Introduction to Label Generation Ruleset (LGR)

An overview

Before starting with the

LGR (The global approach)

let us take a look at

.bharat policy (The Indian approach)

.bharat policy

- Why understanding the .bharat policy is important?
  - It is founding work connecting IDNs and Indian languages
  - It has been demonstrated and appreciated at various National and International forums
  - It has **all the basis components that are required by the "Root LGR" work**, albeit in different forms.

Introduction to Internationalized Domain Names in Indian Languages

An overview

Character classification

Components of the Syllable

–Consonants(C) :

क ख ग घ ङ च छ ज झ ञ ट ठ ड ढ ण त थ द ध न ऩ य य़ र ऱ ल ळ ऴ व श ष स ह

–Vowels (V) :

अ आ इ ई उ ऊ ऋ ऌ ऐ ऍ ऑ ओ औ ऑ

–Matras (M) :

ा ि ी ु ू ृ ॄ े ॅ ै ॅ ॉ ो ौ ॉ

–Vowel modifiers (D) :

ँ ं ः

–Halant (H) :

्

–Nukta (N) :

़

Formalism at a glance ...

Bird's Eye View of the Policy

- Generic binding formalism
- Restriction Rules
- Variant Tables
- Language Tables

Formalism Illustrated…

- **Variables :**

Dash	→	Hyphen -
Digit	→	Indo-Arabic digits [0-9]
C	→	Consonant
V	→	Vowel
M	→	Matra
D	→	Anusvara/Bindi/Tippi/Sunna
B	→	Chandrabindu/Anunasika/Arasunna
X	→	Visarga/Aytham
H	→	Halant/Chandrakala/Virama
A	→	Addak
N	→	Nukta
Y	→	Avagraha/Praslesham
L	→	Chillu
Z	→	Khanda Ta
k	→	Number of possible Consonant Halanta Sequences

Formalism Illustrated…

- Formalism Operators :

|	→	Alternative
[ ]	→	Optional
*	→	Variable Repetition
( )	→	Sequence Group

Formalism Illustrated…

The Formalism:

Consonant-Syllable →

*k(C[N]H) C[N] [H|D|B|X|BD|BX|M[D|B|X|BD|BX]]
| [CH]Z
| L[HC[D|H|M[D]]]
| AC[D|X|M[D|X]]

Vowel-Syllable → V[D|B|X|BD|BX]

Syllable → Consonant-Syllable [Y] | Vowel-Syllable[Y]

IDN-Label → (Syllable | digit)*([dash](Syllable | digit))

Formalism Illustrated..

Consonant-Syllable :

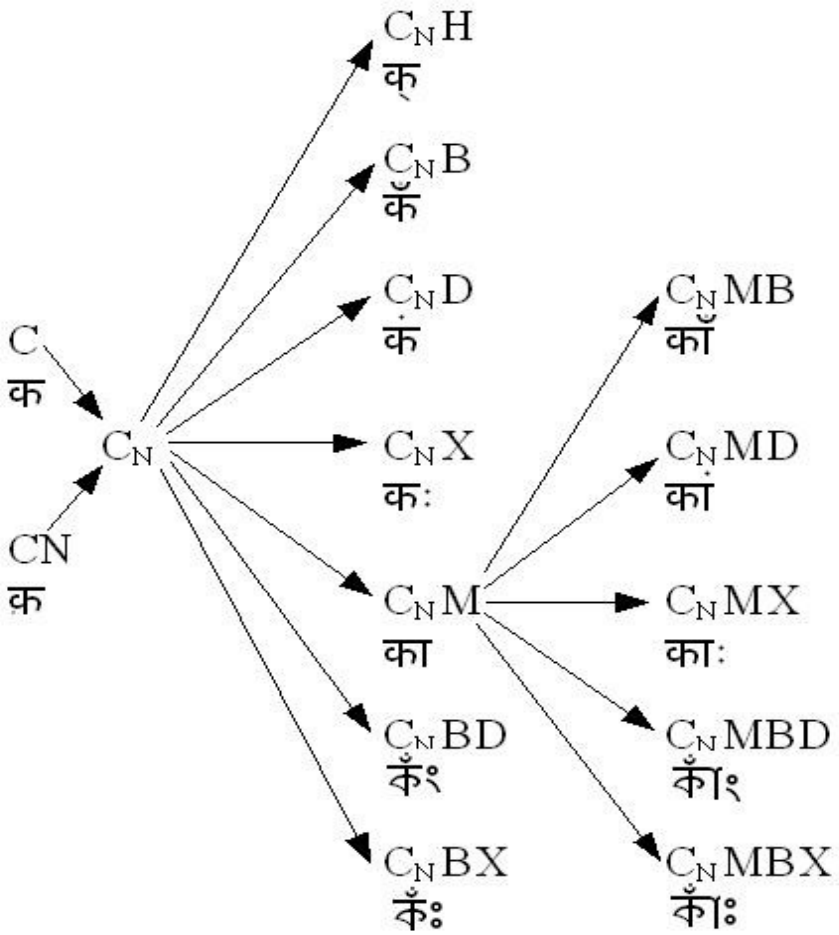
*k(C[N]H) C[N] [
H|D|B|X|BD|BX|M[D|B|X|BD|BX]]
| [CH] Z
| L[HC [D | H | M[D]]]
| AC[D | X | M[D|X]]

ABNF Illustrated..

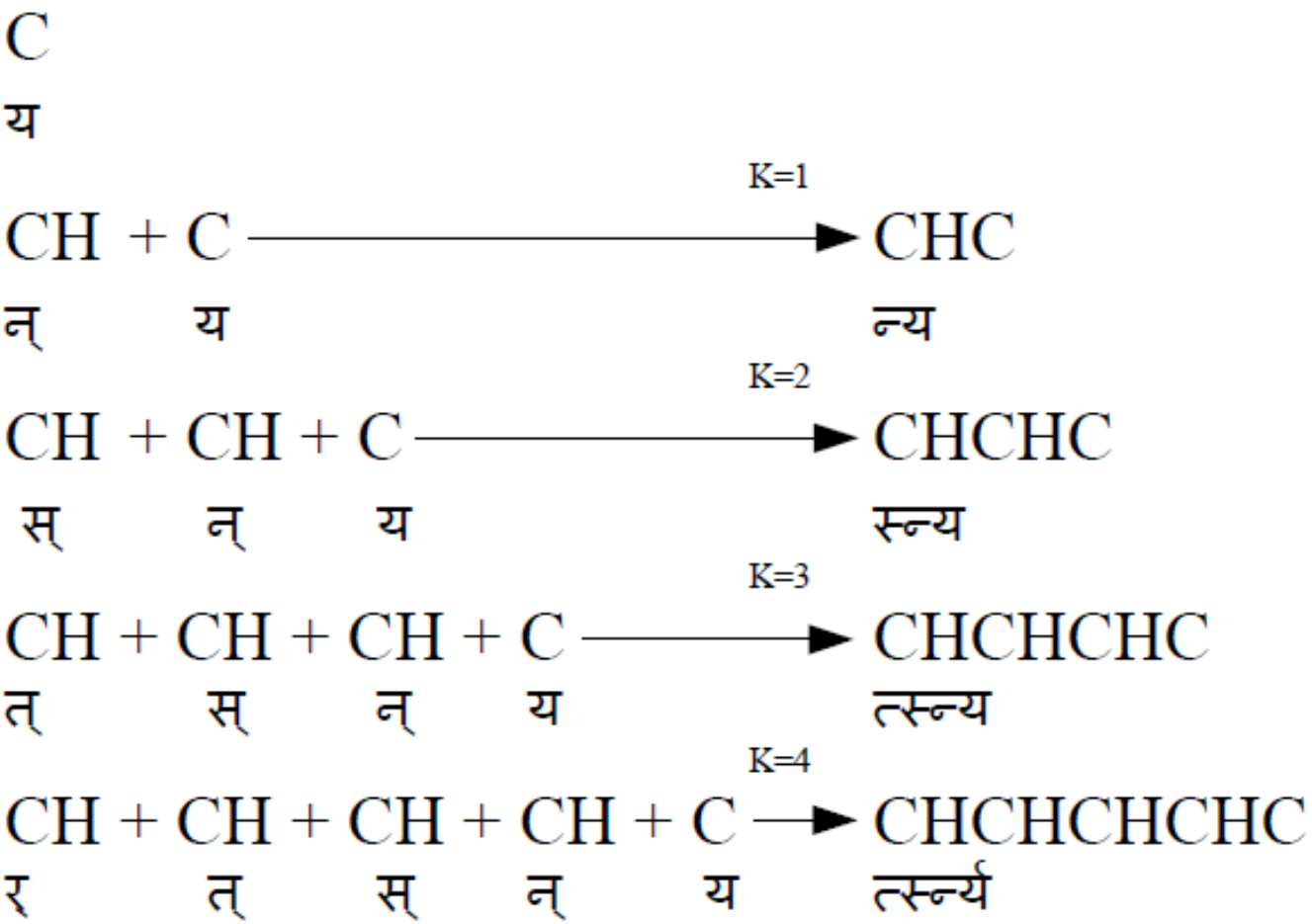
Consonant-Syllable :

*k(C[N]H) C[N] [
H|D|B|X|BD|BX|M[D|B|X|BD|BX]]
| [CH] Z
| L[HC [D | H | M[D]]]
| AC[D | X | M[D|X]]

*k(C[N]H) C[N] [H|D|B|X|BD|BX|M[D|B|X|BD|BX]]



*k(C[N]H) C[N] [H|D|B|X|BD|BX|M[D|B|X|BD|BX]]



ABNF Illustrated...

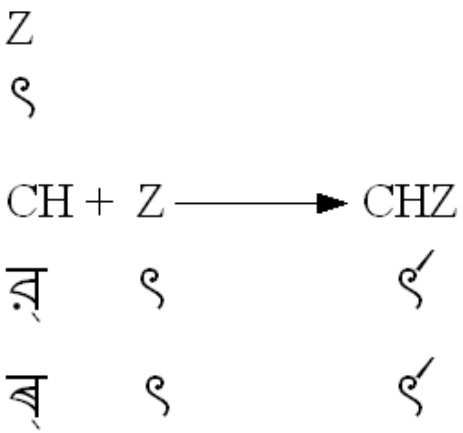
Consonant-Syllable :

*k(C[N]H) C[N] [
H|D|B|X|BD|BX|M[D|B|X|BD|BX]]
| [CH] Z
| L[HC [D | H | M[D]]]
| AC[D | X | M[D|X]]

Consonant Syllable continues…

Syllable with Khanda Ta only exists in Bangla and Assamese language.

[CH] Z



ABNF Illustrated...

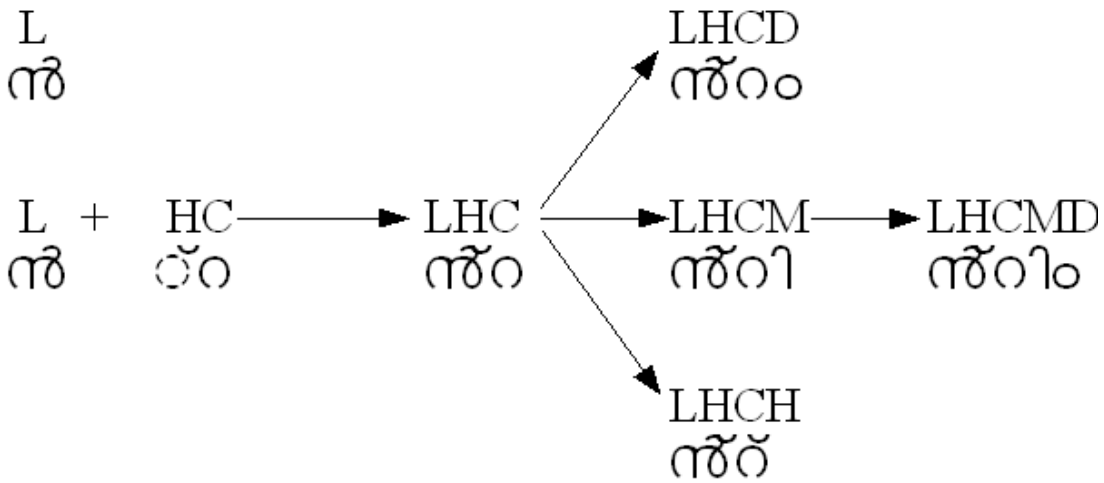
Consonant-Syllable :

*k(C[N]H) C[N] [
H|D|B|X|BD|BX|M[D|B|X|BD|BX]]
| [CH] Z
| L[HC [D | H | M[D]]]
| AC[D | X | M[D|X]]

Consonant Syllable continues…

Syllable with Chillu characters only exists in Malayalam language.

L[HC [D | H | M[D]]]



ABNF Illustrated...

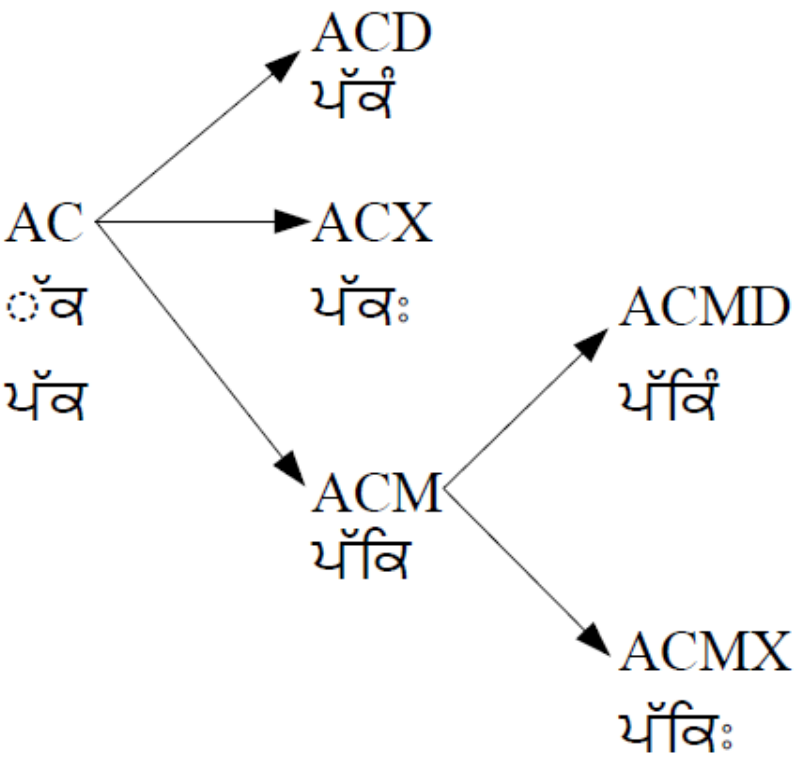
Consonant-Syllable :

*k(C[N]H) C[N] [
H|D|B|X|BD|BX|M[D|B|X|BD|BX]]
| [CH] Z
| L[HC [D | H | M[D]]]
| AC[D | X | M[D|X]]

Consonant Syllable continues...

Syllable with Addak only exists in Punjabi language.

AC[D | X | M[D|X]]



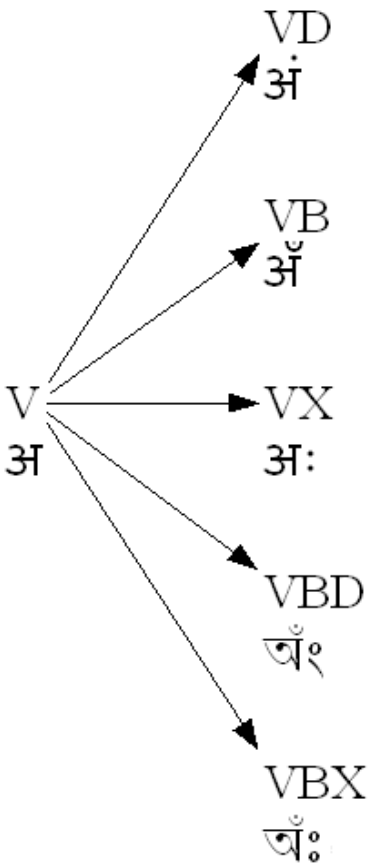
ABNF Illustrated..

Vowel-Syllable :

V [D | B | X | BD | BX]

Vowel Syllable continues..

V [D | B | X | BD | BX]



ABNF Illustrated...

Syllable :

Consonant-Syllable [Y] | Vowel-Syllable[Y]

-where

Consonant-Syllable →
*k(C[N]H) C[N] [H|D|B|X|BD|BX|M[D|B|X|BD|BX]]
| [CH]Z
| L[HC[D|H|M[D]]]
| AC[D|X|M[D|X]]

Vowel-Syllable → V[D|B|X|BD|BX]

Y → Avagraha

IDN Syllable continues…

Consonant-Syllable [Y] | Vowel-Syllable[Y]

Consonant-Syllable + Y ⟶ Consonant-Syllable Y

क s कs

Vowel-Syllable + Y ⟶ Vowel-Syllable Y

अ s अs

ABNF Illustrated...

Complete IDN-Label :

(Syllable | Digit) * ([Dash] (Syllable | Digit))

- where

Syllable → Consonant-Syllable [Y] | Vowel-Syllable[Y]
Dash → Hyphen
Digit → Indo-Arabic digits [0-9]

IDN Label

(Syllable | Digit) * ([Dash] (Syllable | Digit))

IDN Label when starts with

1. Syllable

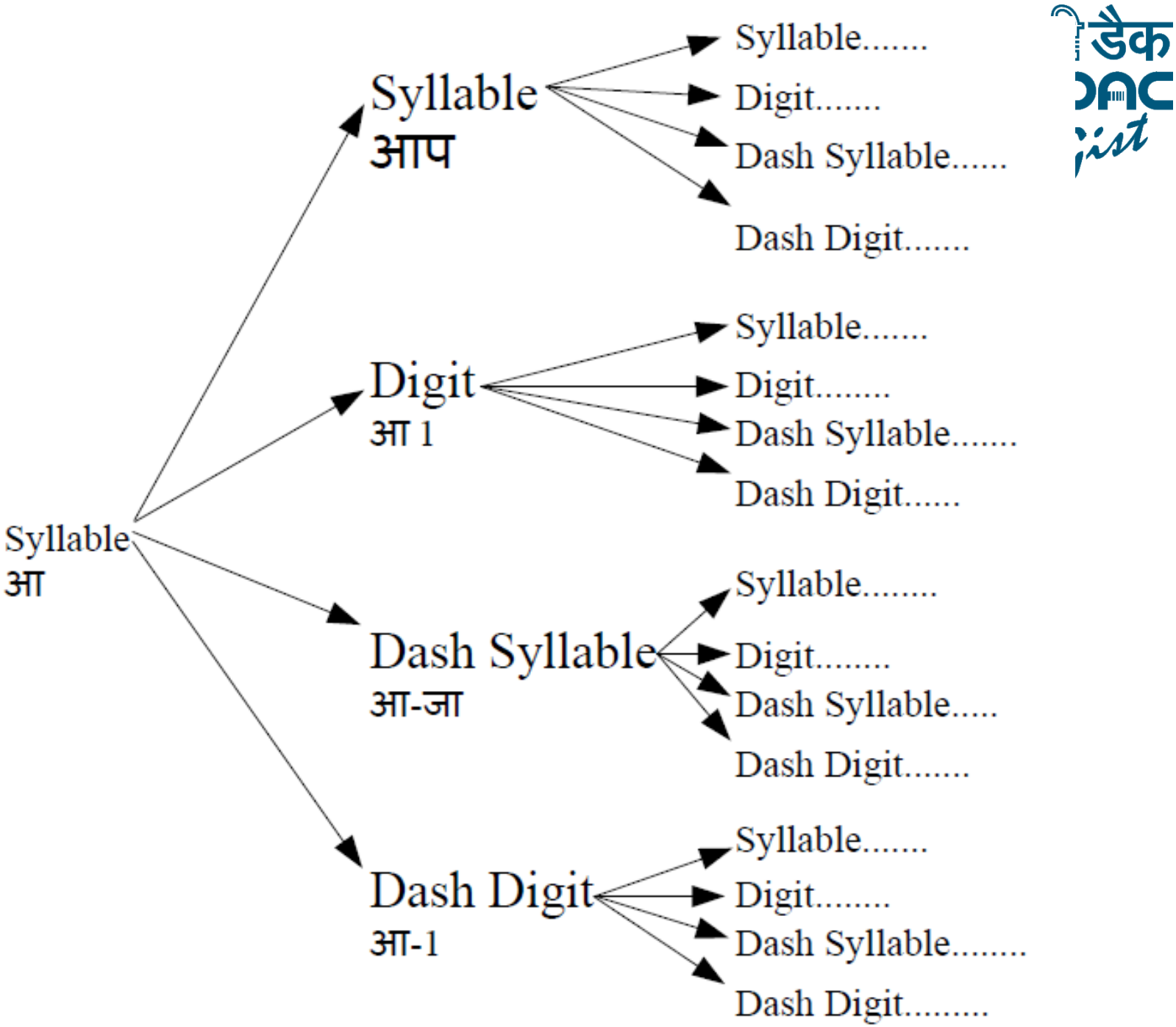
   Syllable * ([Dash] (Syllable | Digit))

2. Digit

   Digit * ([Dash] (Syllable | Digit))

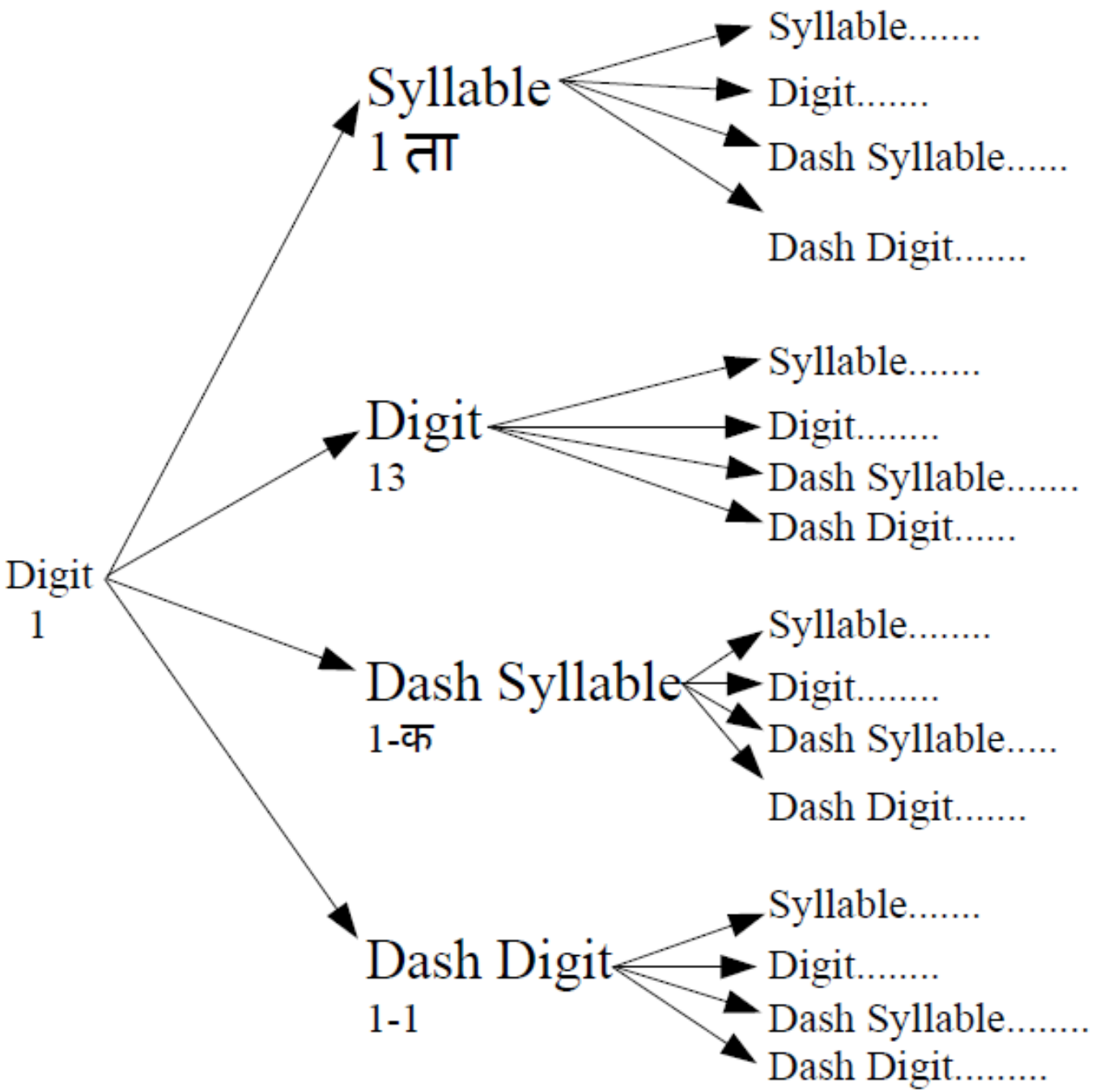
IDN **Label** Continues…

Syllable * ([Dash] (Syllable | Digit))

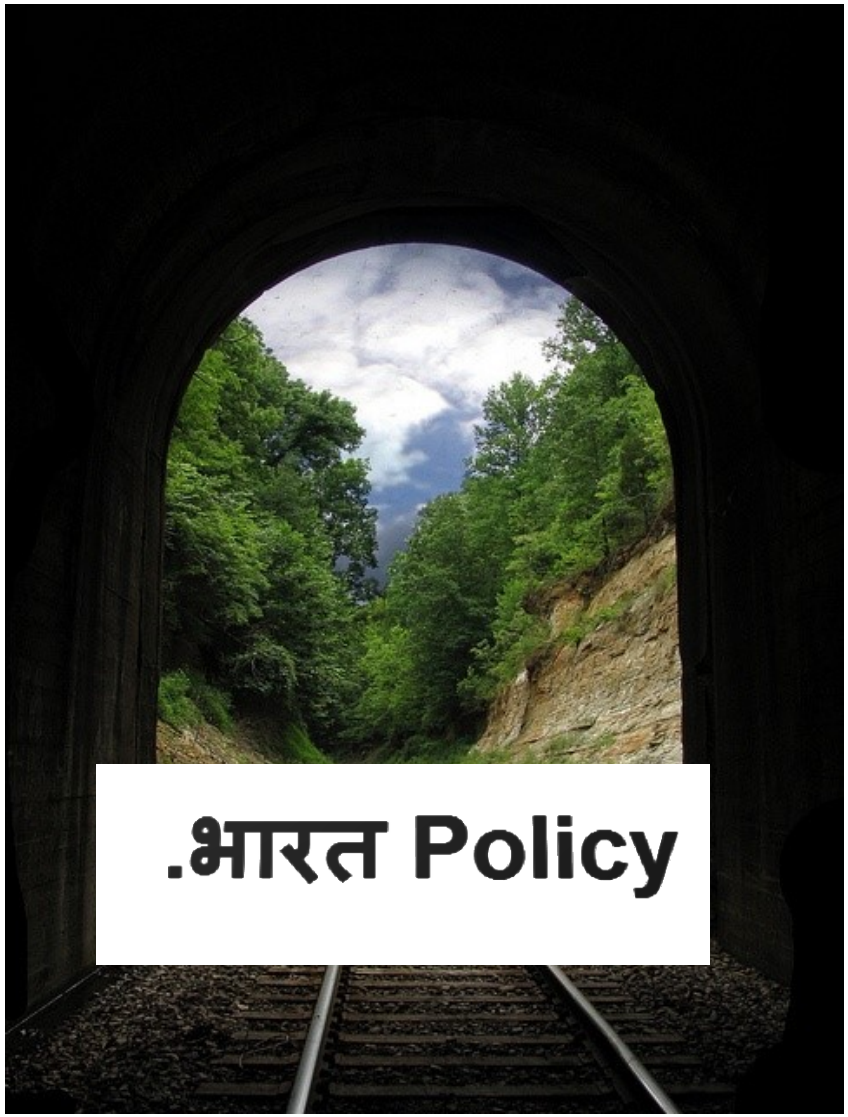

Syllable
आ
Syllable
आप
Digit
आ 1
Dash Syllable
आ-जा
Dash Digit
आ-1
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....

IDN Label Continues…

Digit * ([Dash] (Syllable | Digit))


Digit
1
Syllable
1 ना
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....
Digit
13
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....
Dash Syllable
1-क
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....
Dash Digit
1-1
Syllable.....
Digit.....
Dash Syllable.....
Dash Digit.....

.bharat policy ends...



Introduction to Label Generation Ruleset process for the Root zone

Fundamental differences .bharat and Root LGR

.bharat zone

- It is a focused zone – only the domain names under .bharat TLD
- Restricted only to Indian languages
- Policies can be strict
- Can define our own categories

Root Zone

- It is the most generic zone on the Internet. The root zone.
- Cannot be restricted. Encompasses all the scripts/languages of the world
- Policies have to be simple, yet sufficiently tight
- Have to rely on the Unicode Character properties

Character classes - Differences

.Bharat character classes

C → Consonant
V → Vowel
M → Matra
D → Anusvara/BindiSunna
B → Chandrabindu/Arasunna
X → Visarga/Aytham
H → Halant/Chandrakala
A → Addak
N → Nukta
Y → Avagraha/Praslesham
L → Chillu
Z → Khanda Ta

Unicode character classes

- Mn - Mark, Non-Spacing
  - 0901;DEVANAGARI SIGN CANDRABINDU
  - 093A;DEVANAGARI VOWEL SIGN OE
  - 093C;DEVANAGARI SIGN NUKTA
  - 094D;DEVANAGARI SIGN VIRAMA
- Mc - Mark, Spacing Combining
  - 0903;DEVANAGARI SIGN VISARGA
  - 093E;DEVANAGARI VOWEL SIGN AA
- Lo - Letter, Other
  - 0905;DEVANAGARI LETTER A
  - 0915;DEVANAGARI LETTER KA
  - 093D;DEVANAGARI SIGN AVAGRAHA

Root LGR procedure

- **Fundamental Blocks:**
  - Code point repertoire
  - Variant Rules
  - Whole Label Evaluation rules

0900 Devanagari 097F

	090	091	092	093	094	095	096	097
0	ँ 0900	ऐ 0910	ठ 0920	र 0930	ी 0940	ॐ 0950	ॠ 0960	॰ 0970
1	ं 0901	ऑ 0911	ड 0921	ऱ 0931	ु 0941	ॅ 0951	ॡ 0961	ॱ 0971
2	ः 0902	ओ 0912	ढ 0922	ल 0932	ू 0942	ॆ 0952	ॢ 0962	ॲ 0972
3	ः 0903	ओ 0913	ण 0923	ळ 0933	ृ 0943	े 0953	ॣ 0963	ऄ 0973

ICANN String Similarity Assessment Tool

किताब किताब किताब

Root LGR procedure

- Binding principles:
  - LONGEVITY PRINCIPLE
  - LEAST ASTONISHMENT PRINCIPLE
  - INCLUSION PRINCIPLE
  - SIMPLICITY PRINCIPLE
  - PREDICTABILITY PRINCIPLE
  - STABILITY PRINCIPLE
  - LETTER PRINCIPLE

Root LGR procedure

- **LONGEVITY PRINCIPLE**
  - The panels are supposed to begin using the latest version of Unicode, but also to take into consideration the stability of Unicode character properties.
  - If the panels both fail to behave in this way, then there is a risk either that code points will be permitted for allocation in the root zone that do not work with multiple versions of Unicode, or that code point substitution rules will be adopted that work well in peculiar contexts, but that will work poorly in other (perhaps future) contexts.

Root LGR procedure

- **LEAST ASTONISHMENT PRINCIPLE**
  - The Least Astonishment Principle aims at ensuring that the allocated code points included in the zone repertoire are useful as elements in unique identifiers. To the extent that a code point is confusing to the user population or can be used in surprising ways –whether to members of the original linguistic target community or, in the case of the root, to members of other linguistic communities – use of the code point fails to adhere to the Least Astonishment Principle in that context.
  - The integration panel, especially, is responsible to ensure adherence to the Least Astonishment Principle. Because the Root Zone is a shared resource, the Integration panel is explicitly charged with considering the entire user population, which is everyone on the Internet.

Root LGR procedure

- **INCLUSION PRINCIPLE**
  - The procedure is an example of the Inclusion Principle in action, since every rule or code point is excluded until reviewed and then explicitly included.

Root LGR procedure

- **SIMPLICITY PRINCIPLE**
  - Part of the point of having the integration panel is that it performs a check of the Simplicity Principle. The integration panel cannot possibly include experts in every language and script, but the members must have general knowledge of Unicode, IDNA, DNS, or all of the above. If any member of the integration panel cannot understand the rationale for inclusion of some rule, then that member will not support the rule, and it will not proceed. This is the purpose of the unanimity requirement for the integration panel.

Root LGR procedure

- **PREDICTABILITY PRINCIPLE**
  - The proposal follows the Predictability Principle in much the same way it follows the Simplicity Principle: if the integration panel does not immediately agree with the recommendations of the generation panel, or if members of the integration panel disagree with each other, that is a good reason to suppose that the rule in question is not really predictable.

Root LGR procedure

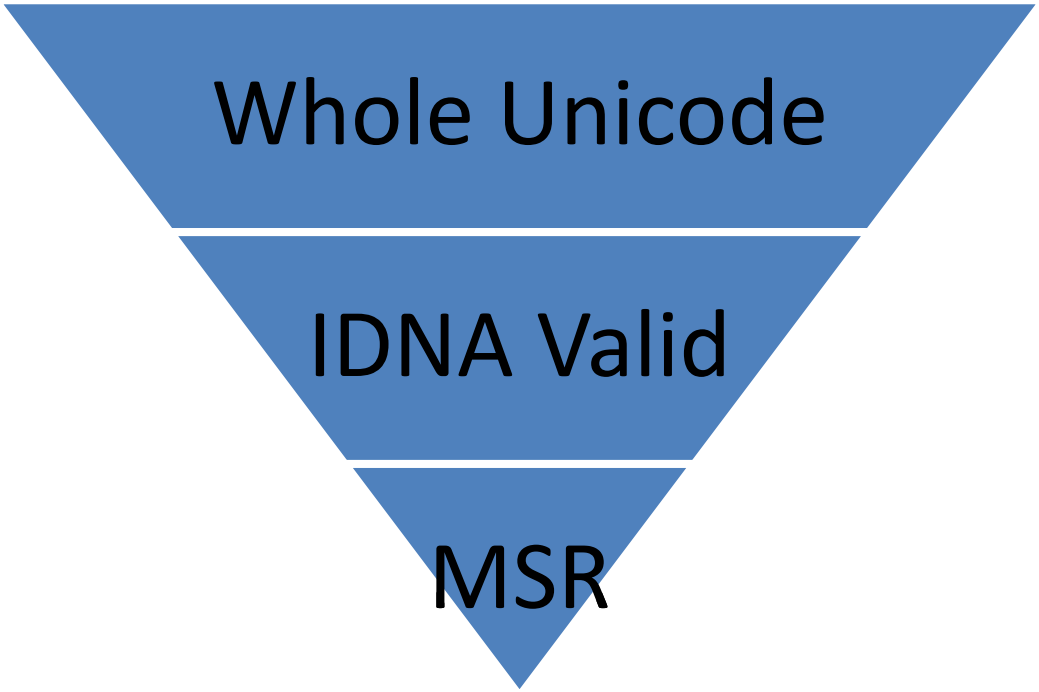
- **STABILITY PRINCIPLE**
  - Especially in the case of the root zone, the Stability Principle is less a matter of guidance and more a statement of fact. The proposed procedure attempts to minimize the possibility that any label generation rule will be permitted for the root zone without that rule having been considered as carefully as possible for any negative consequences. If there is a failure such that the integration panel determines that a previously active rule needs to be removed, this procedure requires that the procedure itself be subject to review.

Root LGR procedure

- **CONSERVATISM PRINCIPLE**
  - The proposal is consistent with the Conservatism Principle in two ways. First and most important, because the integration panel is supposed to reject anything it does not positively think is safe, the Conservatism Principle is built in to the integration panel's criteria. Second, in the event of disagreement between the generation and integration panels, the proposed rule that is the subject of the disagreement is automatically excluded from the root label generation rules.

Root LGR procedure

- Starting point:
  - Maximal Starting Repertoire:



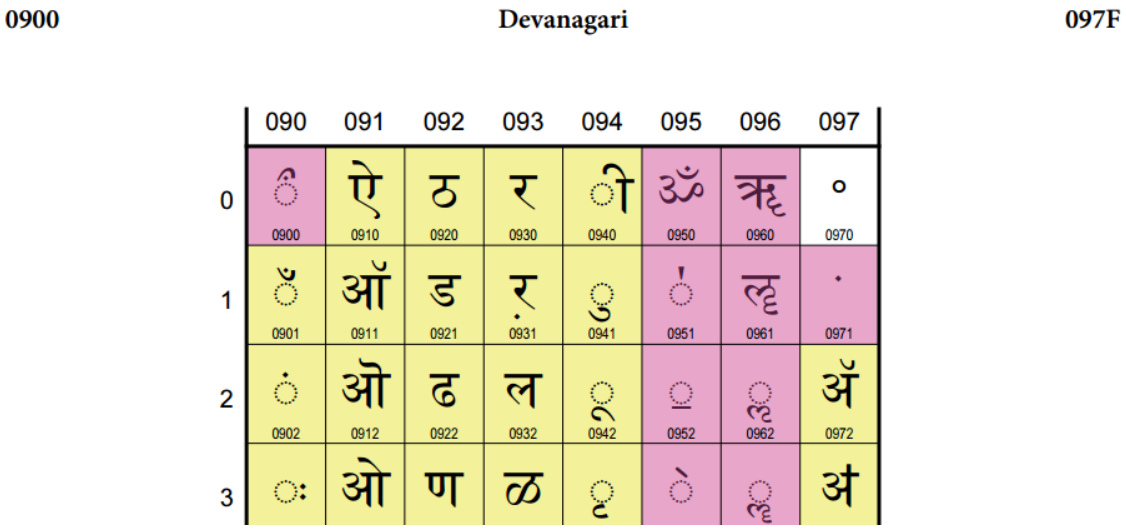
For full language representation

For Domain Names representation

For TLD representation

Root LGR procedure

- Maximal Starting Repertoire:
  - MSR-1 released by the Integration Panel on 20th Jan. ’14
  - MSR-2 released on 15th Dec. 2014



References

- Image: At a glance: http://orientalaviationgr1.ipage.com/

Timelines Discussion

Thanks !