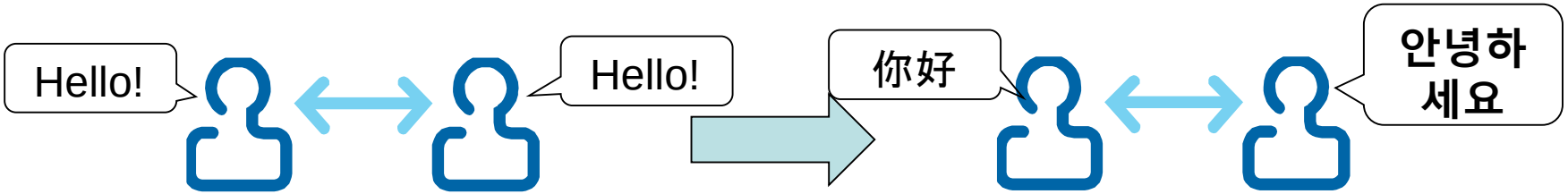
- インターネットのプロトコル標準化を行うIETF(Internet Engineering Task Force)
  - IP(v4、v6)、TCP、SMTP、HTTP、DNS他
  - RFC(Request For Comments)

国際化とは

- 国際化は、OSやアプリケーションを複数の国や地域で使えるようにするための枠組み
  - Internationalization、I18N
- 地域化は、国際化の枠組みを使い特定の国や地域の特色を反映したもの
  - Localization、L10N
  - IME、単位表示、行末処理など
- 多言語化は、国際化の枠組みを使い複数の言語を同時に扱えるようにしたもの
  - Multilingualization、M17N
  - ダイアログメッセージなど

プロトコルの国際化とは(1/2)

- プロトコルで使える文字を各種言語の文字に拡張すること
  - 初期のインターネットではASCII(英数字)のみ



- プロトコルの中でどのような文字をどのような形式で使えるようにするのかを決めること
  - ドメイン名、メールアドレス、URI
  - Unicode、JIS X 0208:2012、ISO/IEC 8859-1
  - Punycode、UTF-8、iso-2022-jp

プロトコルの国際化とは(2/2)

- 国際化ドメイン名
  - Internationalized Domain Name; IDN
  - 2003年にRFC 3454,3490,3491,3492で規定され、2010年にRFC 5890,5891,5892,5893,5894,5895で更新された
- 国際化メールアドレス
  - Email Address Internationalization; EAI
  - 2007年～2008年にRFC 4952,5335,5336,5337,5504,5721,5738, 5825,5983で実験的に規定され、2012年～2013年にRFC 6530,6531,6532,6533,6855,6856,6857,6858で標準化された
- 国際化URI
  - Internationalized Resource Identifier; IRI
  - 2005年にRFC 3987で規定され、現在W3Cで改定作業が進められている

プロトコルの国際化における問題とその解決 (1/3)

- プロトコルが伝達するコンテンツは文字セットとそのエンコーディングが決まっていればよい

  - Ex: UnicodeでUTF-8を使う

プロトコルの国際化における問題とその解決 (2/3)

- 相手を特定するなど、通信を制御する部分を国際化する場合はプロトコル要素(文字列)の比較一致を正確に行う必要がある
  - 大文字小文字 (A ⇔ a)
  - 合成文字 (が ⇔ か゛)
  - 全角半角 (ア ⇔ ｱ)
- 利用者からは同一に見えるプロトコル要素の比較一致が否となった場合の問題
  - コミュニケーションの不成立
  - 不正アクセスの誘導

プロトコルの国際化における問題とその解決 (3/3)

- 比較一致を正確に行うために
  - 文字の正規化や使える文字の定義を行う
    - 大文字を小文字に変換したり全角を半角に変換したり
    - 記号文字は使えなくしたり
  - プロトコルごとに正規化の定義や使える文字の定義は異なる
    - ただし基本的な考え方は共通のはず
    - IETF precis WGで共通部分の定義を実施(後述)

IDN
(**I**nternationalized **D**omain **N**ame)

IDNの背景

- 1990年代の前半に、メールやWebコンテンツの多言語化が進められた
  - MIME(Multipurpose Internet Mail Extensions)の標準化と普及
- 1990年代の後半に入り、Webアクセスの多言語化が要求されるようになった
  - コンテンツだけでなく、アドレスも多言語で表現したいという要求
  - 1998年頃からアジア圏を中心に技術検討が始まった
  - 2000年にIETFでIDN WGが設立された

IDNのチャレンジ

- IETFで最初の識別子(プロトコル要素)を国際化する活動である
  - プロトコルを国際化するためには何が必要かという議論から始められた
- 地域化要求を分離した
  - 英字の大文字小文字を同じ文字として扱うように、漢字の繁体字簡体字を同じ文字として扱いたいという中国語圏からの要求があった
    - 例：國と国
    - プロトコルでは地域化は扱わないこととし、登録時の運用規則で対応することとした
- 特許クレームに対応した
  - ドメイン名バブル期であり、IDNで商売をしようとしていたベンチャーが特許クレームを出してきた
    - 特許に抵触しない方式を採用するようにした

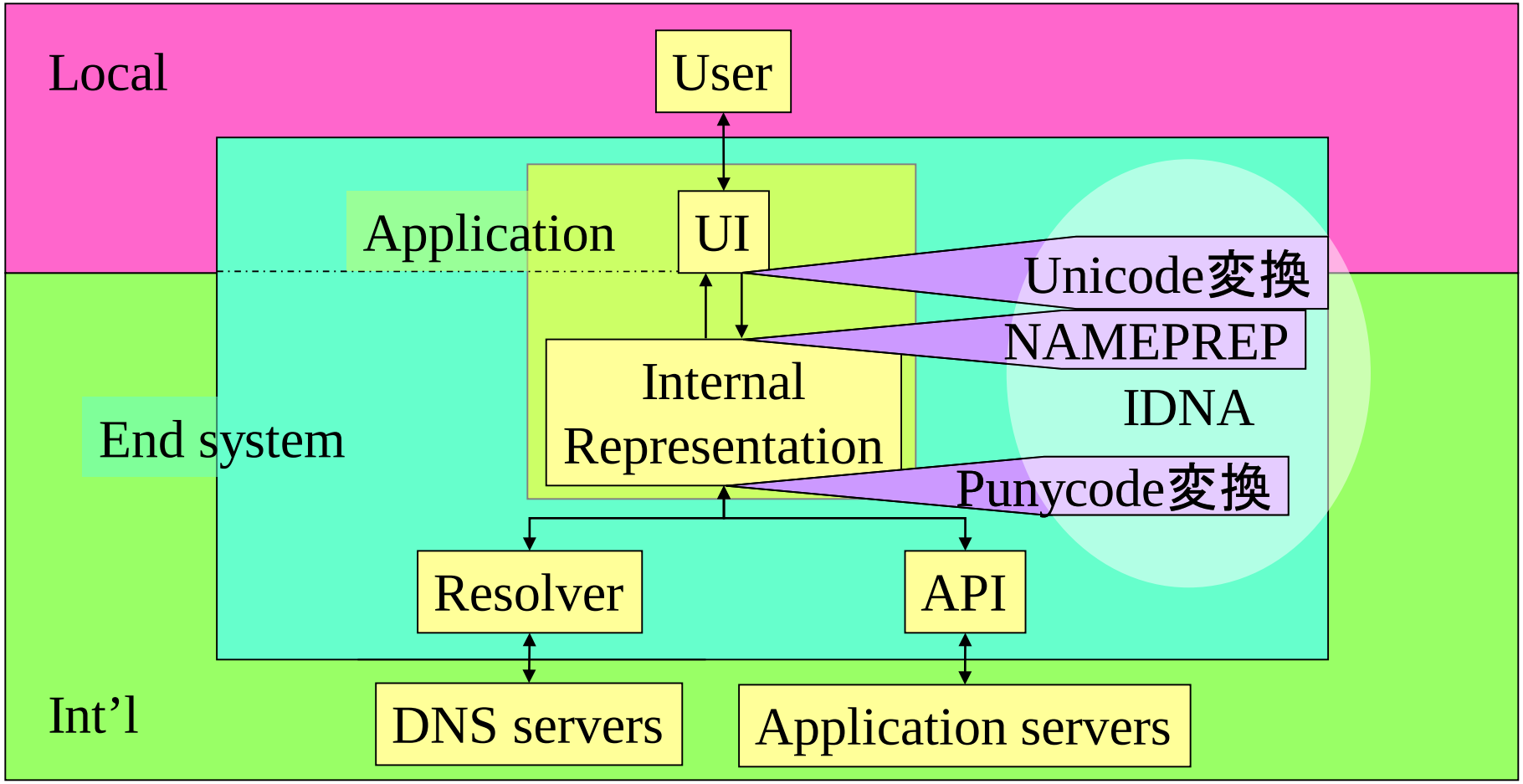
IDNの方式(概要)

- IDNで使用する文字セットはUnicodeとする
- IDNはドメイン名のラベル単位で処理する
  - ドメイン名構造に影響を及ぼさない
- IDNは必ず正規化する
  - 同じ文字の表現形式を揃え比較一致が正しく行われるようにする
- IDNのネットワーク上のエンコーディングはASCII互換エンコーディング(ACE)を使う
  - 既存のシステムに影響を及ぼさない
- IDNの処理はアプリケーションが行う

IDN標準化の成果(2003年版)

- IDNAアーキテクチャ
  - IDNをアプリケーションで処理するというアーキテクチャ(RFC 3490)
- Unicode文字列の正規化の仕組み
  - 複数のプロトコルが共通して使えるフレームワークであるStringprepの規定(RFC 3454)
    - Unicodeのバージョンは3.2.0を指定
  - StringprepをIDNに適用するためのNAMEPREP (RFC 3491)
- IDNを効率的にACEに変換するアルゴリズム
  - Punycode (RFC 3492)

IDNA2003のアーキテクチャ



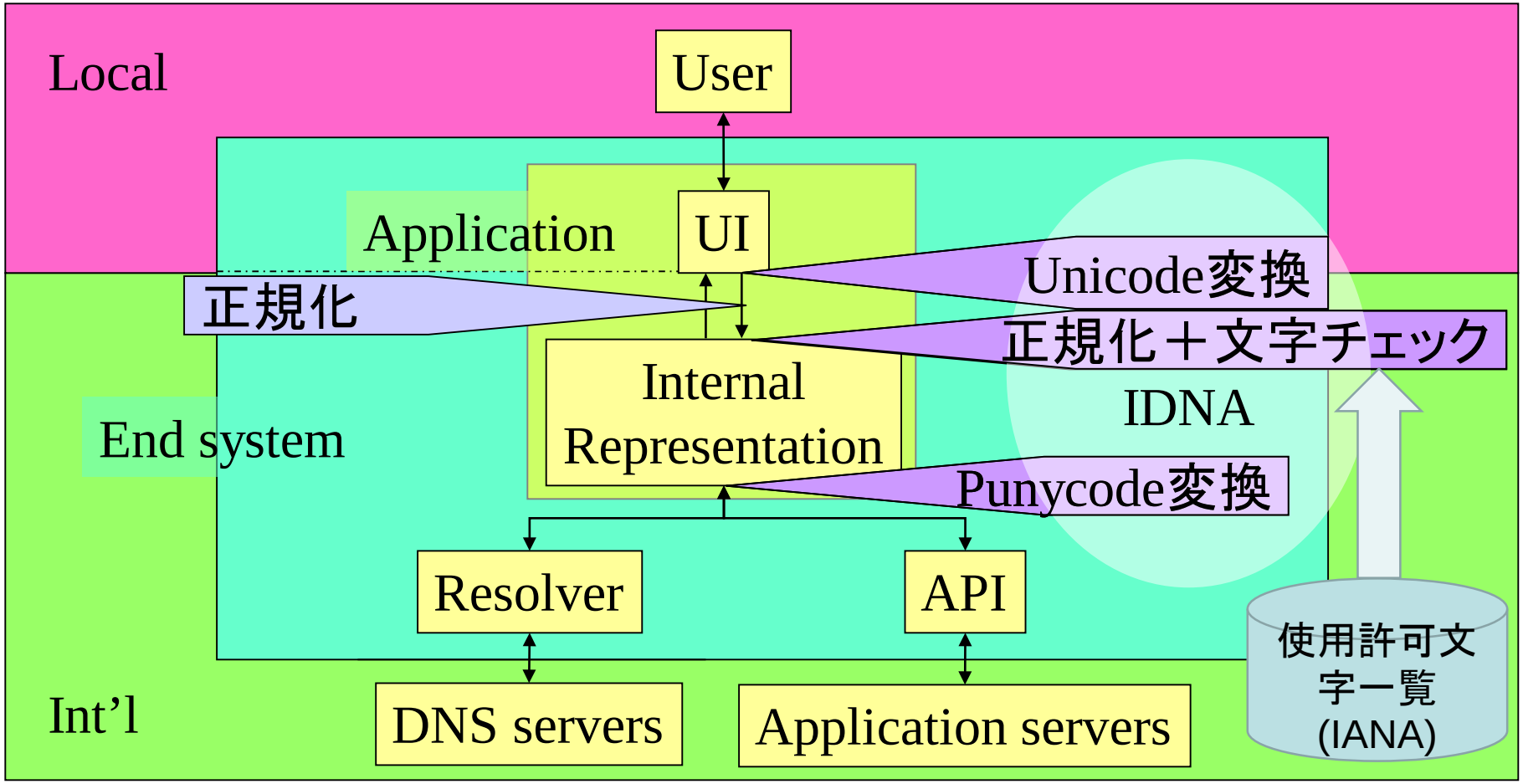
IDNA2003の課題

- IDNA2003が標準化され、運用が始まると以下の課題が明らかになった
  - Unicodeの改版への追従性
    - Unicodeは頻繁に改版され新しい文字の追加などが行われるが、IDNA2003はUnicodeのバージョンを固定している
  - 正規化方式の不適
    - ドイツ語やギリシア語にはIDNA2003の正規化方式が不適切な文字がある
    - プロトコルが正規化を含むことでIDN文字列とACEの相互変換に一意性が保証されない
  - 除外方式の弊害
    - IDNA2003は空白文字など一部文字は使用を禁止されている(除外方式である)が、その他の文字は許可されているため罫線記号や数学記号などドメイン名には不適切な文字が使用できる

IDNA2008による改訂

- IDNA2003の課題を解決するため以下の改定が行われた
  - Unicodeバージョン依存部分の外部化
    - 依存部分を外部パラメータ化しIANAに登録するようにした
  - 正規化方式の変更
    - 正規化処理をプロトコルから外し、正規化されていることをチェックするだけとした
  - 許可方式への変更
    - IDNで使える文字をドメイン名で使用するのに適切な文字(言語を表現するのに必要最小限な文字)に制限し使用許可文字一覧をIANAに登録するようにした

IDNA2008のアーキテクチャ



運用で解決している課題

- 似た文字(homograph)問題
  - Unicodeは多数の文字を含んでいるため、見た目が似ている文字(homograph)が多数ある
    - A(大文字エー)とA(ギリシア文字大文字アルファ)
  - 異なる用字(Script)を混在させるとこの問題が顕著となるため、IDN登録を受け付けるTLDレジストリは受け付ける言語とその言語での文字範囲をIANAに登録している(IDNテーブル)
- 異体字(Variant)問題
  - 中国語の繁体字簡体字など、文字は異なるが発音・意味が同じ文字を、同じ文字とみなして同一登録者に紐付けている(IDNテーブル)

実装状況

- いまや、ほとんどのブラウザはIDNを実装している
  - ただし、IDNA2003対応にとどまっているものがほとんど
  - 互換性のためと思われる
- IDN TLDは既に90以上がRootゾーンに存在
  - 2012年からの新gTLDプログラムで増加
- IDN登録を受け付けるTLDは260以上

EAI

(**E**mail **A**ddress **I**nternationalization)

EAIの背景

- IDNと同様に、母国語を使ったメールアドレスを使いたいという要望はもともとあった
  - 特に、アルファベットに馴染みのない中国語圏、アラビア語圏での要望が高かった
  - ヨーロッパの非英語圏からも母国語の文字が使えるなら使いたいという要望があった
- 2006年にIETFでEAI WGが設立された

EAIのチャレンジ

- ASCII互換エンコーディングを使わず、メールアドレス、メールヘッダに直接Unicode(UTF-8)を使えるようにする
  - IDNと違い、EAIはアプリケーションだけでなくメール中継系(SMTPサーバ)やメールボックスアクセス系(POP/IMAPサーバ)での対応が必要
  - 対応するなら、一貫性のある(すべてUTF-8の)世界がよいという考えかたに基づく
- 原則、下位互換性は確保しない
  - 関連する系が多く複雑になりすぎるため
  - 例外はPOP/IMAP

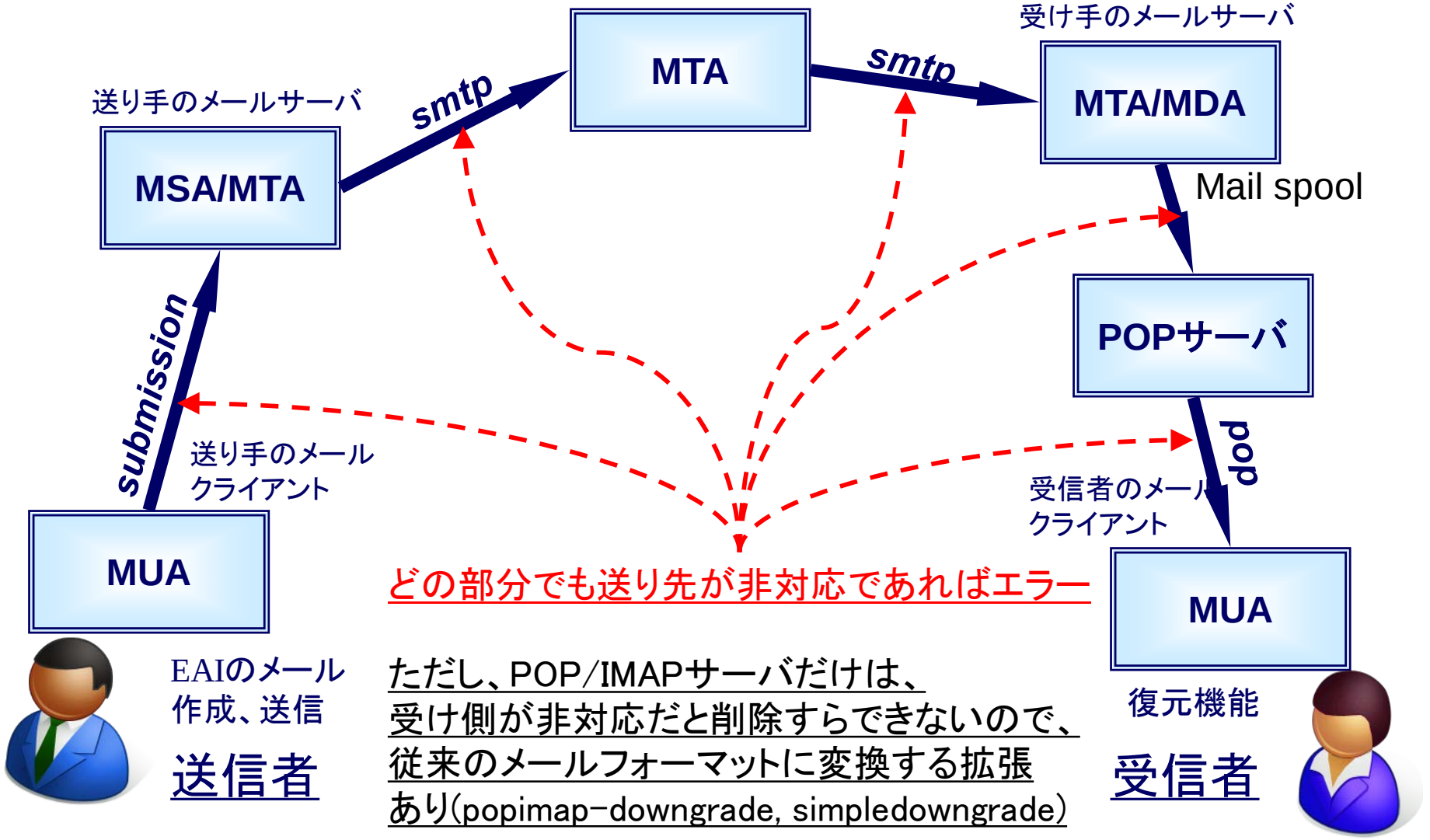
EAIの方式(概要)

- メールアドレスおよびメールヘッダで使用する文字セットはUnicodeとする
- Unicodeのエンコーディングは UTF-8とする
- メール配送系の途中に、EAIに対応していないサーバがあったらそこで配送はエラーとする
- ローカルパート(@の左側)の正規化は規定されていない
  - Unicode規定の正規化を適用することは推奨されている
- メール本文もUTF-8とする

EAI標準化の成果(2012-13年版)

- EAI概要と枠組み(RFC 6530)
- SMTPの拡張(RFC 6531)
- ヘッダフォーマットの拡張(RFC 6532)
- 配送状況・開封通知の拡張(RFC 6533)
- IMAPの拡張(RFC 6855)
- POP3の拡張(RFC 6856)
- POP/IMAPのダウングレード(RFC 6857)
- POP/IMAPの簡易ダウングレード(RFC 6858)

EAI通信モデル



実装状況

- GmailがEAIの送受信に対応
  - EAIアカウント作成には未対応
- CoremailがEAIに対応
- PostfixがSMTPUTF8に対応

precis

(**Pre**paration and **C**omparison of **I**nternationalized **S**trings)

precisの背景

- IDN2003の成果であるStringprepを使って識別子を国際化したプロトコルがいくつかある
  - iSCSI、EAP、XMPP、SASL、LDAPなど
- IDNA2003の改訂版であるIDNA2008はStringprepを使わずにUnicodeバージョン依存性を排した
  - Stringprepは改訂されていない
- Stringprepを使っているプロトコルはいまだにUnicodeバージョン依存性がある
  - 新しいUnicodeに対応したいという要求がある

precisのチャレンジ

- Stringprepのような識別子国際化のためのフレームワークを提供する
  - Unicodeの改版への追従性を持つ
  - 許可方式とする
  - 正規化やマッピングはプロトコルから除外しない
- Stringprepとの下位互換性は完全には確保しない
  - 別途、StringprepからprecisへのIの移行ガイドラインを作成することで対応する

precisの方式(概要)

- 識別子を表す文字としてUnicodeを使用するプロトコルに適用する
- precisを適用するプロトコルは適用方法について以下を決める
  - 文字列クラス
    - 制限的なIdentifierClassか許容的なFreeformClassか
  - マッピング
    - 文字幅、スペース、文字種などを変換するか
  - 正規化
    - Unicodeが規定するNFD、NFKD、NFC、NFKCのどれを使用するか(推奨はNFC)
  - 方向性ルール
    - 右から左に書く文字が含まれているときの扱いをどうするか（RFC 5893を適用するか)

precis標準化の成果

- 課題定義(RFC 6885)
- フレームワーク(RFC 7564)
- ユーザ名とパスワード(RFC 7613)
  - SASLprep改訂版(プロファイル)
- ニックネーム(draft-ietf-precis-nickname)
  - RFC Editor Queue
  - 表示名など(プロファイル)
- マッピング(draft-ietf-precis-mappings)
  - IESG評価中
  - フレームワークの補足(Informational)

precisの残作業

- Stringprepからの移行ガイドラインの作成
  - ボランティア募集中
- Stringprepを使っている既存プロトコルのprecisへの移行（プロファイルの作成）
  - できるだけ既存のプロファイルを適用することが推奨されている

lucid

(**L**ocale-free **U**ni**c**ode **Id**entifiers)

lucidの背景

- Unicode7.0の改訂で正規化(NFC)により合成されない文字が追加された
  - 従来のIETFの前提が崩れた
- IETFでどのような対応をすべきかIABで議論されステートメントが出された
  - <https://www.iab.org/documents/correspondence-reports-documents/2015-2/iab-statement-on-identifiers-and-unicode-7-0-0/>
- IETFでの課題の共有と検討の方向性が議論された
  - lucid BoF@IETF92

lucidの方向性

- 案1：ルールを変える
  - Unicodeに新しい文字プロパティを定義してもらう
  - IETFで新しい正規化ルールを作る
- 案2：ガイドラインを作る
  - プロトコルの変更は行わない
  - 注意すべき文字があり、それは使うべきではないというガイドラインを提示する

lager

(**La**bel **Ge**neration **R**ules)

lagerの背景

- 2012年のICANN新gTLDプログラムの開始
  - 1930件の申請があり、そのうち116件がIDN
  - 文字列の類似性を含む混乱の危険性はパネル(人間)が判断
- IDN TLDはさまざまな言語・用字(Script)で申請されるため、TLDが登録されるRootゾーンにはさまざまな言語・用字が混在する
  - 言語によっては、異体字(字形・コードポイントは異なるが同じ読み・意味の文字)が存在することがある
  - 用字を共有する言語間であっても、異体字の定義が異なることがある
- 次の新gTLDプログラムに向けて、Rootゾーンで、さまざまな言語・用字および異体字を統一的に取り扱うルール(Root zone Label Generation Rules; RootLGR)を決めておく
  - 文字列の適切さや異体字による派生を自動的に判断するため
  - ルールはインターネット標準として定める

lagerのチャレンジ

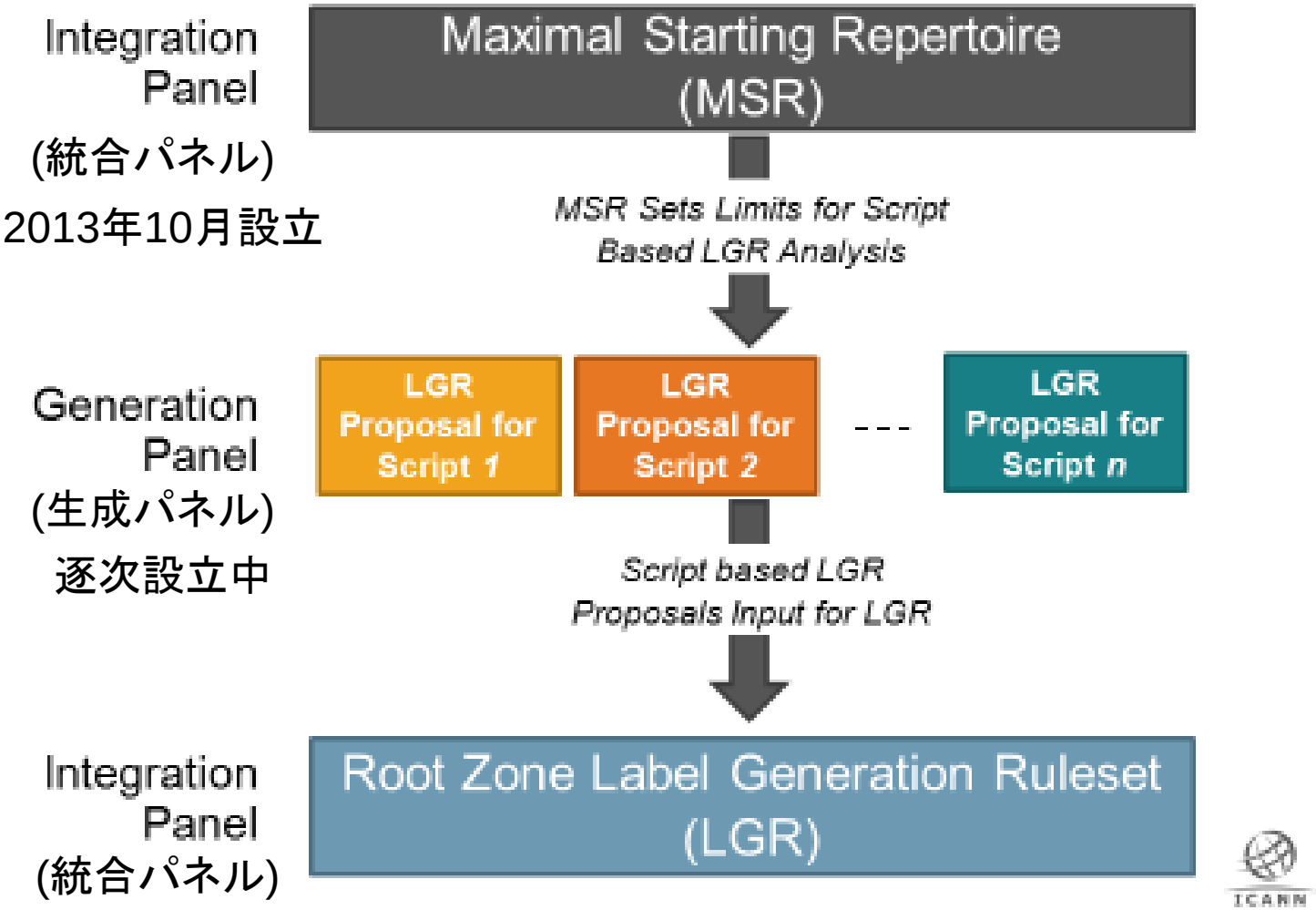
- LGRの汎用フォーマットを決める
  - TLD(RootLGR)だけでなく、ドメイン名の各レベルで使用可能とし、既存のIDNテーブルを置き換えられるようにする
  - XMLで記述し機械処理できるようにする
- 言語・用字ごとに以下を記述できるようにする(後述)
  - ラベルとして申請可能な文字の範囲
  - 申請可能な各文字に対する異体字
  - 各異体字の登録可不可を示す異体字タイプ ★
  - ラベル評価ルール ★

★は従来のIDNテーブルでは十分に記述できない属性

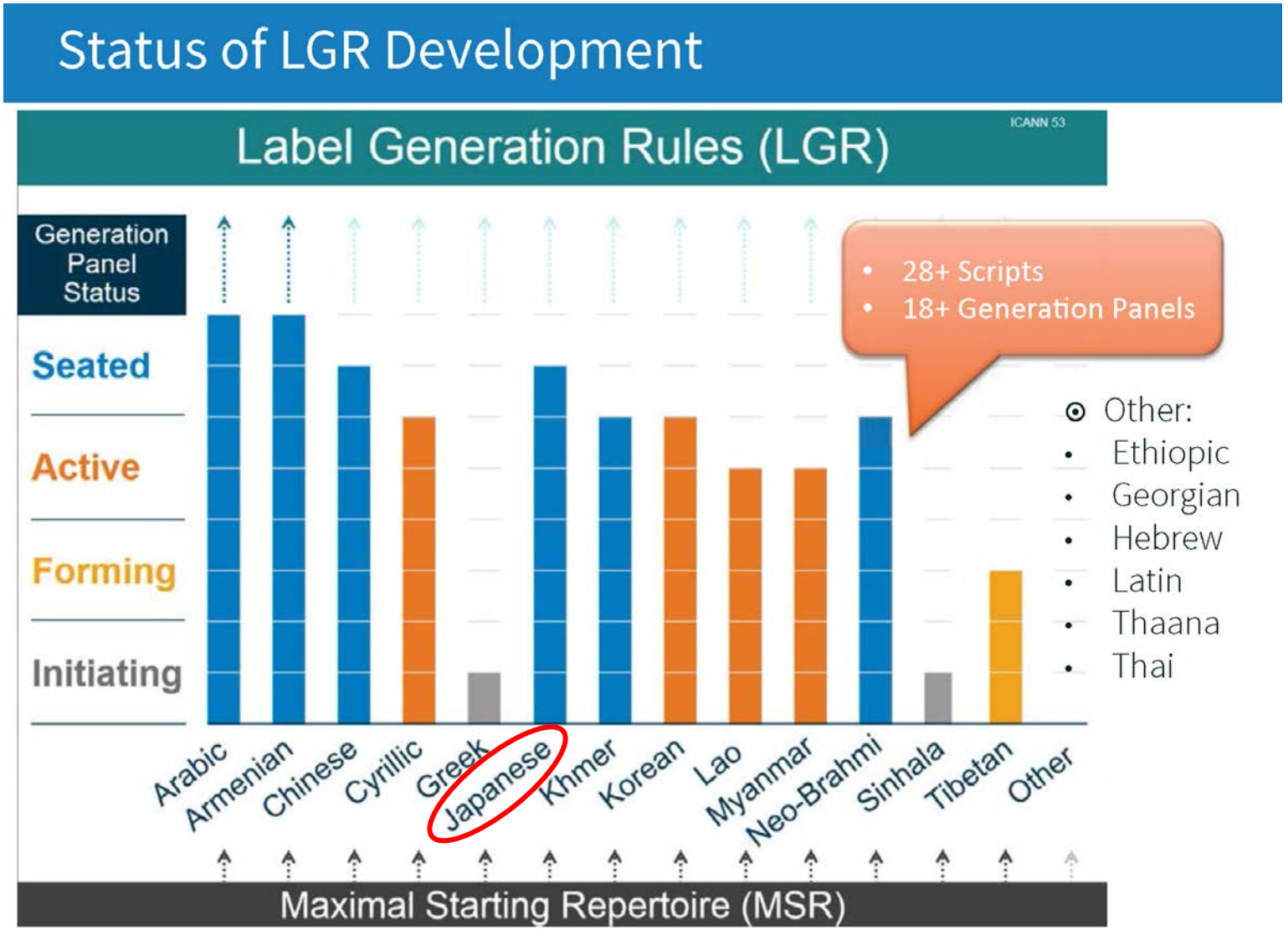
RootLGRとの関係

- RootLGRはlagerのフォーマットを使って言語・用字ごとに以下を定義する(現在のlagerの対象)
  - 文字範囲
    - TLDとして申請可能な文字の集合
    - JIS X 0208:2012の第1水準・第2水準漢字など
  - 文字ごとの異体字の定義
    - 「国」に対する「國」「圀」など
    - 定義することは必須ではない
  - 各異体字の異体字タイプ
    - 「国」「國」はRootゾーンに登録可能だが「圀」は登録不可とするなど
    - 異体字がなければ異体字タイプは定義する必要がない
  - ラベル評価ルール
    - 申請された文字の組み合わせ全体を評価するルール
    - 長音(ー)や踊り字(々)が文字列の先頭にあってはいけないなど
- RootLGRは言語・用字ごとの定義を統合する(将来のlagerの対象？)
  - 原則は和集合
  - 機械的な統合ではなく、用字を共有する言語ごとに調整を行う

RootLGR開発プロセス



各国の言語生成パネルの状況
(2015年6月現在)



設立済

活動中

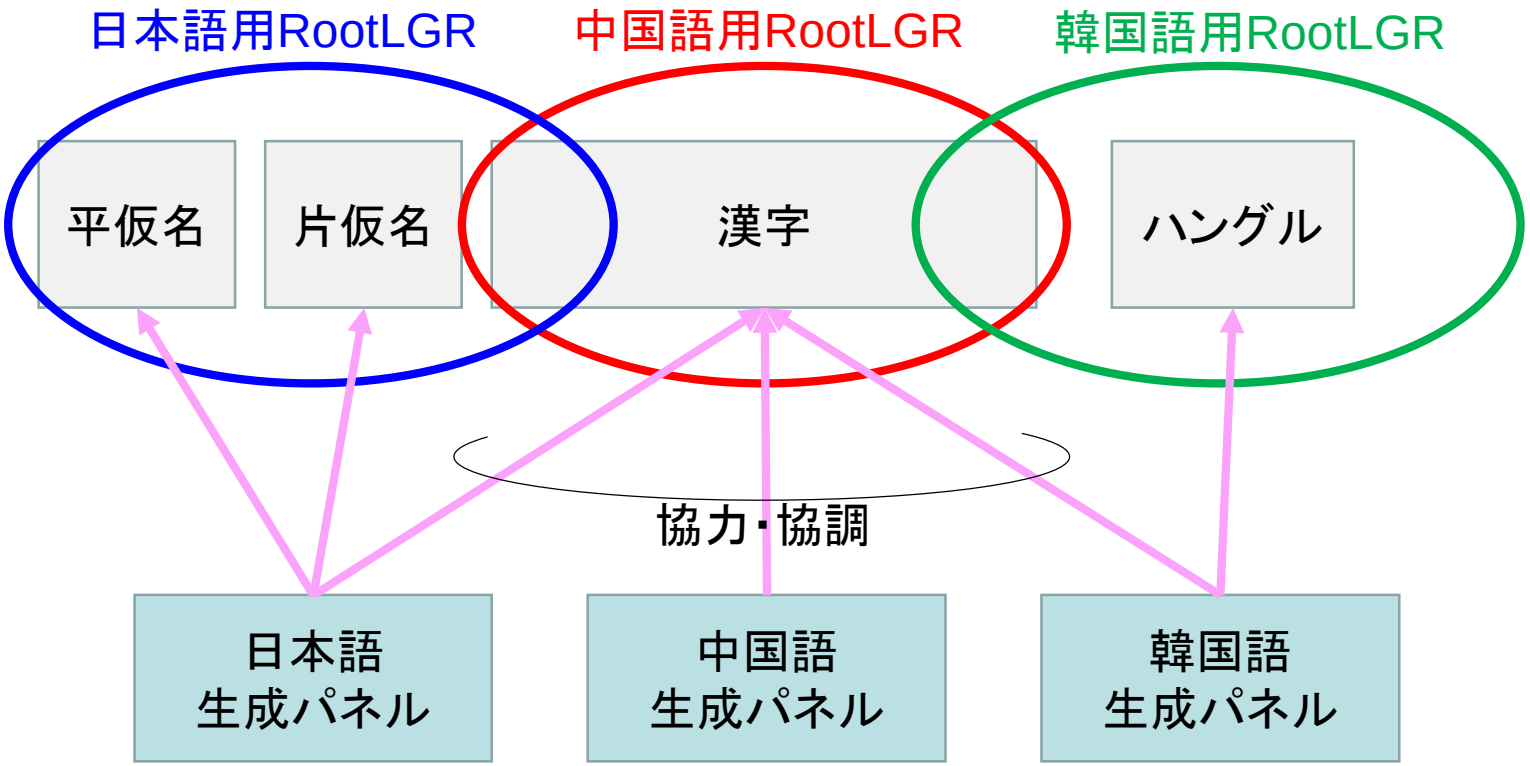
設立中

設立準備中

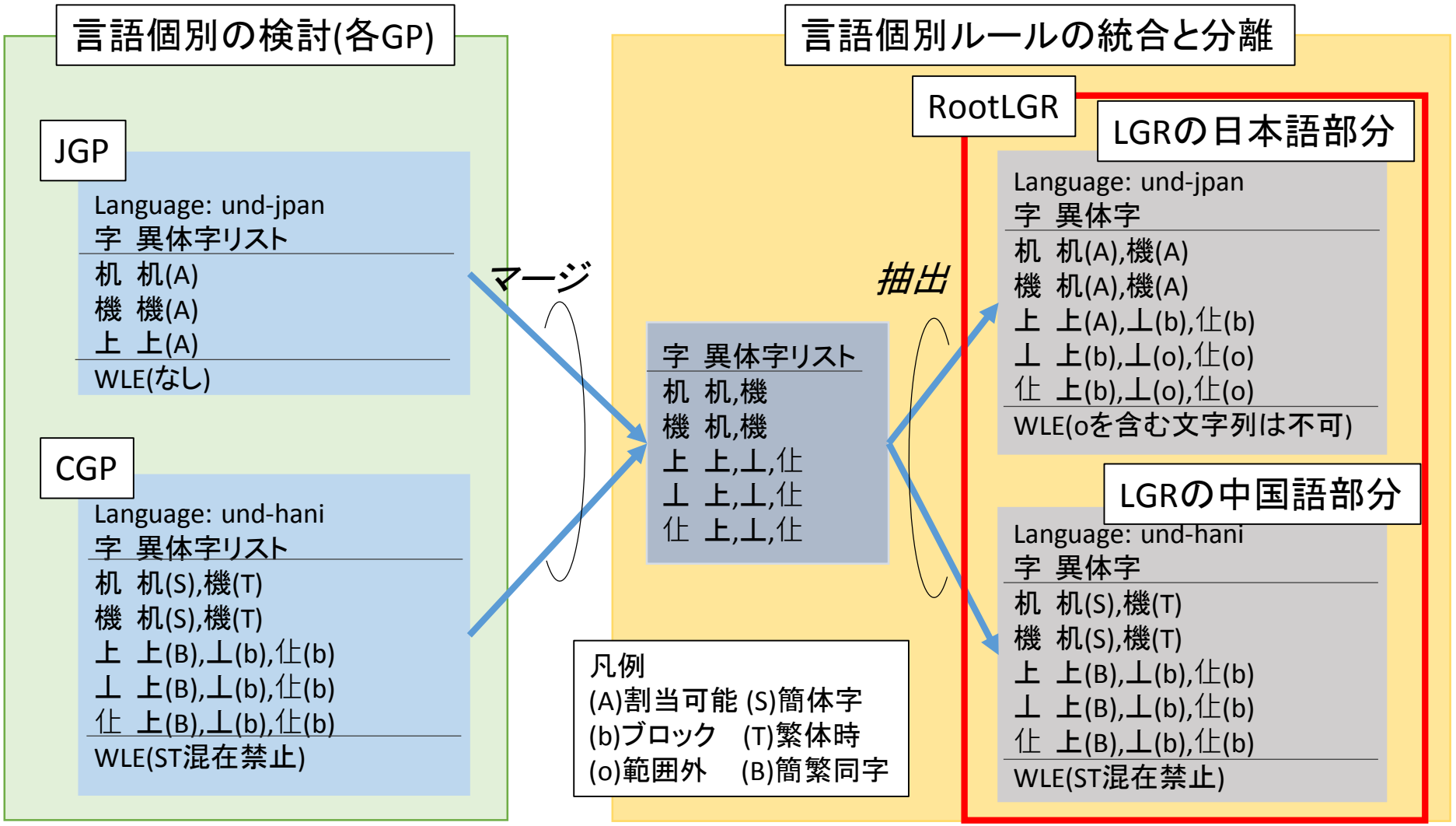
日本語生成パネルの作業

- 統合パネルに提案する日本語用RootLGRの作成
  - 現在の方向性
    - 範囲：JIS X 0208:2012の平仮名・片仮名・漢字・それらに準ずる一部の文字
    - 異体字：定義しない
    - 異体字タイプ：(定義不要)
    - WLE：定義しない
- 漢字を共通に使うCJK(中国語・日本語・韓国語)生成パネル間の調整
  - 漢字(の異体字)の取り扱いをCJKパネル間で合意した上で各言語生成パネルからIPに提案
- 日本語生成パネルの検討状況
  - CJK各生成パネル間で調整するための日本語用RootLGR案を作成
  - CJK各生成パネル間での調整を開始
  - 今後、調整の方向性が見えた時点で日本国内コミュニティに意見募集を実施予定

CJKの各言語用RootLGR



CJK間調整の基本的な考え方



統合後RootLGRの適用例

<日本語の場合>

Language: und-jpan
Applied: 機上
Allocatable: 机上,機上
blocked: 机丄,机仩,機丄,機仩

Language: und-jpan
Applied: 机上
Allocatable: 机上,機上
blocked: 机丄,机仩,機丄,機仩

Language: und-jpan
Applied: 机丄
(申請不可文字を含むため文字列の申請が無効)

Language: und-jpan
Applied: 機机
Allocatable: 机机,机機,機机,機機
blocked: (なし)

<中国語の場合>

Language: und-hani
Applied: 機上
Allocatable: 机上,機上
blocked: 机丄,机仩,機丄,機仩

Language: und-hani
Applied: 机上
Allocatable: 机上,機上
blocked: 机丄,机仩,機丄,機仩

Language: und-hani
Applied: 机丄
Allocatable: 机上,機上
blocked: 机丄,机仩,機丄,機仩

Language: und-hani
Applied: 機机
Allocatable: 机机,機機
blocked: 机機,機机 (S/T mixed)

まとめ

標準化はスタート地点

- 標準化することで使用が開始
  - 標準化により世界中のどこでも同じ方式で母国語が使用できるようになる
  - 使用することで見えてくる課題に対応することで本格的な普及に結びつく
- 下位互換性は重要
  - いったん使用されたものをご破算にすることは困難、下位互換性をできるだけ保ち、非互換な部分に対しては移行ガイドラインを用意する
- 国際化は地域化の準備
  - 多言語が入り混じるということはほとんどないが、他言語の影響は生じ得る
  - 国際化された識別子の使用においては、国際的な協調に基づいた運用が必須

発展途上

- プロトコルの国際化はまだまだ発展途上
  - 必要としている人は多いが、標準を作っている人は少ない
  - UTF-8を使えるようにすることで完成ではない
  - 運用からのフィードバックによる成熟が必要
- まだまだ日本から貢献できる分野
  - IETFでも重要な作業(プロトコル標準化で考慮しなければならない必須項目)として認識されているが、活動は低調である(主要人物が忙しすぎて時間が避けていない)
  - 英語以外の言語知識を持っている人の参加が求められている