

SOIL (Sanskrit Operating System for Indian Languages)

All major Indian languages share essentially the same alphabet (akshara). Given below are the scripts of alphabets in Odia and Hindi. AS you can see they can be considered as scripts like Arial, Tahoma or Bookmanstyle .

थ	अ	था	आ	क	ख	ग	घ	ङ	ड
च	इ	चि	ई	च	छ	ज	झ	ञ	ज
उ	ऊ	उ	ऊ	ट	ठ	ड	ढ	ण	ण
ऋ	ॠ	ऋ	ॠ	त	थ	द	ध	न	न
ए	ऐ	ए	ऐ	प	फ	ब	भ	म	म
ओ	औ	ओ	औ	य	र	ल	ळ	श	श
०	१	४	५	स	ष	ह	क्ष	य	य

Therefore it is possible to create an efficient coding scheme to support content coding of all Indian languages.

1.1 Why new coding system

Currently, Unicode supports Indian languages by allocating separate code space for various languages. Encoding characters using Unicode basically involves the use of multiple Unicode characters (16-bit wide characters) for encoding a single phoneme. This leads to inefficiencies in terms of both coding-size as well as processing time. In addition, content in one language can't automatically be transliterated into other languages – despite sharing substantially the same alphabet (akshara) scheme. It is therefore desirable to create an efficient standard to encode Indian language content.

1.2 Overview of Proposed Scheme “Sanskriti”

Sanskriti is a product of Super computer Consortium of India (SCCI). A single phoneme can be seen as a product of a vowel and one or more consonants. A 16-bit code can therefore be partitioned into 3 parts (Example 4 + 6 + 6) that can be used to combine a vowel and upto 2 consonants. This scheme can have additional code space of 12-bits for punctuation characters and additional alphabet schemes such as ASCII by designating a particular 4-bit prefix for that purpose.

1.2.1 Benefits of The Proposed New Scheme

1. A single encoding for content in all Indian languages.
2. Simplifies development of hardware/software for display drivers.
3. Simplifies data entry.

4. Nearly 66% reduction in document size compared to unicode encoding
5. Smaller documents lead to more efficient use of communication and storage bandwidth of uncompressed documents
6. Simplifies development of machine translators

1.2.2 Disadvantages of Proposed New Scheme

1. The proposed scheme may require some harmonization of alphabets or using additional 16-bit Sanskriti characters in some corner cases.

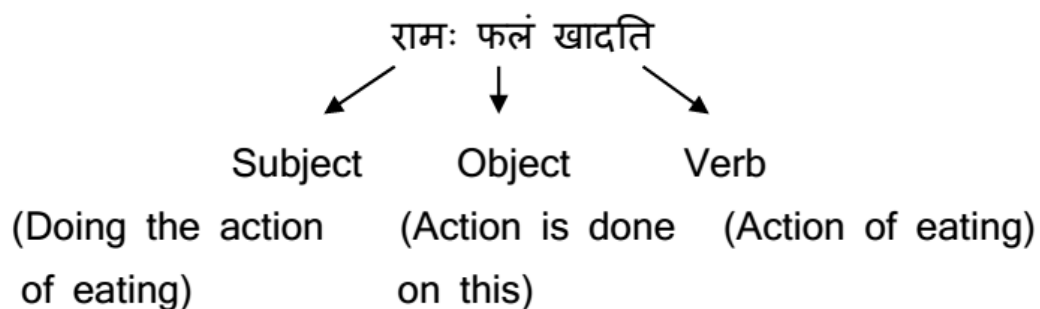
2 Operating System

Create an operating system with native first class support for Indian languages.

- Support a simple programming language for Indian language speakers.
- Create an embedded computer for developing useful applications.
- Grounds up approach to creating a digital universe for India's native language speakers.
- Empower Indians living in villages and towns with low penetration of English to design software powered solutions for local needs.
- Avoid bloat associated with current solutions where Indian language support is an after thought, yet another layer that adds to the cost and complexity of solutions

2.1 The specialities of Sanskrit

There are very systematic rules which can be very well digitalised and can be stored in terms of machine code 0101 in very efficient manner. Let us take an example:

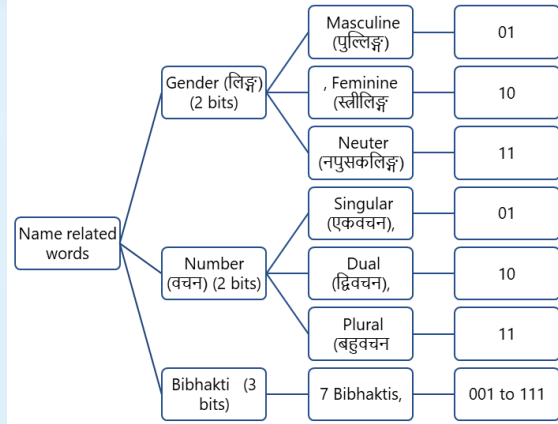
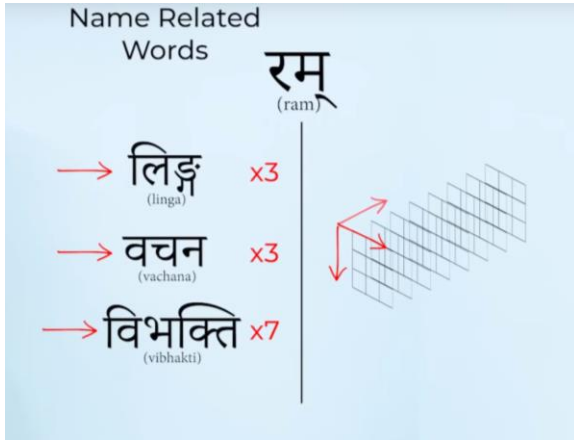


The above simple sentence has one object, one subject and one verb.



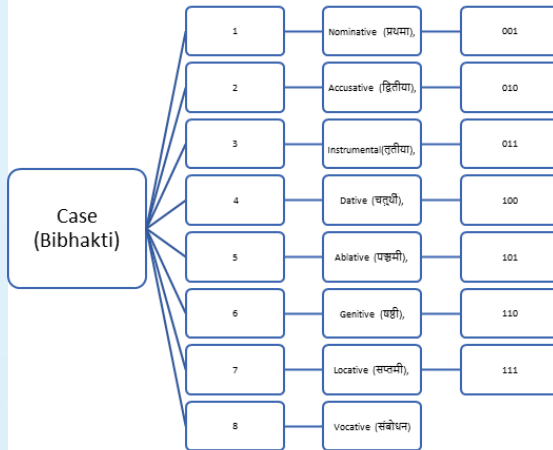
In Sanskrit there are two types of words name related words and verb related words which can be derived a root word called “Dhatu”. Here Dhatu can be developed into many action related words or name related words. It is like the root of the tree. This word can be in the middle expanding on both sides depending on whether it is action or name. If it is a name related word,

it can have 3 attributes, Gender, Number and Bibhakti which can be considered as 3X3 matrix. For 3 types of gender 2 bits, 3 types of Number 2 bits and 7 types of Bibhaktis 3 bits. So prefix can have 7 bits in total to denote all permutation and combination of name related words.



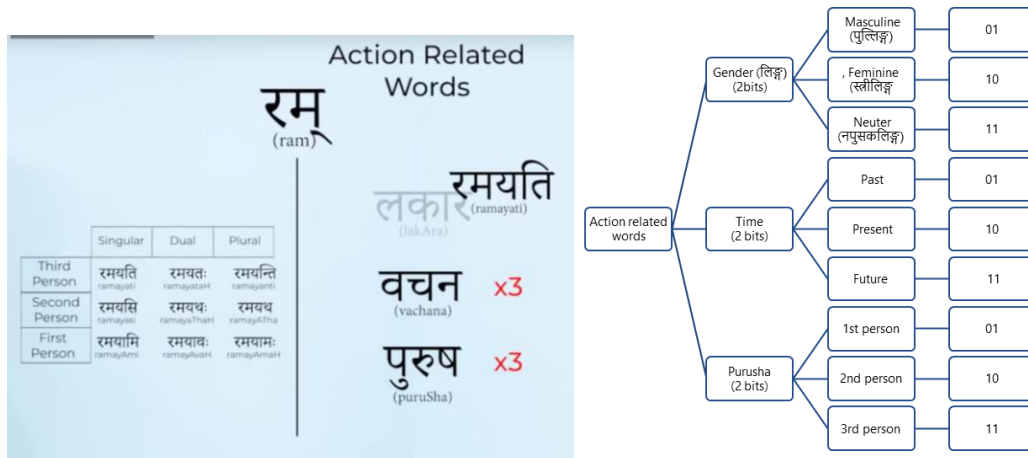
Now consider bibhaktis

	Singular	Dual	Plural
First Division	रामः rAmah	रामौ rAmau	रामाः rAmah
Second Division	रामम् rAmam	रामौ rAmau	रामान् rAmAn
Third Division	रामेण rAmENa	रामाभ्याम् rAmAbhyAm	रामैः rAmah
Fourth Division	रामाय rAmAya	रामाभ्याम् rAmAbhyAm	रामेभ्यः rAmEbhyaH
Fifth Division	रामात् rAmAt	रामाभ्याम् rAmAbhyAm	रामेभ्यः rAmEbhyaH
Sixth Division	रामस्य rAmasya	रामयोः rAmayOH	रामाणाम् rAmANAm
Seventh Division	रामे rAmE	रामयोः rAmayOH	रामेषु rAmEShu
Alt-First Division	राम rAma	रामौ rAmau	रामाः rAmah



So the name related words can have 3X3X7 matrix. Next let us consider action related words. It can be past, present and future (2 bits), Bachan singular, plural and dual (2 bits), Purusha (person) (First person, second

person and 3rd person (2 bits). So action related words can have 6 bits of suffix.



So the action related words can have 3X3X3 matrix.

	Singular	Dual	Plural
Third Person	रमयति ramayati	रमयतः ramayataH	रमयन्ति ramayanti
Second Person	रमयसि ramayasi	रमयथः ramayaThaH	रमयथ ramayATha
First Person	रमयामि ramayAmi	रमयावः ramayAvaH	रमयामः ramayAmaH

The way we have coded the individual alphabets, same way the words can also be coded. With Sanskrit we can deal the words and sentences as some series of mathematics since they are very systematic. The changing the sequence of the words do not change the meaning of the subject. Since all akshars are also unique, the complexity of the entire language can be simplified. Since the words are always derived linked with meaning, the science of learning will open up when we digitalize Sanskrit in the SOIL way. The imagination can be endless.

What I see the light at the end of the tunnel needs a group of Sanskrit scholars, software architects and Chip designers. SCCI is confident that this can also revolutionise the chip designs since complex software code can be represented by a simple hardware. With systematic arrangements of words and sentences the stored data will take very less space. The shlokas and sutras which are in a very symbolic and cryptic form will be decoded using SOIL, since the computer will analyse its meaning keeping all parameters and constraints in to account. So it will encode the natural intelligence as well as Artificial Intelligence.

For further details, contact subratprusty1@gmail.com , gopi@sankhya.com