「マルチメディア，分散，協調とモバイル
(DICOMO2021) シンポジウム」 令和3年6月-7月

IDNA2008 と PRECIS Framework の相互運用性における考察

根本 貴弘[1]　三島 和宏[1]　萩原 洋一[1]　辻澤 隆彦[1]　青山 茂義[1]

概要：本研究では，インターネット技術における標準化を行う Internet Engineering Task Force にて策定した国際化文字列を含む識別子等を使用するプロトコルに用いられる国際化技術である IDNA2008 と PRECIS Framework における相互運用時の課題を明らかとするために，それぞれの技術において利用可能な文字を定義した Derived Property Value の差分を調査するとともに，その差分が生じた文字に対して，文字列変換処理を利用することで相互利用可能な文字を増やすことが可能か，Unicode13.0.0 に収録された 1,114,112 文字に対して調査を実施した．本研究を通じて，IDNA2008 と PRECIS Framework 間では 14,403 文字において差分が生じていることを明らかとするとともに，文字列変換処理によって相互利用可能な文字がどの程度増えるのか明らかとした．また，文字列変換処理を行っても相互利用できない文字についても明らかとすることで，IDNA2008 と PRECIS Framework における相互運用時の課題となる文字についても明らかとした．

Considerations on the interoperability between IDNA2008 and PRECIS Framework

Takahiro Nemoto[1]　Kazuhiro Mishima[1]　Yoichi Hagiwara[1]　Takahiko Tsujisawa[1]　Shigeyoshi Aoyama[1]

1. はじめに

情報システムの連携範囲が多様化する今日において，従来，閉じたシステムの中で利用されていた情報資源がインターネット上の情報資源として広く活用されることが期待されている．特に近年では IoT 機器の発見に DNS インフラストラクチャを利用する提案等 [1][2] が行われており，異なるプロトコル間で，識別子等の情報資源名が相互利用されることがある．この情報資源名には，ASCII 文字集合の範囲の文字によって作られる名前を利用するシステムもあれば，平仮名や漢字，アクセント記号付きの文字等の ASCII 文字集合の範囲外の国際化文字列の利用も許容するシステムもある．

通常，国際化文字列を使用する場合，その文字の多様な文字を扱う際には，利便性や安全性を考慮し，情報資源参照時の比較一致の機会を増やすための文字列変換処理や意図しない情報資源を参照するリスクを軽減させるための利用可能な文字の制限等の国際化技術が必要となる．

インターネット技術における主要な標準化団体の一つである Internet Engineering Task Force(IETF) では，国際化文字列を扱うプロトコルに対する国際化技術として Internationalized Domain Names for Applications(IDNA2008)[3][4][5][6][7][8] や PRECIS(Preparation, Enforcement, and Comparison of Internationalized Strings in Application Protocols) Framework[9][10] を標準化している．従来，これら国際化技術はそれぞれ利用対象とするプトロコルが分かれており，異なる国際化技術を使用するプロトコル間で識別子を相互利用することは想定されてこなかった．そのため，これら異なる国際技術間における相互運用時の課題についても十分に検討されてこなかった．そこで本研究では，異なる国際化技術を使用するシステム間を行き来する識別子等の情報資源名の利用における相互運用性の向上に寄与することを目指し，相互運用時の課題の範囲を明らかとするために，IETF が策定した代表的な国際化技術である IDNA2008 と PRECIS Framework を対象にそれらが定義する利用可能な文字に焦点を絞り，両国際化技術を相互運用した際に，問題が生じる文字が存在するか調査を行うこととした．

[1] 東京農工大学
Tokyo University of Agriculture and Technology

2. IETF における国際化技術概要

2.1 IDNA2008

IDNA2008 は，ドメイン名に国際化文字列を使用する国際化ドメイン名に関する国際化技術である．国際化文字列には様々な文字が含まれることから，IDNA2008 では，利便性や安全性の観点から文字列の照合の精度を向上させるための文字列変換処理や DNS ラベルとして不適切な文字が含まれていないかを確認するための仕組みを標準化している．IDNA2008 の主な特徴として，国際化識別子の照合を行う際に，Unicode Consortium[11] が定義する Unicode Character Database[12] に基づいた文字列変換及び確認方法を行うことで，現在も改版作業が行われている Unicode の特定のバージョンに依存せず利用可能であるという特徴がある．

文字列変換処理では，照合時の一致機会を向上させるために，以下の処理を定義している．

(1) Case Mapping
UnicodeData の Lowercase mapping の定義に従い大文字を小文字に変換する．

(2) Width Mapping
UnicodeData の Character decomposition mapping の定義に従い文字幅を通常使用される文字幅に変換する．

(3) NFC
Unicode 正規化形式 C を利用し，結合文字列を合成済み文字等の正規等価な文字に変換する．

また, 文字列変換処理では, DNS ラベルの分割を行う区切り文字と同様の役割を持つ「。(U+3002:IDEOGRAPHIC FULL STOP)」を「.(U+002E:FULL STOP)」に変換する特殊な文字列処理も用意されている．

一方，利用可能な文字の定義では，この Unicode Character Database に基づき行われ，IDNA2008 の特性に合わせたカテゴリ (A)-(J) に分類を行い，この分類を利用して Derived Property Value[13] と呼ばれる以下の値を割り当て，利用可能な文字を定義している．

- PVALID - 許可
- CONTEXTJ - コンテキスト制約を持つ文字（結合制御文字）
- CONTEXTO - コンテキスト制約を持つ文字（その他の文字）
- DISALLOWED - 禁止
- UNASSIGNED - 未割り当て

(A)-(J) のカテゴリ分類に基づく Derived Property Value の算出方法については，**図 1** の通りである．

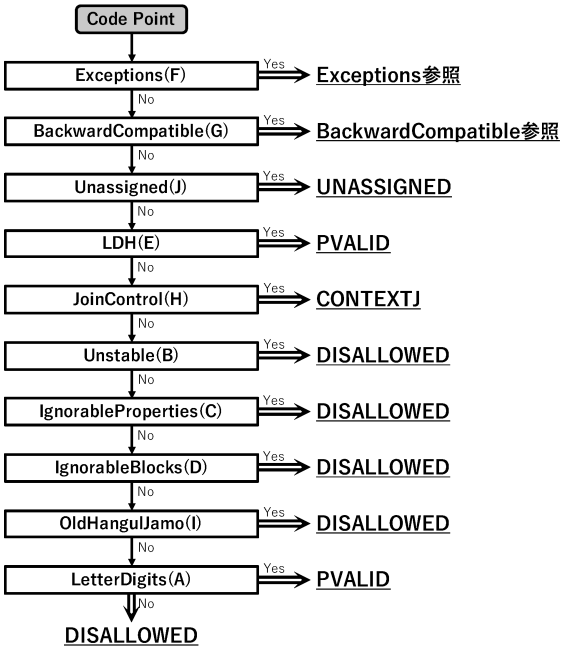

図 1 IDNA2008 の Derived Property Value の算出方法

2.2 PRECIS Framework

PRECIS Framework は，国際化文字列を使用するプロトコロル毎に国際化手法を標準化することや標準化した仕様の維持，管理による負担を軽減するために，各プロトコルで共通して利用可能な国際化手法をフレームワークとして標準化している．

PRECIS Framework を利用するプロトコルには，Stringprep を利用する SASL[14] や XMPP[15]，LDAP[16] 等のプロトコルがあり，それらのプロトコルは PRECIS Framework のプロファイル [17][18] に従い，PRECIS Framework を利用する．

なお，PRECIS Framework 以前は，Stringprep[19] と呼ばれる国際化技術が使用されていたが，Stringprep では，Unicode3.2.0 以外の Unicode のバージョンが利用できないという課題があったことから，PRECIS Framework では，IDNA2008 同様に，Unicode Character Database に基づいた文字列変換及び確認方法を定義している．

文字列変換処理では，プロトコルによって利用すべき文字列変換処理を選択可能としつつ，その代表的な文字列変換処理として以下の変換処理等を定義している．

(1) Width Mapping

(2) Case Mapping

(3) NFC

また，これらの文字列変換処理の他には，プロトコル独自の文字列変換に対応するために，Additioanl Mapping と呼ばれる，視覚的に見分けることが困難な連続した空白文字を一つの空白文字に変換する文字列変換処理等も定義されている．

一方で，利用可能な文字の定義では，IDNA2008 同様に Derived Property Value[20] を割り当て定義している．なお，PRECIS Framework では，Derived Property Value 算出前に用いられるカテゴリは (A)-(R) まで定義している．

PRECIS Framework でとりうる Derived Property Value の値は以下の通りである．PRECIS Framework では，識別子等での利用を想定し定義した IdentifierClass とパスワード等での利用を想定し定義した FreeformClass があり，ID_DIS or FREE_PVAL の値を取る文字は，IdentifierClass では利用が禁止され，FreeformClass では利用の許可され扱いが異なる．

- PVALID - 許可
- ID_DIS or FREE_PVAL - IdentifierClass では禁止，FreeformClass では許可
- CONTEXTJ - コンテキスト制約を持つ文字（結合制御文字）
- CONTEXTO - コンテキスト制約を持つ文字（その他の文字）
- DISALLOWED - 禁止
- UNASSIGNED - 未割り当て

(A)-(R) のカテゴリ分類に基づく Derived Property Value の算出方法については，**図 2** の通りであり，IDNA2008 と Derived Property Value の算出方法が異なる．

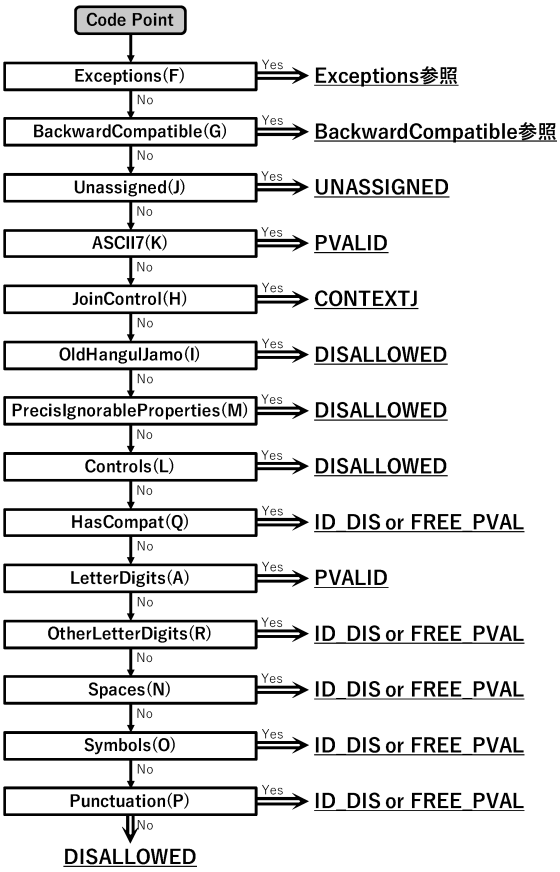

図 2 PRECIS Framework の Derived Property Value の算出方法

3. 調査概要

3.1 調査目的

本研究では，国際化文字列を含む識別子等を使用するプロトコルに用いられる国際化技術である IDNA2008 と PRECIS Framework における相互運用時の課題を明らかとするために，以下の 2 つの項目において調査を実施する．

(1) 両国際化技術が定義する利用可能な文字の差分
(2) 文字列変換処理による利用可能文字の変化

両国際化技術が定義する利用可能な文字の差分では，IDNA2008 と PRECIS Framework においてそれぞれ利用可能な文字を定義した Derived Property Value の差分を調査を行うことで，両国際化技術で異なる Derived Property Value が割り当てられている文字が存在するか調査を行う．これにより，IDNA2008 では利用が許可されているが，PRECIS Framework では利用が禁止されている，またはその逆となる相互運用上問題となる文字が存在するかを明らかとする．

次に文字列変換処理による利用可能文字の変化では，両国際化技術が定義する利用可能な文字の差分調査において差分が生じた場合，IDNA2008 及び PRECIS Framework で定義している文字列変換処理を利用することで，利用不可能な文字をどの程度利用可能な文字に変換できるか調査を行う．これにより，いずれかの国際化技術における Derived Property Value 上では使用を禁止されているが，実際は両国際化技術で使用可能であり相互運用上問題のない文字を明らかとする．また，それとともに，文字列変換処理を利用しても差分が生じる，IDNA2008 もしくは PRECIS Framework のみでしか使用できない文字がどの程度存在するかも明らかとする．

3.2 調査方法

本調査では，IDNA2008 と PRECIS Framework の Derived Property Value を用い調査を行った．Unicode には様々なバージョンが存在することから調査対象とする Unicode のバージョンを揃える必要がある．本調査では，調査時において最新の Unicode のバージョンである Unicode13.0.0 を対象とし，そこに収録された 1,114,112 文字分のコードポイントにおける IDNA2008 と PRECIS Framework の Derived Property Value を比較し，その差分を抽出することとした．また，どのような文字において差分が生じたかを俯瞰するために，UnicodeData の General Category に従い分類した差分の生じた文字数をその内訳として集計する．なお，両国際化技術が定義する利用可能な文字は Derived Property Value として Internet Assigned Numbers Authority（IANA）[21] にて管理されているが，Unicode13.0.0 に基づく Derived Property Value は登録されていないため，本調査においては，IDNA2008 及び PRECIS Framework がそれぞれ定める Derived Property Value の計算アルゴリズムを実装したプログラムを用い，それによって算出された Derived Property Value を利用し調査を行った．また PRECIS Framework では，適用するサブク

ラスが IdentifierClass か FreeformClass によって利用可否が異なる文字が存在する．これに該当する文字については，より広範囲な文字を使用可能としている FreeformClass に従い使用可能文字としてみなし比較を行う．

続いて，両国際化技術が定義する利用可能な文字の差分調査において差分が生じた場合に行う文字列変換処理による差分縮小効果の調査では，IDNA2008 及び PRECIS Framework でプロトコル要素に渡す前の前処理として実施される以下の文字列変換処理を順番に実施し，差分が縮小するか確認をする．

(1) Case Mapping
(2) Width Mapping
(3) NFC

なお，両国際化技術におけるこれらの文字列変換は，文字の視覚的ないし意味論上同等と見なせる文字を同等の文字として扱うことで，情報参照時の利便性を向上させることを目的として用意されているため，本調査においても，これらの変換処理を行うことで，差分が生じた文字のうち利用不可能な文字を利用可能な文字に変換可能かを調査する．

また，本調査ではそれぞれの文字列変換を実施した段階で，どの程度の文字が利用可能な文字に変換可能されたかについても確認するとともに，最終的にこれら文字列変換を実施しても利用可能な文字に変換不可能だった文字も明らかとする．

4. 調査結果

4.1 両国際化技術が定義する利用可能な文字の差分

本節では，両国際化技術が定義する利用可能な文字の差分の調査結果として，IDNA2008 と PRECIS Framework における Derived Property Value に差分が生じた文字数について述べる．

Unicode13.0.0 における IDNA2008 と PRECIS Framework の Derived Property Value を比較した結果, 1,114,112 文字のうち 14,403 文字で異なる Derived Property Value が割り当てられていることが明らかとなった．その内訳として IDNA2008 または PRECIS Framework のみで利用可能な文字数を**表 1** に示す．

表 1 IDNA2008 または PRECIS Framework のみで利用可能な文字数

国際化技術名	文字数
IDNA2008	0
PRECIS Framework	14,403

IDNA2008 で PVALID が割り当てられ, PRECIS Framework で Disallowed が割り当てられた文字はなく，一方で，IDNA2008 で Disallowed が割り当てられ, PRECIS Framework で PVALID が割り当てられた文字は 14,403 文字であった．また，その他の Derived Property Value において差分がないことから，Unicode13.0.0 における IDNA2008 と PRECIS Framework の Derived Property Value の差分全てにおいて，PRECIS Framework で使用が許可されている文字が IDNA2008 では使用が禁止されていることが確認できた．なお，IDNA2008 及び PRECIS Framework でそれぞれ PVALID が割り当てられている文字の総数は，128,637 文字及び 143,040 文字であり，その差分も 14,403 文字であり，表 1 の内容と整合していることも確認できた．

また，差分の内訳として**表 2** に UnicodeData の General Category に従い分類した文字数を示す．なお，本表では，差分が生じなかった General Category の表記は省略している．本調査で差分が生じた 23 個の General Category の略称と意味を以下に示す．

- Ll - Lowercase Letter
- Lm - Modifier Letter
- Lo - Other Letter
- Lt - Titlecase Letter
- Lu - Uppercase Letter
- Mc - Spacing Mark
- Me - Enclosing Mark
- Mn - Nonspacing Mark
- Nd - Decimal Number
- Nl - Letter Number
- No - Other Number
- Pc - Connector Punctuation
- Pd - Dash Punctuation
- Pe - Close Punctuation
- Pf - Final Punctuation
- Pi - Initial Punctuation
- Po - Other Punctuation
- Ps - Open Punctuation
- Sc - Currency Symbol
- Sk - Modifier Symbol
- Sm - Math Symbol
- So - Other Symbol
- Zs - Space Separator

General Category 毎に分類した結果様々なカテゴリにおいて差分が生じていたことがわかった．特に，これらの分類のうち日本語入力環境から入力可能な文字において差分が生じていることが確認できており，例えば，最も差分が生じた General Category は「So」では「㈱ (U+3231:PARENTHESIZED IDEOGRAPH STOCK)」や「㋿ (U+32FF: SQUARE ERA NAME REIWA)」等の記号文字が含まれていた．また，次いで差分が生じた General Category は「Lo」では「ｧ (U+FF67:HALFWIDTH KATAKANA LETTER SMALL A)」の半角カタカナや「勉 (U+FA33:CJK COM-

表 2 各 General Category における差分の生じた文字数

General Category	文字数
Ll	704
Lm	146
Lo	2,157
Lt	31
Lu	1,705
Mc	8
Me	13
Mn	69
Nd	70
Nl	235
No	895
Pc	10
Pd	24
Pe	73
Pf	10
Pi	12
Po	588
Ps	75
Sc	62
Sk	122
Sm	948
So	6,429
Zs	17

PATIBILITY IDEOGRAPH-FA33)」の CJK 互換漢字等が含まれていた．また，「Lu」では「A (U+FF21:FULLWIDTH LATIN CAPITAL LETTER A)」の全角英字，「Po」では「@ (U+FF20:FULLWIDTH COMMERCIAL AT)」の全角記号，「Zs」では「　(U+3000:IDEOGRAPHIC SPACE)」の全角空白文字と日本語入力環境で一般的に見かける全角文字も含まれていた．

4.2 文字列変換処理による利用可能文字の変化

前節にて利用可能な文字に差分が生じた結果を受けて，本節では，文字列変換処理による差分縮小効果として IDNA2008 及び PRECIS Framework で定義している文字列変換処理を順番に実施し，利用不可能な文字をどの程度利用可能な文字に変換可能かについて述べる．なお，前節にて生じた差分は，全て PRECIS Framework では使用可能だが IDNA2008 では使用不可能な文字であったことから，文字列処理を実施することにより，それらの文字が IDNA2008 でも使用可能となるかという観点から調査結果を整理する．

4.2.1 Case Mapping 実施後の Derived Property Value の変化

まず，Derived Property Value に差分の生じた 14,403 文字を対象に Case Mapping を実施した結果を，**表 3** に示す．

Case Mapping の対象となる文字は，差分の生じた 14,403 文字中 1,307 文字であった．また，Case Mapping を行うことで，「A(U+0041:LATIN CAPITAL LETTER A)」や「À(U+00C0:LATIN CAPITAL LETTER A GRAVE)」等の大文字を対応する小文字に変換し，1,194 文字が IDNA2008 でも利用可能な文字に変換可能であることが確認できた．一方で，Case Mapping を行っても利用不可能な 113 文字の中には，「A (U+FF21:FULLWIDTH LATIN CAPITAL LETTER A)」等の全角英字の大文字や「IJ(U+0132:LATIN CAPITAL LETTER I J)」等の対応する小文字も利用が禁止されている文字が含まれていた．

表 3 Case Mapping 後の利用可能文字の変化

項目	文字数
変換対象文字数	1,307
変換後に利用可能となった文字数	1,194
変換後も利用不可能な文字数	113
利用不可能な文字の総数	13,209

4.2.2 Width Mapping 実施後の Derived Property Value の変化

続いて，Case Mapping 後も Derived Property Value に差分の生じた 13,209 文字を対象に Width Mapping を実施した結果を，**表 4** に示す．

表 4 Width Mapping 後の利用可能文字の変化

項目	文字数
変換対象文字数	225
変換後に利用可能となった文字数	121
変換後も利用不可能な文字数	104
利用不可能な文字の総数	13,088

Width Mapping の対象となる文字は，Case Mapping 後も差分の生じた 13,209 文字中 225 文字であった．また，Width Mapping を行うことで，「ｧ (U+FF67:HALFWIDTH KATAKANA LETTER SMALL A)」等の半角カタカナ，「ａ (U+FF41:FULLWIDTH LATIN SMALL LETTER A)」や「１ (U+FF11:　FULLWIDTH DIGIT ONE)」，「－ (U+FF0D:FULLWIDTH HYPHEN-MINUS)」等の全角文字を対応する文字幅の文字に変換し，121 文字が IDNA2008 でも利用可能な文字に変換可能であることが確認できた．また，Case Mapping または Width Mapping のみの実施では利用可能な文字に変換不可能な「A (U+FF21:FULLWIDTH LATIN CAPITAL LETTER A)」等の全角英字の大文字は，これら Mapping を組み合わせることで利用可能な文字へと変換できることが確認できた．一方で，Case Mapping 後に Width Mapping を行っても利用不可能な 104 文字の中には，対応する文字幅の文字の利用が禁止されている「@ (U+FF20:FULLWIDTH COMMERCIAL AT)」等の全角記号が含まれていた．

なお，Width Mapping を行うことで「・

(U+FF65:HALFWIDTH KATAKANA MIDDLE DOT)」のみ CONTEXTO という特殊な Derived Property Value が割り当てられている「・(U+30FB:KATAKANA MIDDLE DOT)」に変換されることが確認できた．CONTEXTO は，コンテキスト制約を持つ文字を分類する特殊な Derived Property Value で，文字列をなすコンテキスト次第で問題を引き起こす特性を持つコードポイントとされており，この値を持つ文字は，他の文字とのコンテキストに従い，使用の可否が決まる．JPRS の汎用 JP ドメイン名登録等に関する技術細則 [22] ではこの文字を国際化ドメイン名における日本語ラベルとしての利用が認められているが，RFC5894[7] では，混乱や問題を引き起こす可能性があることから，ゾーン管理者が十分な理解を持つべきであるとしている．そのため，日本語に対応するための適切な文字列処理が実装されていないシステムでは，この文字は使うべきでないとみなし，表 4 では，利用不可能な文字として計上している．

4.2.3 NFC 実施後の Derived Property Value の変化

続いて，Case Mapping 及び Width Mapping 後も Derived Property Value に差分の生じた 13,088 文字を対象に NFC を実施した結果を，**表 5** に示す．

表 5 NFC 後の利用可能文字の変化

項目	文字数
変換対象文字数	4,813
変換後に利用可能となった文字数	1,094
変換後も利用不可能な文字数	3,719
利用不可能な文字の総数	11,994

NFC の対象となる文字は，Case Mapping 及び Width Mapping 後も差分の生じた 13,088 文字中 4,813 文字であった．また，NFC を行うことで，「勉 (U+FA33:CJK COMPATIBILITY IDEOGRAPH-FA33)」等の CJK 互換漢字や U+0958:DEVANAGARI LETTER QA 等のデーヴァナーガリー文字，U+0F43:TIBETAN LETTER GHA 等のチベット文字等が対応する正規等価な文字に変換され，1,094 文字が IDNA2008 でも利用可能な文字に変換可能であることが確認できた．また，Case Mapping または NFC のみの実施では利用可能な文字に変換不可能な U+1FBB:GREEK CAPITAL LETTER ALPHA WITH OXIA 等のアクセント記号付きのギリシア文字の大文字は，これら文字列変換処理を組み合わせることで利用可能な文字へと変換できることが確認できた．一方で，NFC を行っても利用不可能な 3,719 文字の中には，「ⅰ (U+2170:SMALL ROMAN NUMERAL ONE)」等のローマ数字，「㈱ (U+3231:PARENTHESIZED IDEOGRAPH STOCK)」や「㋿ (U+32FF: SQUARE ERA NAME REIWA)」等の記号等の文字が含まれていた．

4.2.4 General Category 毎の利用可能文字の変化

Case Mapping，Width Mapping 及び NFC による文字列変換処理を実施することで生じた利用可能文字の変化をまとめる．表 5 より，文字列変換処理実施後の最終的な差分数は 12,001 文字であることが明らかとなった．そのため，文字列変換処理実施前の 14,403 文字と比べ，2,402 文字が文字列変換処理を実施することで利用可能となることがわかった．また，どのような文字において変化が生じたかを俯瞰するために，General Category 毎に集計した利用可能文字の変化を**表 6** に示す．

表 6 各 General Category における差分の生じた文字数

General Category	文字列処理前差分数	文字列処理後差分数	利用可能文字に変換された文字数
Ll	704	668	36
Lm	146	142	4
Lo	2,157	1,041	1,116
Lt	31	31	0
Lu	1,705	478	1,227
Mc	8	8	0
Me	13	13	0
Mn	69	54	15
Nd	70	60	10
Nl	235	235	0
No	895	895	0
Pc	10	10	0
Pd	24	23	1
Pe	73	73	0
Pf	10	10	0
Pi	12	12	0
Po	588	588	0
Ps	75	75	0
Sc	62	62	0
Sk	122	122	0
Sm	948	948	0
So	6,429	6,429	0
Zs	17	17	0

文字列変換処理によって利用可能な文字へと変換された文字が分類される General Category は，「Ll」「Lm」「Lo」「Lu」「Mn」「Nd」「Pd」の 7 カテゴリであった．

「Ll」では，「ａ (U+FF41:FULLWIDTH LATIN SMALL LETTER A)」等の全角英字の小文字 26 文字と U+1F71:GREEK SMALL LETTER ALPHA WITH OXIA 等のアクセント記号付きのギリシア文字の小文字 10 文字が利用可能な文字へと変換されていることを確認した．

「Lm」では，「ｰ (U+FF70:HALFWIDTH KATAKANA-HIRAGANA PROLONGED SOUND MARK)」等の半角カタカナで使用される長音記号，濁点と半濁点の 3 文字と，U+0374:GREEK NUMERAL SIGN のギリシア文字のア

クセント記号 1 文字が利用可能な文字へと変換されていることを確認した．

「Lo」では，「ｧ (U+FF67:HALFWIDTH KATAKANA LETTER SMALL A)」の半角カタカナや「勉 (U+FA33:CJK COMPATIBILITY IDEOGRAPH-FA33)」の CJK 互換漢字，U+0958:DEVANAGARI LETTER QA 等のデーヴァナーガリー文字, U+0F43:TIBETAN LETTER GHA 等のチベット文字，その他言語で利用される文字等を含む 1,116 文字が利用可能な文字へと変換されていることを確認した．

「Lu」では，「A (U+FF21:FULLWIDTH LATIN CAPITAL LETTER A)」の全角英字の大文字 26 文字や U+1FBB:GREEK CAPITAL LETTER ALPHA WITH OXIA 等のアクセント記号付きのギリシア文字の大文字 7 文字，その他言語で利用される文字等を含む 1,227 文字が利用可能な文字へと変換されていることを確認した．

「Mn」では，U+0340:COMBINING GRAVE TONE MARK や U+0F73:TIBETAN VOWEL SIGN II 等の結合文字 15 文字が利用可能な文字へと変換されていることを確認した．

「Nd」では,「１ (U+FF11: FULLWIDTH DIGIT ONE)」等の全角数字 10 文字が利用可能な文字へと変換されていることを確認した．

「Pd」では，「－ (U+FF0D:FULLWIDTH HYPHEN-MINUS)」の 1 文字が利用可能な文字へと変換されていることを確認した．

また，表 5 中の利用不可能な文字の総数より，文字列変換処理を実施しても IDNA2008 と PRECIS Framework における Derived Property Value に差分が生じる文字数は，11,994 文字あることが明らかとなった．

5. 考察

5.1 IDNA2008 のみで利用可能な文字が存在しなかった理由

Unicode13.0.0 における IDNA2008 と PRECIS Framework の Derived Property Value を比較した結果，表 1 に示す通り，14,403 文字が IDNA2008 と PRECIS Framework 間における相互運用上課題となる文字であることが示唆された．

また，これらの文字全てにおいて PRECIS Framework でのみ利用可能な文字であり，IDNA2008 のみで利用可能な文字は存在しなかったことから，その理由について考察する．

これは，Derived Property Value を算出する際に使用される IDNA2008 と PRECIS Framework でそれぞれ定義する文字の分類方法が異なるためであると考えられる．両国際化技術ともに，この文字の分類は Unicode Character Database に従い分類を行う．IDNA2008 以降に策定された PRECIS Framework では，IDNA2008 で定義された文字の分類カテゴリ (A)-(J) を参照しつつも，IDNA2008 特有の文字カテゴリである (B)-(E) は使用せず，別途，(K)-(R) までのより汎用的なプロトコルでの利用を想定した文字カテゴリを定義しており，これらの文字カテゴリを用いて Derived Property Value を算出したことにより，IDNA2008 で利用可能な文字は全て PRECIS Framework で利用可能な文字となったと考えられる．

5.2 文字列変換処理実施後も使用不可能となる文字

本調査を通じ，IDNA2008 と PRECIS Framework 間における相互運用が困難な 14,403 文字のうち，2,409 文字については，文字列変換処理を利用することで相互運用可能となることが明らかとなった．

一方で，文字列変換処理後も PRECIS Framework で利用可能だが IDNA2008 では使用不可能となる文字が 11,994 文字存在することが明らかとなり，これらの文字が IDNA2008 と PRECIS Framework 間における相互運用が困難な課題となる文字であることが示唆された．これらの文字の中には，「Zs」カテゴリに分類される「 (U+0020:SPACE)」や「Nl」カテゴリに分類される「Ⅰ (U+2160:ROMAN NUMERAL ONE)」等のように情報システムにおいて利用者が任意に設定する名前として使用される文字が含まれている [23]．そのため，IDNA2008 と識別子を共有する可能性がる文字列にたいしては，設定可能な文字を IDNA2008 で利用可能な文字に限定するような実装が必要となることが考えられる．

5.3 IDNA2008 と PRECIS Framework における相互運用性

本調査結果を基に IDNA2008 と PRECIS Framework における相互運用性における考察を行う．IDNA2008 のみで利用可能な文字が存在しなかったことから，IDNA2008 で利用可能な文字を PRECIS Framework の FreeformClass で利用する場合は，全ての文字が利用可能であることが明らかとなり，相互運用上問題がないことが示唆された．

一方で，字列変換処理後も IDNA2008 では使用不可能となる文字が 11,994 文字が相互運用が困難な課題となる文字であることが示唆されたとともに，PRECIS Framework の FreeformClass で利用可能な文字を IDNA2008 で使用する場合は，文字列変換処理を行うことで，文字列変換処理前は使用できない 14,403 文字のうち 2,409 文字を利用可能とすることが明らかとなり，文字列変換処理が IDNA2008 と PRECIS Framework の相互運用性を向上させる上で有効となることが考察できた．

また，文字列変換処理によって利用可能な文字に変換された 2,409 文字の中には，日本語入力環境において，一般的に入力可能な半角カタカナや半角長音記号，濁点と半濁

点，全角の英字や数字，記号等も含まれていた．そのため，文字列変換処理は，日本語入力環境においても相互運用性を向上させる上で有効であることが明らかとなった．

6. まとめと今後の課題

本研究では，国際化文字列を含む識別子等を使用するプロトコルに用いられる国際化技術である IDNA2008 と PRECIS Framework における相互運用時の課題を明らかとするために，両国際化技術が定義する利用可能な文字の差分及び文字列変換処理による利用可能文字の変化について調査を行った．その結果，PRECIS Framework のみで利用可能な文字が 14,403 文字あることがわかった．また，これらの文字のうち文字列変換処理を利用することで IDNA2008 でも利用可能となった文字は 2,409 文字あることがわかった．これにより，IDNA2008 で利用可能な文字は PRECIS Framework でも利用可能であることが明らかとなり，相互運用上問題がないことが示唆された．一方で，PRECIS Framewor で利用可能な文字を IDNA2008 で使用する場合は，文字列変換処理を行うことで，IDNA2008 でも 2,409 文字の文字を利用可能とすることが明らかとなり，文字列変換処理が IDNA2008 と PRECIS Framework の相互運用性を向上させる上で有効となることが考察できた．

なお本調査では，PRECIS Framework のサブクラスのうち，より広範囲な文字を使用可能としている FreeformClass に従い，IDNA2008 と比較を行った．しかし，IDNA2008 が国際化ドメイン名という識別子を対象とした国際化技術であることから，実際に相互利用される文字列は PRECIS Framework においても識別子である可能性があり，この場合，本調査結果よりも，IDNA2008 と PRECIS Framework における差分はより小さいものとなることが想定される．そのため，今後は PRECIS Framework の IdentifierClass に対しても同様の調査を行うことで，両国際化技術を利用する際の相互運用性の課題をより明確にすることで，異なる国際化技術を使用するシステム間を行き来する識別子の利用における安全性及び利便性の向上に寄与することを目指す．

参考文献

[1] Peter Van der Stok, Michael Koster, ChristianAmsüss. CoRE Resource Directory: DNS-SD mapping. Internet-Draft draft-ietf-core-rd-dns-sd-05, Internet Engineering Task Force, 2019 年 7 月. Work in Progress.

[2] Ted Lemon, Daniel Migault, and Stuart Cheshire. Homenet Naming and Service Discovery Architecture. Internet-Draft draft-ietf-homenet-simple-naming-03, Internet Engineering Task Force, October 2018. Work in Progress.

[3] J. Klensin. Internationalized Domain Names for Applications (IDNA): Definitions and Document Framework. RFC 5890 (Proposed Standard), August 2010.

[4] J. Klensin. Internationalized Domain Names in Applications (IDNA): Protocol. RFC 5891 (Proposed Standard), August 2010.

[5] P. Faltstrom. The Unicode Code Points and Internationalized Domain Names for Applications (IDNA). RFC 5892 (Proposed Standard), August 2010.

[6] H. Alvestrand and C. Karp. Right-to-Left Scripts for Internationalized Domain Names for Applications (IDNA). RFC 5893 (Proposed Standard), August 2010.

[7] J. Klensin. Internationalized Domain Names for Applications (IDNA): Background, Explanation, and Rationale. RFC 5894 (Informational), August 2010.

[8] P. Resnick and P. Hoffman. Mapping Characters for Internationalized Domain Names in Applications (IDNA) 2008. RFC 5895 (Informational), September 2010.

[9] Peter Saint-Andre and Marc Blanchet. PRECIS Framework: Preparation, Enforcement, and Comparison of Internationalized Strings in Application Protocols. RFC 8264, October 2017.

[10] Y. Yoneya and T. Nemoto. Mapping Characters for Classes of the Preparation, Enforcement, and Comparison of Internationalized Strings (PRECIS). RFC 7790 (Informational), February 2016.

[11] The Unicode Consortium. https://home.unicode.org/.

[12] Unicode Character Database. UCD, September 2009.

[13] IDNA Rules and Derived Property Values. https://www.iana.org/assignments/idna-tables-11.0.0/idna-tables-11.0.0.xml.

[14] A. Melnikov and K. Zeilenga. Simple Authentication and Security Layer (SASL). RFC 4422 (Proposed Standard), June 2006.

[15] P. Saint-Andre. Extensible Messaging and Presence Protocol (XMPP): Core. RFC 3920 (Proposed Standard), October 2004. Obsoleted by RFC 6120, updated by RFC 6122.

[16] J. Sermersheim. Lightweight Directory Access Protocol (LDAP): The Protocol. RFC 4511 (Proposed Standard), June 2006.

[17] Peter Saint-Andre and Alexey Melnikov. Preparation, Enforcement, and Comparison of Internationalized Strings Representing Usernames and Passwords. RFC 8265, October 2017.

[18] Peter Saint-Andre. Preparation, Enforcement, and Comparison of Internationalized Strings Representing Nicknames. RFC 8266, October 2017.

[19] P. Hoffman and M. Blanchet. Preparation of internationalized strings (”stringprep”). RFC 3454 (Proposed Standard), December 2002.

[20] PRECIS Derived Property Value. https://www.iana.org/assignments/precis-tables-6.3.0/precis-tables-6.3.0.xhtml.

[21] Internet Assigned Numbers Authority. https://www.iana.org/.

[22] 株式会社日本レジストリサービス. 汎用 JP ドメイン名登録等に関する技術細則. https://jprs.jp/doc/rule/saisoku-1-wideusejp.html.

[23] 根本貴弘, 三島和宏, 萩原洋一, 辻澤隆彦. 東京農工大学の統合管理運用システムにおける登録文字の実態調査と考察. 学術情報処理研究, Vol. 24, No. 1, pp. 85–93, 2020.