Title:	Review of existing ISO 639 standards and recommendations for possible improvements
Document type	Working group document
Status	Individual contribution
Source	Peter Constable
Date	2014-06-19
Expected action	For consideration by TC37/SC2/JWG7, TC46/SC4/JWG7

Overview

At the May 2014 meeting in Washington, D.C., JWG7 reviewed document N007, in which an initial assessment of possible business plans for JWG were presented. After discussion, this author took an action item to conduct a more detailed review of the existing ISO 639 standards and to make recommendations for how coherence among the standards could be improved. This document reports findings and recommendations from that review.

Assumptions regarding ISO 639 language codes

There is broad consensus among stakeholders that the ISO 639 standards deal with "language coding", but there have not been clear consensus statements as to what assumptions and implications are for a coding standard created by ISO for broad use across multiple industries, business sectors and other fields of interest. This section delineates assumptions regarding ISO 639 that lie behind the assessment and recommendations that follow. This perspective is informed by other industry coding standards such as ISO/IEC 10646.

1. ISO standards exist to foster international business, commerce and other activities among different parties with shared goals. As such, ISO 639 standards exist to foster interactions between parties where there is a need to represent language that pertain to information objects in a publicly-interoperable manner. They are not created purely as a research or documentation endeavour. They may be informed by scientific research and serve needs in the scientific taxonomy and documentation of societies and the language variants that they use, but scientific itself documentation is not a primary purpose.

2. As with other encoding standards, ISO 639 provides an encoded representation of concepts. Just as ISO 15924 provides a symbol such as "Cyrl" to represent the concept 'Cyrillic script', and just as ISO/IEC 10646 provides a numeric encoding such as U+0041 to represent the concept LATIN CAPITAL LETTER A, in the same way, ISO 639 provides encoded representations of concepts of languages, such as "fr" to represent the concept 'French language', or 'haw' to represent the concept 'Hawaiian language'.

There has at times been confusion on this matter as a result of the title used for the standards, "code for the representation of *names of* languages. That title reflects that the mechanism used for the encoded representation of concepts were string identifiers each based on the name of the language being represented. That mechanism notwithstanding, the items that are given encoded representations are the concepts of particular languages, not particular names of languages. For example, "Lue" and "Xishuangbanna Dai" are two names for a single language, and the different names might be preferred

in different contexts or at different times; but there is only one symbol (khb) as there is only one encoded concept, the language alternately known as “Lue” or as “Xishuangbanna Dai”.

3. ISO 639 identifiers may be used for purposes related to many different types of activities, including commerce, linguistic research, streaming of audio-video content to a personal device, and many other kinds of activity. In every instance, however, ISO 639 identifiers are being used in some information-processing system — software, databases, etc. — created to facilitate those various activities. The ISO 639 standards should be understood to be *information technology* standards.

4. ISO 639 provides symbols to be used as metadata elements for declaring particular properties of information objects. Specifically, ISO 639 provides symbols (identifiers) each of which represents in a concise manner a language that can be attributed as a metadata property of information objects to indicate the language associated with those information objects. The symbols are not an end in themselves, nor are they themselves information objects of primary interest in any usage scenario, apart from systems designed specifically to generate or interpret these symbols.

The way in with the denoted language is associated with the information object may vary. An identifier may be applied to content to indicate that the content is expressed in a particular language. An identifier may be applied to a software resource to indicate that they software provides processing applicable to a particular language. Or an identifier may be applied to a person to indicate they have a preference to use a particular language or have competence in use of the language.

5. As an international standard providing an encoded representation of certain concepts for use as metadata elements, a primary goal must be interoperable interchange of information using those metadata elements. Implications include the following:

- There must be no ambiguity as to what symbols are defined and available for use.
- There must be clarity to all parties as to what each symbol represents, and any conforming implementation that interprets a given symbol cannot assume a different meaning.
- Symbols and their denotations should remain stable over time so as to not break interoperability of existing implementations. Symbols that have been assigned and defined must remain assigned and should not have their denotation changed in ways that would break existing use.

An encoding scheme to be adequate as an international encoding standard should ensure these requirements.

Summary: These assumptions can be summarized in the following representative example: ISO 639 serves and is adequate for its purpose by

- allowing an item of language content such as “aloha” to be attributed within a database using the symbol “haw”,
- allowing any information system created by anyone, anywhere and at any time to be able to recognize that “haw” is a valid language identifier and that the item of content is being declared to be an expression in the Hawaiian language, and
- allowing the information system to correlate that item of content appropriately with particular users, processes or other content on the basis of that symbol.

In contrast, being able to document "Hawaiian is a language" or that "'haw' is a symbol for the Hawaiian language" are not primary purposes for which the standards are created.

Recommendation: Unity of concepts

While ISO 639 parts 1, 2, 3 and 5 are different standards, they have important interrelationships. There is text in some of the parts describing the interrelationships, but this is not stated in all parts and so may not always be clear. In order to ensure interoperable interchange in implementations, it is essential to be clear that there is a single, unified set of language concepts that is encoded in these various parts of 639. For example, the concept represented by the symbol "en" in ISO 639-1 is the same concept as that represented by the symbol "eng" in ISO 639-2 or in ISO 639-3.

Some follow-on recommendations:

- The language concepts are documented in the standards by means of language names — a single reference name in Part 3; English and French names in Part 1 and Part 2. The code tables for the different parts should be consistent with one another in regard to the names used to document the concepts denoted by symbols.
- Part 3 defines properties for language concepts, "scope" and "type", which aid in providing clarity as to the concept being denoted. Such properties should be applied to all items in all parts of 639.
- As described in Constable and Simons (2000), names alone are not truly adequate to document the language concepts denoted by the symbols in 639.[1] The RA for Part 3 has provided links to additional information that documents each concept in greater detail. Ideally, the RAs for all parts would work together to provide such additional documentation for all concepts from every official site on which code tables are documented.

Recommendation: sub-set relationships between parts 1, 2, 3

ISO 639-3 is designed to provide comprehensive coverage of all known languages. ISO 639-2 is designed to cover particular languages that are typically in scope for particular information systems, particularly in the MARC system used by libraries. ISO 639-1 is the legacy coding system intended to cover languages of particular interest for terminology and localization. In terms of language usage, the languages encompassed by Part 1 are inherently going to be relevant for the primary audience of Part 2.

Thus, there designed purposes for parts 1, 2 and 3 suggest a natural subset relationship between these parts: any language included in Part 1 would be appropriate for inclusion in Part 2, and by definition any individual language (but not collections) in Part 2 should be included in Part 2.

These subset relationships should ideally be formalized in the ISO 639 standards and reflected in the procedures used by RAs and the JAC is evaluating requests for addition: for a language to be included in any part of 639, it should first be determined to be in scope for Part 3. Once approved for inclusion in Part 3, it could be tested against additional criteria to determine whether it is in scope for Part 2.

[1] Constable, Peter, and Gary Simons. 2000. "Language identification and IT: addressing problems of linguistic diversity on a global scale." SIL Electronic Working Papers. Also presented at the 17th International Unicode Conference. Available online at http://www.sil.org/resources/archives/7861.

There is one case of an item being included in Part 1 but not in Part 2: Serbo-Croatian “sh”. (This is encoded in Part 3 as “hbs”.) This anomaly to the principle of sub-set relationships should be reviewed and action take to improve coherence among the various parts of 639.

Since the different parts deal with a shared set of concepts yet may appear to users as “different codes”, it’s important the inter-relationships are clear and the reasons why a concept might be encompassed in one part but not another. The relevant criteria for determining inclusion in each part should be clear. This is an area that may need review for possible improvement.

Recommendation: Stability of items

The identifiers provided in 639 get used in information system implementations and data (documents, databases, etc.) that get widely distributed. Once an identifier is assigned, removing it from the standard or re-assigning it to an incompatible concept would create major problems of non-interoperability for existing implementations.

It is critical, then, that identifiers, once added, never be removed from the standard or changed. This point is made in some parts of 639 but is not consistently reflected in all parts. For example, A.3.3 of Part 2 makes reference to deletions and code changes. All parts of 639 should reflect the same policy.

While identifiers cannot be removed without having costly, breaking impact on implementations, they can be deprecated. Deprecations can have cost implications for some implementations, but they leave existing behaviours and relationships intact.

There may be various reasons why a deprecation might be needed. For instance, it may be discovered that two items separately encoded are, in fact, duplicates. Or, language communities may split and diverge over time resulting in a single language splitting to become multiple distinct languages (as may happen, for instance, in the case of Serbo-Croatian). Whenever possible, it would be beneficial for implementations if the reason for deprecation were provided along with recommendations on what actions might be appropriate in implementations.

The terms “delete”, “withdraw” and “retire” have sometimes been used in the standards or by the RAs. These are ambiguous terms that can easily be interpreted by users to suggest that identifiers can be removed. These terms should be avoided. Instead, “deprecate” / “deprecation” should be used whenever applicable.

Reports presenting the code tables should be designed so as to avoid possible confusion regarding deprecations. For instance, the current practice on the 639-3 site is to present reports that exclude deprecated items. Similarly, the machine-readable “ISO 639-3 Code Set” data file does not include any deprecated items. These have led to confusion for users in some cases. These practices should be reviewed to consider how best to avoid confusion and misperceptions regarding deprecated items.

In some cases, there may be reasons to change the names given in the code tables to document the denotation of a given identifier. For example, additional, alternate names might be recorded, or a pejorative name might be removed. Such changes are acceptable provided that the denotation of the identifier has not changed, and provided that there is clear continuity from one revision of the code table to the next. (In other words, when looking at two revisions of the code table, it should be

reasonably obvious to users that the denotation for an identifier has remained the same even if some change to the name has been made.)

In some instances, it may be discovered that an item that had been thought to represent a single, individual language is actually best understood to be two or more distinct languages. The text of ISO 639-3 discusses such cases (clause 4.5.2) and requires that the denotation of an identifier must never be narrowed, but that instead new identifiers for the distinct languages should be created. The rationale for this was that narrowing the denotation on an identifier may have a breaking impact on some existing implementations. In the years since 639-3 was published, however, it has been seen that there may be cases in which narrowing of the denotation might actually have less harmful impact on implementations than having new identifiers that supersede the existing one. Specifically, if the identifier denotes a language with strong support institutionally and in information systems but it also has been deemed to encompass a variety that is rarely used in information systems and that is now understood to be a distinct language, then there may be less risk and impact for existing implementations to deem that the identifier no longer encompasses that lesser-used variety than to propose that some new identifier supersede the existing identifier that is widely implemented. Best practice for such situations should be re-evaluated.

Principles for stability of denotations and documentation of denotations, as discussed here, are not reflected across all parts of 639, but should be.

Recommendation: Single encoded representations

Whenever alternate encoded representations are provided for a given concept, this creates opportunities for failure in interoperation. As a result, alternate encoded representations of concepts within information systems is never a good thing. It should be avoided whenever possible. If it cannot be avoided, then mechanisms to mitigate against potential failures should be provided.

There are certain cases of duplicate representation that have gotten designed into the 639 language coding system for legacy reasons:

- The legacy alpha-2 identifiers in Part 1 have equivalent alpha-3 identifiers in Part 2 / Part 3.
- Within Part 2, twenty-two languages were given dual “B” and “T” identifiers. (All of these have legacy alpha-2 identifiers.)

In addition, there have been some incidental cases of duplication. For example, “Moldovan” was given separate identifiers in Parts 1 and 2 from “Romanian”, yet these are the same language.

In order to avoid creating new cases of dual representation, the alpha-2 code space should be deemed as legacy and frozen. This is analogous in the character encoding space to the ISO 8859 series, which provided single-octet encoded representations of characters. These have now been superseded by multi-octet representations for the Universal Character Set defined in ISO 10646, and the ISO 8859 standards are now stabilized and frozen. Some have suggested that we shouldn’t actually freeze the alpha-2 code space but should simply set some very high bar for additions. That high bar would amount to deeming that there was some highly-critical need to support an additional language in some information system that could accommodate only alpha-2 identifiers — so high as to warrant placing additional costs and burdens on millions of other implementations to deal with the non-interoperability problem of a new dual representation. But there is not any one user of ISO 639 whose needs so

outweigh those of every other user of the standard to warrant placing such a burden on all the others for the benefit of the one.

For existing cases of dual representation, machine-readable data files that provide mappings could be provided. In additional, named normal forms / foldings could be defined that would allow processes to identify or negotiate which encoded representations are used in interchange.

- An “A2” normal form might be defined mapping a particular subset of alpha-3 identifiers to their alpha-2 equivalent.
- A “T” normal form might be defined every alpha-2 or alpha-3 identifier to its equivalent “T” alphpa-3 representation.
- A “B” normal form might be defined mapping every alpha-2 or alpha-3 identifier to its equivalent alpha-3 “B” representation.
- A “BCP47” normal form might be defined mapping every alpha-2 or alpha-3 identifier to the equivalent identifier sanctioned by BCP 47.
- A “remove duplicates” folding could map deprecated identifiers that are duplicates to the recommended equivalent representation.

Parts 1, 2 and 3 allow for reservation of identifiers. This idea may have originated in ISO 3166, which makes use of the notion of reserved identifiers in certain cases. Specifically, when a symbol that is not the sanctioned identifier for some concept has nonetheless been in widespread use in certain contexts, then assigning that symbol to some different concept could create problems for existing implementations. This has been done in ISO 3166-1, for example, in the case of “UK”: there were existing systems that had used this to represent the United Kingdom of Great Britain and Northern Ireland, the 3166 identifier for which is “GB”. The only cases of this nature in relation to 639 are certain historic and erroneous alpha-2 symbols, discussed in Annex B of 639-1:2002 (“iw”, “in”, “ji” and “jw”). There are no additional cases warranting reservation of identifiers, and any requests to reserve identifiers for some purpose (as in some resent requests made to the JAC) would lead effectively to alternate, de facto encoded representations. Thus, the mechanism of allowing new reservations is anachronistic and no longer required, as well has creating opportunity for harmful changes, and hence should be removed.

Recommendation: Clear application scenarios and business needs

Everything in ISO 639 should exist to serve clearly-identifiable application scenarios and business needs. This is certainly the case for parts 1, 2 and 3. It is not clear that the same is true, however, for parts 5 and 6.

The motivation for Part 5 was that Part 2 included both individual languages and collections, that Part 3 expanded the set of individual languages, and so by symmetry one could expand the set of collections. The problem, however, is that there is limited utility for language collection identifiers in information systems, with the only clear application scenario being that of the librarians, whose needs were already accommodated in Part 2. Hence, there is no clear application scenario for the additional collection identifiers in Part 5. Moreover, there is no obvious set of criteria to be applied to evaluating further possible additions (a problem that the Registration Authority is facing without guidance from the standard).

The motivation for Part 6 was that BSI had presented Linguasphere as a possible candidate for expanded encoding beyond Part 2, there was understanding that some kind of support for sub-language variants was needed in information systems, and Linguasphere offered something of that nature. The problem, however, is that Linguasphere provided one particular view of language variations, whereas in principle there may be multiple ways to characterize variations depending on the usage scenario. (This point was made in presentations by Sebastian Drude and by Sue Ellen Wright in the TKE2014 conference.[2] Moreover, the very audience with the greatest unmet need for handling language variations — linguists involved in language documentation — have rejected Part 6 as not meeting their needs.

For these reasons, both Part 5 and Part 6 are problematic. A review of whether these should be maintained or perhaps withdrawn appears to be warranted.

In regard to Part 5, if it is to be kept, then some clarification on usage scenarios and decision criteria to determine inclusion of potential additions would be essential.

Going forward, greater care is needed when evaluating possible additions to the 639 family of standards to ensure there are clear application scenarios and clear principles for operation and maintenance in order to ensure that the product is actually useful and maintainable.

Summary

The following summarizes recommendations:

- Different parts should reflect that there is a single set of concepts. The same appellations for a given concept should always be used across all parts.
- Properties introduced in Part 3 should be applied to all parts.
- Links to external references that provide detailed description of concepts should be provided for all parts.
- Consistent subset relationships between parts 1, 2 and 3 should be enforced, and criteria for inclusion in each part should reflect this.
- All parts should reflect that identifiers cannot be removed or changed. (Parts 1 and 2, in particular, do not reflect this.)
- RAs and published standards should be consistent in using “deprecation”, not “withdraw”, “retire”, etc. Code table reports and data files should clearly reflect the complete encoding, including deprecated items.
- Best practice in regard to “splits” in ISO 639-3 should be reconsidered.
- Part 1 should be frozen in order to avoid new duplicate encodings.
- Named normal forms or symbol foldings with associated data files should be considered.
- Mechanism for reservation of symbols should be removed.
- The business need for Part 5 should be revisited. If there is a clear business need, then the usage scenarios and decision criteria for inclusion should be made clear.

[2] Wright, Sue Ellen. 2014. “Language codes and language tags.” Presentation at the TKE 2014 “Language Codes at the Crossroads” workshop, June 19 – 21, 2014, Berlin.
Drude, Sebastian. 2014. “Current research in the field of language documentation.” Presentation at the TKE 2014 “Language Codes at the Crossroads” workshop, June 19 – 21, 2014, Berlin.

- The business need for Part 6 should be revisited; withdrawal of Part 6 is recommended.