Draft CEN Workshop Agreement on

'Functional Multilingual Extensions to European Keyboard Layouts'

Status: CWA Final Draft for public comments, 2 October 2009

Contents

Foreword

The production of this document which addresses the considerations and guidance for functional multilingual extensions to European keyboard layouts, primarily for users of the Latin script, was agreed by the CEN/ISSS Workshop Functional Multilingual Extensions to European Keyboard Layouts (WS/MEEK) in the Workshop's Kick-Off meeting on 2008-01-25.

The document has been developed through the collaboration of a number of contributing partners in WS/MEEK. The WS/MEEK representation gathers a wide mix of interests, coming from academia, public administrations, IT suppliers, and other interested experts. The present CWA (CEN Workshop Agreement) has received the support of representatives from each of these sectors. A list of experts who have supported the document's contents can be obtained from the CEN/ISSS Secretariat. The final WS/MEEK internal review/endorsement round of this CWA started on 2009-mm-dd and was successfully closed on 2009-mm-dd. In addition, the CWA has been the subject of a public comment period from 2009-mm-dd to 2009-mm-dd. The final text of this CWA was submitted to CEN for publication on 2009-mm-dd.

The document has been developed in close liaison with ISO/IEC JTC 1/SC 35, "User Interfaces", and the Unicode Consortium. SC 35 is responsible within ISO for e.g., keyboard layouts; the participation in ISO is through the National Standards Organizations. Members of the Unicode Consortium include major computer corporations, software producers, database vendors, government ministries, research institutions, international agencies, various user groups, and interested individuals.

This CEN Workshop Agreement has only been made in English.

Introduction

Year 2008 was proclaimed the International Year of Languages by the United Nations General Assembly. It was also declared the European Year of Intercultural Dialogue by the European Parliament and the Council of the European Union. The MEEK Workshop can be seen as a small step in support of these themes.

The current CEN membership consists of 30 countries: Austria, Belgium, Bulgaria, Cyprus, Czech Republic, Denmark, Estonia, Finland, France, Germany, Greece, Hungary, Iceland, Ireland, Italy, Latvia, Lithuania, Luxembourg, Malta, Netherlands, Norway, Poland, Portugal, Romania, Slovakia, Slovenia, Spain, Sweden, Switzerland, and United Kingdom.

These countries have several official languages and even more regional and minority languages. Some of the regional and minority languages have certain official status, either by national law or by the ratification of the applicable Council of Europe Treaty. Of the official languages, Bulgarian is written with Cyrillic letters and Greek with Greek letters, whereas all the others are written with Latin letters. The Latin repertoire includes several letters beyond the “basic a-to-z” and a very large number of letters with diacritics, some of which are perceived as basic letters in one or more languages, and also collated separately (e.g., å, ä and ö after z in Swedish and Finnish).

In addition, many more native and immigrant languages written in a number of different scripts are spoken in Europe.

1 Scope

This CWA is aimed to assist in the preparation of functional multilingual extensions to European keyboard layouts. They are aimed to allow “ordinary users” and Public Authorities to input primarily Latin-script characters, especially in the light of current and potential future legal and common educational requirements. The CWA builds upon existing implementations of common official and de facto standards for national and regional computer keyboard layouts and related input methods. Based on this, practical guidance is given on handling multilingual data entry requirements, taking account of existing international standards in the field. Outside the scope of this CWA are the needs of specialist usage, such as the need to regularly and rapidly generate large volumes of text in a wide range of languages.

Since the European single market allows for free movement of people and goods, one should be able to correctly enter the names of people, places, products, and companies and other legal entities in a consistent, easily comprehensible manner. This implies that the capability should exist in all kinds of applications, including those based on open source software that are traditionally based on freely available information.

The CWA does not define any specific, let alone Pan-European, keyboard layout. Liaison has been established with ISO/IEC JTC1/SC35 to minimize the risk of further divergence between actual implementations and formal standards (ISO/IEC 9995 series and other relevant standards). Liaison has also been established with the Unicode Consortium.

Comment [Marc Küst1]: „sought“ implies for me that we wanted it, but that it was not granted

Deleted: sought

Deleted: sought

2 Informative references

2.1 Overview

In the past, the number of simultaneously representable characters in a computing environment was severely limited because of the 8 bit character encoding schemes in use. These encodings only allowed 256 unique values. In fact, even fewer characters could be represented, as some of the values were reserved for control characters. As a consequence, each country has historically concentrated on having its own official languages properly supported in ICT, more recently also its own regional and minority languages.

The expanding European single market, however, with free movement of goods and people, is drastically changing the support requirements. For both accuracy and politeness, all kinds of proper names should be communicated in their correct form. This change is driven not only by increasing business and legal reasons but also as a personal choice by many individuals in appreciation of cultural diversity.

2.2 Formal reasons for expanding the character repertoire

As witnessed by key legal documents cited here respect for cultural diversity is recognized as a human right in International and European law

2.1.1 The Lisbon treaty

"The Union shall respect cultural, religious and linguistic diversity." (§???)

2.1.2 UNESCO Convention on the Protection and Promotion of the Diversity of Cultural Expressions

Approved in October 2005, entered into force in March 2007, see http://ec.europa.eu/culture/portal/action/diversity/pdf/st08668.EN06.pdf.

- to protect and promote the diversity of cultural expressions
- to create the conditions for cultures to flourish and to freely interact in a mutually beneficial manner
- to encourage dialogue among cultures with a view to ensuring wider and balanced cultural exchanges...
- to foster interculturality...
- to promote respect for the diversity of cultural expressions and raise awareness of its value at the local, national and international levels
- to reaffirm the importance of the link between culture and development for all countries...
- to give recognition to the distinctive nature of cultural activities, goods and services as vehicles of identity, values and meaning
- to reaffirm the sovereign rights of States to maintain, adopt and implement policies and measures that they deem appropriate for the protection and promotion of the diversity of cultural expressions on their territory
- to strengthen international cooperation and solidarity in a spirit of partnership...

2.1.3 European Charter for Regional or Minority Languages

Signed in 1992, in force since 1996, see http://conventions.coe.int/Treaty/en/Treaties/Html/148.htm.

- Concentrates on languages traditionally used in a state by a minority and not an official language of that state.
- Ratification of this Council of Europe treaty (by the national parliaments) entails certain responsibilities, details of which are specified upon ratification:
    - the facilitation and/or encouragement of the use of regional or minority languages, in speech and writing, in public and private life (§7.1.d)
    - the provision of appropriate forms and means for the teaching and study of regional or minority languages at all appropriate stages (§7.1.f)
    - to ensure that users of regional or minority languages may submit oral or written applications [to public authorities] and receive a reply in these languages (§10.1.a.iii)
    - to ensure that users of regional or minority languages may validly submit a document in these languages (§10.1.a.v)
    - to allow the administrative authorities to draft documents in a regional or minority language (§10.1.c)

2.1.4 On the recording of surnames and forenames in civil status registers, Convention relative à l'indication des noms et prénoms dans les registres de l'État Civil, CIEC 14

Signed in 1973 in the context of the International Commission on Civil Status, the Commission Internationale de l'État Civil. Some member states have not signed it and some signatories have not ratified it, though. The official French text and the unofficial English translation are available at http://www.ciec1.org/ ListeConventions.htm.

Spelling of names (Article 2):

- Where a record is to be made in a civil register by an authority of a Contracting State and there is produced for that purpose a copy of or extract from a civil status record or some other document that shows the surnames and forenames in the same characters as those used in the language in which the record is to be made, those surnames and forenames shall be reproduced literally without alteration or translation.
- Any diacritic marks forming part of such surnames and forenames shall also be reproduced, even if such marks do not exist in the language in which the record is to be made.
- Where a record is to be made in a civil status register by an authority of a Contracting State and there is produced for that purpose a copy of or extract from a civil status record or some other document that shows the surnames and forenames in characters other than those used in the language in which the record is to be made, those surnames and fore-names shall be reproduced as far as possible by transliteration, without being translated.
- If there are standards recommended by the International Organisation for Standardisation (ISO), they shall be applied.

In practice, this would mean that letters such as the Danish (also Norwegian, Icelandic, Faroese, and Kalaallisut) letter AE (æ, which is an ae-ligature in some European languages), or the Icelandic THORN (þ->th) and (also Faroese) ETH (ð->d), and several other characters would not have to be supported in several other countries as such. It would appear from the timing of this agreement that a stroke should probably be treated as a diacritic.

2.1.5 German-Polish Treaty on good neighbourly relations and friendly cooperation

Signed 17 June 1991, see http://untreaty.un.org/unts/60001_120000/28/1/00054050.pdf.

Art. 20 § 3 … shall in particular have the right, individually or in community with other members of their group:

- To use their mother tongue freely, both privately and in public, to disseminate and exchange information in it and to have access to such information; …
- To express their given and family names in the mother tongue form; …

§ 4 … membership in the groups referred to in § 1 …may not place him at any disadvantage.

Background

Most of the keyboard layouts presently in common use in Europe have been designed at a time when the number of different characters that could be entered was severely limited by the then prevalent character encoding schemes. In addition, some limitations of the original mechanical typewriters have been carried over into their design. These keyboards have been designed for each local language environment, although often with some capability to enter “foreign” letters.

The advent of the Universal Character Set (ISO/IEC 10646 and Unicode) has fortunately removed the limitations imposed by the ICT systems on the number of different characters to be handled, i.e., processed and rendered, at once. Consequently, one could expect that any European solution for interoperable public registers is likely to follow the lines of CIEC 14 (ref. 2.1.4 above), which is not being currently enforced. Also, some countries are already committed to using the proper names in their formally recognized minority languages as a result of the European Charter on Regional or Minority Languages.

What remains to be done is to provide a way to enter the correct data. To avoid disrupting current users of existing national and regional keyboards, the mechanisms have to build on the way the keyboards are currently used instead of starting from scratch. As the formal international standards (the ISO/IEC 9995 series) don’t address national layouts beyond providing the base to build on, there is an urgent need to come up with ways to meet the rapidly emerging needs.

4 Usage scenarios

There can be found three basic scenarios in which users have a need to enter characters which are not directly provided by present keyboards.

(1) The first scenario involves a user who needs to enter characters from one or more foreign languages, using his or her own national keyboard. An example would be a clerk entering a foreign name.

(2) In the second scenario, a user needs to enter foreign characters on a foreign keyboard. An example would be a clerk in a multilingual work environment who needs to enter foreign names and must also be able to move from one workstation (and keyboard) to another.

(3) The third scenario applies to users who visit Internet cafes or must work in a foreign office or use a foreign keyboard for some other reason. This scenario also applies to multilingual users. The common factors are that these users need to produce text for some language which is not supported by the keyboard they are using, and they need to enter the characters frequently.

Scenarios one and two should be solved with the same mechanism. In both scenarios, the user must enter characters from a vast array of possible characters. The characters can be alien to the user.

The user will need visual aids for locating and selecting the correct characters. There should also be textual aids to help in the selection process, so the user can make the correct decision, e.g., information on the languages and contexts where the character is used. These visual aids and documentation are outside the scope of this document.

In scenario 3, users wish to enter text in a single language, other than the one the keyboard was designed for. Typically, users will frequently need to enter certain letters of their own alphabet and that small set of special characters that are used in e-mail addresses and website URLs (e.g. ~). The repertoire of the needed characters is probably limited to 1-6 characters (e.g., a Finnish user on a French keyboard should probably get by with finding the characters å, ä, ö, @ and ~).

Users should ideally be able to enter the characters using the same method, regardless of which foreign keyboard is being used. This means that once users learn how to enter their own language's characters on one foreign keyboard, they can use the skill on all foreign keyboards. It is important that as few key strokes as possible are required for entering text, as the user will be needing them often.

The solution should not rely on an Input Method (IM) for the targeted language being available on the system. The IMs that are used should be designed for people who are not touch typists, and who therefore rely on keyboard engravings. After all, touch typists could just switch to their national layout on any keyboard. Since they don't need the engravings, they can use any keyboard, regardless of the engravings, provided that their familiar keyboard driver is available.

Comment [Marc Küst2]: Except when said touch typists try to login with the keyboard layout set to another locale then their own (a frequent problem because in many systems users can change the keyboard layout only after successful login)

It should be recognized that touch typists and users who rely on keyboard engravings view keyboards differently. As an example, a touch typist who is accustomed to a qwerty layout, but using a qwertz keyboard, thinks the letter ‘y’ should be on the key with ‘z’ engraved. From a non-touch typist's perspective, the key with the engraving for ‘y’ should produce ‘y’, even though he or she might first look for ‘y’ in the position after “qwert”, with the engraving for ‘z’.

It should be apparent from the above example that it is impossible to create a mechanism to allow users to switch between keyboards in a totally transparent manner. Some keys will always have conflicting engravings on different national keyboards.

5 Proposed solutions

5.1 A pragmatic approach to extend current keyboards

One approach which has been made into a national standard, with several experimental implementations, uses the dead key method. Many European users of keyboards, other than the UK one, are familiar with the dead key method for keying “foreign” letters with diacritics. In this method, the key for the required diacritical mark is pressed first, followed by the base letter with which the diacritic is to be combined. This method has been in wide use since early mechanical typewriters, and there could be considerable benefit in continuing its use and expanding it to cover essentially all letters with a single diacritic. It should be noted, though, that most users consider as base characters those letters of their own alphabet that have diacritics, and thus cannot be asked to utilize any other method than a single keystroke to enter them on their national keyboards. Such letters, e.g., å, ä and ö in Swedish and Finnish, ñ in Spanish, etc. will thus have to continue to appear directly on their respective keyboards, although they would be created stepwise, as combinations, on most other keyboards.

Although one could envisage the dead key method to be extended to entering letters with multiple diacritics, this is not generally supported by the current table-driven implementation methods used by some major players in the industry to create keyboard drivers (Microsoft, in particular). This, however, could well be the method of choice for many among the few users who need to enter characters with multiple diacritics. This method of character entry will be referred to as the "extended dead key method" in the remainder of the document.

Several of the letters with multiple diacritics (e.g., many of the Lithuanian accented letters) cannot be encoded as precomposed characters, and they will thus have to be generated as decomposed characters by the keyboard driver. They could also be entered using the above extended dead key method.

Another method to enter composite characters would be to enter the base letter followed by the combining diacritical marks. This method could be more suitable for those users who are not familiar with the dead key method. It could also be easier and possibly faster to implement than the support for multiple diacritics using the dead key method. It would not, however, produce precomposed encoding even for those characters that could be precomposed, unless the driver would be accompanied by a normalization routine.

If both of these modes of operation are to be supported, it would make sense to implement them as two separate keyboard drivers utilizing the same keyboard layout.

The underlying principle for the design approaches described herein is not to impose any disturbing changes to those users who are content with current implementations. Those who need additional capabilities could easily choose between the two keyboard drivers, as appropriate.

The dead key method could, and probably should, also be extended to letters with diacritic-like additions, specifically to characters with a stroke (although they are not marked as decomposable in terms of Unicode character properties). This would help avoid having to assign key positions to each of the various letters, such as the Polish L WITH STROKE [ł] or the Sámi T WITH STROKE [ŧ]. In addition to keeping key positions available for other use, this would also meet the criterion that the method and the positioning of the characters should be intuitive.

For ease of use, the diacritical marks should be engraved on the key tops, since they can hardly be placed following any intuitively recognizable scheme. If properly placed, the other additional characters don’t necessarily need to be engraved.

It should be noted that the intuitive position for a character is not absolute. A user who knows the pronunciation of a character will likely think that its intuitive position is on a key which corresponds to its pronunciation. However, a user who is not familiar with the character is likely to think that the intuitive position is on a key which matches its visual appearance. A good example is the Kalaallisut or Greenlandic letter KRA (ĸ) with an apparent similarity to letter K but with a different pronunciation.

The details pertaining to the “pragmatic” design principles described above are the result of the discussions at the BOF session on the Design Principles for a Regional, Multilingual Keyboard at the 29th IUC

(Internationalization and Unicode Conference) in San Francisco on 6 March 2006 (see: http://www.unicodeconference.org/iuc29/program-d.htm#BOF_3).

A sample specification (for Finnish and Swedish and the recognized minority languages in the Nordic countries plus the official EU languages using the Latin script) was published in Finland in November 2008 as SFS 5966, “Keyboard layout: Finnish-Swedish multilingual keyboard setting”. In addition, this layout also provides punctuation used in a number of European languages. Since 2006 there had been available at least three different evaluation or trial versions implementing a near-final basic (precomposed) mode of operation, two for the Windows environment (of which one was a formal evaluation version by Microsoft) and one for the X11 (GNU/Linux etc.) environment. To illustrate its use, Annex 1 contains an informal translation of the text of a quick users' guide produced by SFS, the Finnish Standards Association.

For comparison, a keyboard layout intended for governmental use with a similar multilingual repertoire had been published in Sweden as SS 662241/T1:2006. The approach taken there was to distribute a large number of characters with diacritics or strokes over practically all available key positions.

It should be noted that the requirement to cover the characters of the local minority languages, which the Finnish standard responds to, is of course dependent on the region.

5.2 A global input method

A proposed design for Scenario 3 relies on access to IM configuration files which may not be already installed on the system. Each of these IM files relates to the user's requirements for their own needed characters and the specific keyboard designs that are being used. These IM configuration files could be obtained in a simple manner from an online store or could be made available as part of a general session personalization process (e.g. as proposed in ETSI EG 202 325 and, more specifically in relation to public internet access, in ETSI TS 102 577).

In addition to a library of such default common cross-keyboard IM (configuration) files, the proposed solution describes a very simple mechanism that would permit users to create their own personalized IM files using a simple user interface that could be installed on all PCs for public use. These IM files could be created in advance for later use or could be created on-the-fly when first encountering a new keyboard and stored online for subsequent reuse. These personalized IM files could contain mappings for characters that are of particular interest to that specific user and could be located in a position that the user prefers, rather than in the default position that would exist in the “standard” IM files that were available.

5.3 Ongoing work on international standardization

The requirements discussed in this paper are recognized also by ISO (International Organization for Standardization). Therefore, there is ongoing work on the following projects:

1. The standard ISO/IEC 9995-3 "Keyboard layouts for text and office systems – Complementary layouts of the alphanumeric zone of the alphanumeric section" is currently undergoing a major revision.

This standard defines a "secondary group", which means that two other characters will be associated with every alphanumeric key (unshifted and shifted). They will be entered by enabling this "secondary group" by a special key (which may be the "AltGr" key present on some keyboard layouts, or a dedicated "Group 2 select" key). The secondary group contains characters, symbols etc. It also contains diacritical marks, which adhere to the "dead key" mechanism described in section 5.1.

This secondary group as defined by the present edition ISO/IEC 9995-3:2002, however, does not cover all characters needed for the European languages even if only the ones using the Latin script are considered.

This is likely to be changed by the ongoing revision, which is now at the FCD (Final Committee Draft) stage; the DIS version (Draft International Standard) could thus be submitted to the combined ISO and IEC ballot at the earliest towards the end of 2009.

This revision is intended to contain all Latin characters in modern use in the official languages of Europe, besides the letters needed for South Africa and the indigenous languages of the Americas. It employs a third level (additional to "unshifted" and "shifted"), and also strikethrough diacritics to select characters with a stroke as outlined in the previous section.

However, the arrangement of the characters in the secondary group is constrained by compatibility requirements of the existing applications (especially a Canadian standard layout) and therefore is not considered to be optimal. (But the latter also can be said about the base qwerty/qwertz layout itself.)

The keyboard layout shown in Annex 3 is an illustration of the current status of the ongoing revision work.

2. A new ISO project has been started from scratch with the tentative name "ISO/IEC 9995-9: Multilingual-usage, multiscript keyboard group layouts". It is currently a New Work Item without a formal working draft or schedule.

The basic idea is to assign groups (e.g. sets of characters belonging to a script) to the 26 letter keys of the Latin layout, which can be switched by special key sequences.

The keyboard layout shown in Annex 3 uses these mechanisms to enable the typing of Cyrillic and Greek as well as Latin letters, especially to enable convenient entering of a set of Cyrillic letters which is considerably larger than the basic set of 26 Latin letters.

6 Summary

In the solution described as “pragmatic” in 5.1, all base characters (without diacritics) can be found in levels 1-4 (with stroke-modified letters considered having a diacritic). All letters with a diacritic are entered by first entering a diacritic, which is behind one key in levels 1-4, and then a base letter. In addition, nationally defined characters with a diacritic can have their own position so that they can be produced with a single key in levels 1-4. The SFS solution given as an example also covers many symbols commonly used in Europe. Additional work is required to cover all needed symbols, even if the work ends up concluding that all symbols are covered.

The global IM solution covers all characters, and even non-character strings. No implementations are currently available, but due to its simplicity of implementation, one could be achieved in a short amount of time. The advantage is that all characters can be entered with a single keystroke in levels 1-4, even characters with a diacritic (as well as even a limited number of characters with multiple diacritics). The disadvantage is that users must always select a specific keyboard layout before starting to type (for sometimes a very small amount of text). This burden can be reduced by using a simple interface for selecting the layout.

Extended solutions based on the future ISO/IEC-9995-9 may cover all Latin characters, not just those in use in Europe. They may include support for the Cyrillic, Greek and other scripts, and symbols. The principal advantage is that once a user has learned the method, it will work on all keyboards. The disadvantages are that (a) all “international” letters (outside “a-zA-Z0-9,.-”) require multiple keystrokes and (b) each character's sequence must be memorized, which can be difficult for ordinary, casual users. No implementation is currently available. Implementation work probably cannot start before the method has been finalized.

7 Assessment

A desirable characteristic for any expansion of keyboard capabilities is that it should preferably be intuitive and, as such, easily comprehensible and carefully based on current layouts. The SFS 5966 keyboard is a good sample solution for usage scenario 1 for Finland, as it is based on the existing Finnish keyboard. Scenario 1 can be solved for other countries using similar methods, since most users are familiar with their existing national keyboard layout only, and would not welcome any disturbing changes to it.

Comment [Marc Küst3]: striking?

Usage scenario 2 cannot really have an “ideal” solution without the emergence of user friendly technologies that provide for dynamic changing of keytop engravings (i.e., displays).

Usage scenario 3 cannot be solved satisfactorily with the SFS 5966 solution, as frequent use of certain characters requires that those characters be produced with a single key press (with at most one modifier key in addition to the shift key). Multiple key presses to generate one character, except in the rare cases of characters with multiple diacritics, is not an acceptable solution for a user who needs to enter large volumes of text.

To be effective, it should be possible to implement the solution for scenario 3 in a range of countries without waiting for new keyboards to be introduced in all of these countries. It is also important that the solution will not negatively affect residents of a country who wish to use the keyboards in ways that are already familiar to them. The Global IM solution fits this requirement as it can be applied to any existing keyboard (typically a national keyboard) or to any new keyboard that solves scenarios one and two.

As the future Annex 3 (ISO/IEC-9995-9) type solutions attempt to provide a universal solution to the generation of the widest range of characters, they can be said to provide a solution to all three scenarios. However, they are more complex than most existing keyboard solutions and represent quite a wide departure from the keyboards with which users are currently familiar. This characteristic is a factor that makes ISO/IEC-9995-9 type solutions difficult to introduce without causing a degree of disruption to the well-learnt familiarity that users have with the keyboards that they currently use. Such a disruption could be a particular problem in scenario 3 where users wish to use to an unfamiliar keyboard to generate the characters that they normally generate on their own keyboards without the frequent use of multiple keystrokes.

8 Further considerations

One should not ignore the ramifications of the requirements by the Greeks (users of the Greek script) and Bulgarians (users of the Cyrillic script) to enter Latin characters and vice versa, although the details thereof are clearly not within the scope of this workshop. In Greece, in practice, qwerty keyboard layouts with the addition of Greek characters are common in hotels et al., and they could be extended to cover multilingual Latin characters along the lines proposed here. It should also be noted that new member states may use the Cyrillic script in a more complex form than that needed for Bulgarian.

Whenever a keyboard layout is being worked on, a requirement is often brought up to revamp the layout and base it on the Dvorak keyboard. Any such activity would clearly not be within the scope of this Workshop. Additionally, although the Dvorak keyboard is based specifically on the English language, it has not gained wide acceptance among the English speaking users in spite of it having been available since pre-WWII time. Furthermore, extensive analysis would be required to come up with a specific layout for any other language.

Another requirement stated sometimes is the design for a Pan-European keyboard layout which, however, would not be practical and is clearly outside the scope of this Workshop.

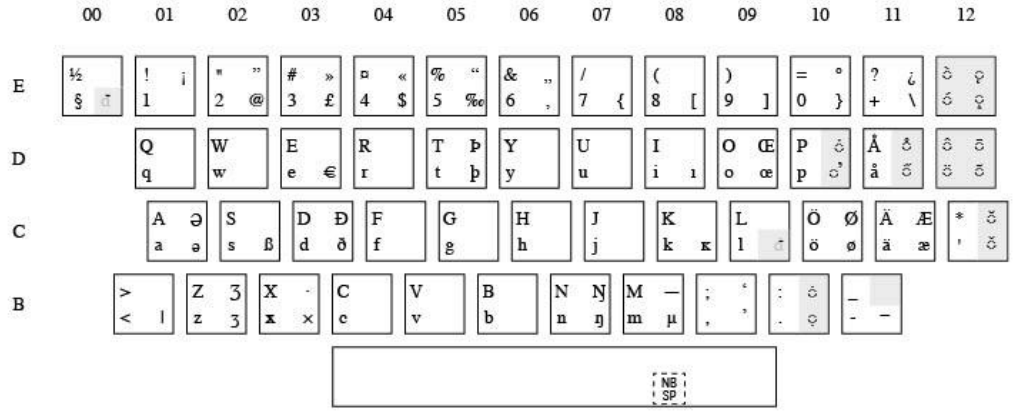
Other methods include special mnemonics for use in sequences to create characters with diacritics. These have been used widely to represent non-ASCII characters in system tables that had to be defined in ASCII, particularly in the UNIX environment, but they cannot be characterized as being easily comprehensible to non-technical users.

Annex 1: A quick guide to the basic setting of the keyboard layout standard SFS 5966

NOTE: The Finnish keyboard layout described here is an example how an extension of a national layout can be done to accomplish the goals outlined in the paper. The selection of this example does not imply that the decisions regarding specific key locations, key access methods, etc. done for this standard are recommended by MEEK as preferred ways to accomplish the MEEK goals.

Character layout

(combining characters in grey)



The Finnish-Swedish general purpose keyboard layout has been enhanced with new capabilities. Its prior functionality, however, is not affected except in rare special situations. The new functionality is typically effected by pressing the AltGr key, situated to the right of the space bar.

Additions to the layout include several letters, punctuation marks and special characters used in many European languages. The goal has been to place them in intuitively recognizable key positions.

The diacritical marks are added to the base letters utilizing the so called dead key method. In this method, the diacritical mark is being entered ahead of the letter to which it will be combined. While all diacritical marks used to be positioned on the two specific diacritical keys, new diacritical marks have been added to other keys as well.

Also the letters with a stroke are being produced utilizing the dead key method as if the stroke were a diacritical mark.

Additional letters

The new letters are being produced by pressing the AltGr key together with the appropriate letter key. The capital letters are being produced by pressing also the shift key.

Key	**Resulting letter**	**Glyph**	**Exemplar language**
letter A	small and capital letter schwa	[ə Ə]	Azerbaijani
letter D	small and capital letter eth	[ð Ð]	Icelandic

letter I	small letter dotless i	[ı]	Turkish
letter K	small letter kra	[ĸ]	Kalaallisut
letter N	small and capital letter eng	[ŋ Ŋ]	Northern Sámi
letter O	small and capital ligature oe	[œ Œ]	French
letter S	small letter sharp s	[ß]	German
letter T	small and capital letter thorn	[þ Þ]	Icelandic
letter Z	small and capital letter ezh	[ʒ Ʒ]	Skolt Sámi
letter Ä	small and capital letter ae	[æ Æ]	Danish
letter Ö	small and capital letter o with stroke	[ø Ø]	Danish

Letters with stroke

A stroke will be produced by pressing simultaneously the AltGr and § keys. Instead of the § key, also the L key may be used, since many laptops don't have the § key at all. The stroke will be followed by the letter to which it will be attached. For capital letters, the shift key will be pressed together with the letter key.

The stroke can be added to the following letters (with the exemplar language in parenthesis):

letter D [đ Đ] (Northern Sámi), letter G [ǧ Ǧ] (Skolt Sámi), letter H [ħ Ħ] (Maltese), letter L [ł Ł] (Polish), letter O [ø Ø] (Danish) and letter T [ŧ Ŧ] (Northern Sámi).

Additional special characters

Key	Position	Resulting	Glyph
# 5	AltGr	per mille sign	[‰]
# 0	AltGr+shift	degree sign	[°]
letter E	AltGr	euro sign	[€]
letter M	AltGr	micro sign	[µ]
letter X	AltGr	multiplication sign	[×]
letter X	AltGr+shift	middle dot	[•]

Additional punctuation marks

Key	Position	Resulting character	Glyph	Exemplar language
# 1	AltGr+shift	inverted exclamation mark	[¡]	Spanish
# 2	AltGr+shift	quotation mark	[”]	Finnish
# 3	AltGr+shift	double angle quotation mark	[»]	Finnish
# 4	AltGr+shift	double angle quotation mark	[«]	Danish
# 5	AltGr+shift	quotation mark	[“]	English
# 6	AltGr	single low quotation mark	[‚]	German
# 6	AltGr+shift	low quotation mark	[„]	German
Plus sign	AltGr+shift	inverted question mark	[¿]	Spanish
letter M	AltGr+shift	m-dash	[—]	US English
Comma	AltGr	single quotation mark	[’]	Finnish
Comma	AltGr+shift	single quotation mark	[‘]	English
Hyphen-minus	AltGr	n-dash	[–]	Finnish
Space bar	AltGr	no-break space	[ ]	Finnish

Additional combining diacritical marks and their new functionality

With the new keyboard setting, the character repertoire that can be produced with the traditional diacritical marks has been extended. This also leads to some functional differences. If the diacritical mark is followed by a character to which it cannot be attached, the result is a separate character resembling the diacritical mark. Thus, e.g., a circumflex followed by letter c will now produce letter ĉ, whereas it used to produce the string ^c. That string is now produced by a circumflex followed by a space and letter c.

The traditional diacritical marks can now be used to create the following European small and capital letters:

Acute	[´]	á Á, ć Ć, é É, í Í, ĺ Ĺ, ń Ń, ó Ó, ŕ Ŕ, ś Ś, ú Ú, ẃ Ẃ, ý Ý, ź Ź, ǽ Ǽ and ǿ Ǿ
Grave	[`]	à À, è È, ì Ì, ò Ò, ù Ù, ẁ Ẁ and ỳ Ỳ
Diaeresis	[¨]	ä Ä, ë Ë, ï Ï, ö Ö, ü Ü, ẅ Ẅ and ÿ Ÿ
Circumflex	[^]	â Â, ĉ Ĉ, ê Ê, ĝ Ĝ, ĥ Ĥ, î Î, ĵ Ĵ, ô Ô, ŝ Ŝ, û Û, ŵ Ŵ and ŷ Ŷ
Tilde	[~] (AltGr)	ã Ã, ẽ Ẽ, ĩ Ĩ, ñ Ñ, õ Õ, ũ Ũ and ỹ Ỹ

The added diacritical marks can be used to create the following European small and capital letters:

Diacritical		**Key**	**Position**	**Letters**
Cedilla	[¸]	Acute [´]	AltGr	ç Ç, ģ Ģ, ķ Ķ, ļ Ļ, ņ Ņ, ŗ Ŗ, ş Ş and ţ Ţ
Ogonek	[˛]	Acute [´]	AltGr+shift	ą Ą, ę Ę, į Į, ǫ Ǫ and ų Ų
Double acute	[˝]	letter Å	AltGr	ő Ő and ű Ű
Ring above	[˚]	letter Å	AltGr+shift	å Å and ů Ů
Macron	[¯]	Diaeresis [¨]	AltGr+shift	ā Ā, ē Ē, ī Ī, ō Ō, ū Ū, ā̃ Ā̃ and ǣ Ǣ
Caron	[ˇ]	Apostrophe [']	AltGr	ǎ Ǎ, č Č, ď Ď, ě Ě, ǧ Ǧ, ȟ Ȟ, ǐ Ǐ, ǩ Ǩ, ľ Ľ, ň Ň, ǒ Ǒ, ř Ř, š Š, ť Ť, ǔ Ǔ, ž Ž and ǯ Ǯ
Breve	[˘]	Apostrophe [']	AltGr+shift	ă Ă, ĕ Ĕ, ğ Ğ, ĭ Ĭ, ŏ Ŏ and ŭ Ŭ
Dot below		Full stop [.]	AltGr	ạ Ạ, ẹ Ẹ, ị Ị, ọ Ọ, ụ Ụ and ỵ Ỵ
Dot above	[˙]	Full stop [.]	AltGr+shift	ḃ Ḃ, ċ Ċ, ḋ Ḋ, ė Ė, ḟ Ḟ, ġ Ġ, İ, ṁ Ṁ, ṗ Ṗ, ṡ Ṡ, ṫ Ṫ and ż Ż

Optionally implemented diacritical marks for Vietnamese:

Horn	letter P	AltGr	ơ Ơ and ư Ư
Hook above	letter P	AltGr+shift	ả Ả, ẻ Ẻ, ỉ Ỉ, ỏ Ỏ, ủ Ủ and ỷ Ỷ

In addition, the following position is reserved for use to support Romanian:

Comma below	Hyphen-minus [-]	AltGr+shift	ș Ș and ț Ț

Annex 2: Description of a proposal for a Global Input Method

Description of the Problem

There are multiple scenarios for users of keyboards, based on their needs to enter text. This proposal is aimed for a scenario where the user is using a keyboard and needs to enter text in a single language, other than the one the keyboard was designed for. Further, an IM for the targeted language is not available on the system. In this scenario, the user will typically have to enter certain letters of her own alphabet frequently. The repertoire of the needed characters is probably limited to 1-5 characters (e.g. a Finnish user on a French keyboard should get by with finding the letters å, ä, ö and @). The user should ideally be able to enter the characters using the same method, regardless of which foreign keyboard is being used. This means that once the user has learned to enter his or her own language's characters on one foreign keyboard, he or she can use the skill on all foreign keyboards. It is important that as few key strokes as possible are required for entering text, as the user will be needing them often.

What would this group of users want? Ideally they would want to enter text just like they do on their own national keyboards. This goal is not achievable, however, because of the sheer number of keyboard layouts and the many conflicting key assignments (qwerty, azerty, languages with many composed characters etc.). Also, the extra keys they need are usually not engraved on the keyboard. Because any combination of (x) physical layout and (y) the user's desired layout is possible, it is impossible to design a universal system where users always get exactly what they want. When a user sits down in front of a computer, we know x (the physical layout), but not y (user's desired layout), and we know that we want the new layout to be as close to y as possible. Therefore, we can devise a scheme where we ask the user for the value of y (the desired layout), so that we can provide the user a layout which is optimized for x and y. Designing a layout for each combination of x and y is not really a problem, at least for many of the combinations, since we can restrict the keys used in the layout to a set which is available on all values of x (national keyboards). A problem arises if we consider the fact that some users will also use some characters from outside y (e.g. symbols from a neighboring country), which they have available on their national keyboards at some nationally defined locations (e.g. Sámi, German and Danish characters on a Finnish keyboard). In other words, the way users' desires differ is not really a known quantity.

Some keys on a keyboard are inherently more convenient than others. If we have predefined positions for all characters, then some keys are always less convenient for somebody.

The use of diacritics is sometimes needed, especially in some languages, but most users prefer to enter combined characters from their own language with a single key stroke.

A very common scenario is that the national keyboard (the local keyboard) has keys which the foreign user does not need. The keys assigned to local uses on the national keyboard could be used to implement some of the characters in the user's own set, the characters he or she needs the most often. E.g. a Finnish user using a Spanish keyboard could have 'ñ' mapped to 'ä' and 'ç' to 'ö'. This is only possible if we have keyboard layouts designed for each combination of x (national keyboard) and y (user's desired layout), or if the user is able to modify the mappings herself.

For some languages, it is also common to use an Input Method (IM) for entering text. An IM is a piece of software which provides a service very similar to a keyboard driver, but includes some extra functionality. E.g. Microsoft's Japanese IM provides various convenience facilities such as dictionary lookups. This proposal involves the use of an IM. There is nothing which prevents a modern IM from using the network, or from presenting the user with graphic presentations which are pleasing to the eye, with menus and other facilities.

An input method for Internet cafe users needs to cater to the needs of various users from various locales. The keyboard is usually a national, US or UK one. Each user group has a finite set of characters which needs to be produced. In most cases with Latin characters, the number of characters is small. In some cases (Cyrillic, Greek) it is larger. In addition to the characters, some symbols must also be produced. It would be ideal, from the user's perspective, if we could turn the keyboard to a mode where the necessary

characters and symbols are produced using either the key directly (level 1), with shift (level 2), with AltGr (level 3) or AltGr with shift (level 4).

Which is more important for users, having the same key positions on all foreign keyboards, or having the most needed characters in easy to use (but possibly differing) key positions on each foreign keyboard? Can we provide users with both options?

Since the user is foreign to the system, and will possibly only use it rarely, the number of things to memorize should be limited to the minimum.

The Proposed Solution

The above described issues were the premise for this proposal.

Since there is a need to have a full IM anyway, why not come up with a method where the user can select from multiple different layouts, and select the one he or she likes the best? The user could choose a national set of characters, or one with some of his or her own characters. The user should also be able to specify the key positions for entering the characters.

Of course, it would be cumbersome to specify a new layout each time. So why not allow the user to save the settings in some global location? Why not have servers where the user's own settings are stored, and can be retrieved from anywhere in the world?

By allowing users to specify key positions, we can allow everybody to use the ergonomically best positions. We can also deliver a large character repertoire. Actually, the repertoire can include all characters and symbols in UCS/Unicode.

When a user arrives at a new computer, there should be a single button (or a well known command which is common to all systems) which the user can use to get started. Reasonably, this method should be a menu or a menu item. This is achievable on all personal computer systems, and even on portable devices such as phones.

Brief Description of the Solution

Users are provided with a single menu or menu item for configuration. The configuration screen allows users to configure the system and to load a new layout. Users also select a key combination to switch between the keyboard layouts (one layout is the national keyboard layout for the physical keyboard, and the other one is the layout loaded by the user). The default layout is the one provided by the national keyboard and its extensions. The user loaded layout is superimposed on top of the national layout, so any keys not specified in the user loaded layout will work the same as they do in the national layout. Also, when switched back to the national layout, the default layout will continue working as before.

Requirements from a Compliant System:

- No administrative privileges required to use.
- Single menu or menu item to activate configuration.

Needs from the Method:

- Must work on any national keyboard which is able to produce the ASCII letters "A-Z/,.+-".
- Minimal keystrokes to produce characters.
- Must be able to produce all national characters and symbols for the languages that are to be supported.
- Must work even without a network connection.

The system is comprised of the following components within the operating system:

1. Method for activating the configuration menu.
    - This should be a single menu or menu item.
2. Configuration system. This is the system used to select a User Customized Layout (herein abbreviated as UCL) and to save one.
3. Method for switching between keyboard layouts (local layout and the UCL).
4. Method for latching to a UCL (for entering a single character from the UCL).
5. IM which implements the UCL.

Other related components within the operating system:

1. National keyboard layout.
2. Other IMs installed on the system.

In addition, there are servers which store the UCLs. These are regular HTTP (www) servers which store the UCL files.

A. Configuration System

This is the system used to select a UCL and to save one.

The user is presented with a screen where he or she can create a new UCL, modify one, and save one.

A computer should cache some common UCLs, such as those created for various locales. This way a network connection is not needed for common usage. If no usable UCL is found and no network connection can be established, the user can always create a UCL manually.

For phones and other systems with a limited GUI, there could be a universal (or a device type specific) command for loading a new UCL.

B. UCL Design Tool

The UCL design tool allows the user to create a customized UCL. The UCL design tool should have a map with all UCS/Unicode characters laid out in tables, and other tools.

Automatically Created UCLs: It should be possible to create default variants for all the major locales using CLDR's exemplar characters. This way we would have a sample UCL for each locale in CLDR.

Using this method, we are also in the position to allow users to specify new positions for characters such "$@{}[]()", as it suits them. These and other symbols are often in different locations on national keyboards, and their use varies by user group. Programmers need them often, and having a programmer specific UCL, programmers could be assured that they can find all special symbols which are found on a US-ASCII keyboard.

C. Switching/Latching to a UCL

Switching:

A method for switching between keyboard layouts (local layout and the UCL):

- One command (e.g. CTRL+'/'). Every time the command is executed, the system switches between the default layout and the user's chosen UCL.
- Two commands (e.g. CTRL+'/' and CTRL+'\'). Each command always switches to its designated layout. E.g. CTRL+'/' could always switch to the default layout, while CTRL+'\' could always switch to the user's UCL.

Latching:

Method for latching to a UCL.

A command (e.g. CTRL+',') which allows the user to enter a single character using the UCL, and then changes back to the previous layout.

D. Protocol for Reading a UCL from a Server

UCL files are stored on a server. In order to download a UCL, the user only needs to know the hostname (server address) and the UCL's name.

E. Protocol for Saving a UCL to a Server

From a user's perspective, only three steps are required: (1) click the save button, (2) login to the server using a web browser, and (3) name the UCL.

F. Additional Notes

The reason we must have key commands for switching and latching is so that users can enable the IM in a fast and easy manner.

All keys not specified in a UCL retain their functionality on the national keyboard.

G. Terminology

UCL	User-customized Layout
Switchin	The layout becomes the active layout for all future input.
Latching	The layout becomes the active layout only until a single a character is output. The previous layout then becomes active again.
IM	Input method
CLDR	Common Locale Data Repository (http://www.unicode.org/CLDR/)

Annex 3: A tentative “Five Scripts Keyboard” layout design

NOTE: This layout design is included as an example of how ongoing work on international standardization is related to the goals outlined in this paper. It is explicitly not the subject of MEEK to develop or propose any Pan-European keyboard. Thus, the inclusion of this example is not to be misunderstood as a proposal.
Also, as this example is related to international standardization work, it includes features which go beyond the specific European needs (while of course not disregarding these).

The Five Scripts Keyboard layout is related to the ongoing work in international standardization. It is based on the common US standard keyboard and is fully compatible with it (especially for touch typists). It is enhanced by an inherent mechanism to switch between Greek, Cyrillic, and other scripts.

Also a second character group is added according to the undergoing revision of the international standard ISO/IEC 9995-3, which is currently (as of October 2009) in the "Final Draft International Standard" ballot state (which means that it is stable in its technical contents and can be reliably referred to). The characters contained in that group are shown in blue in the picture above. (Shown in dark green are characters and functions where the ISO/IEC 9995-3 revision originally has characters which are contained in the base US layout anyway; thus these positions could be used for other purposes.)

This is similar to the way the keyboard layout described in Annex 1 is extended, except that all three levels defined in ISO/IEC 9995-2 are employed ("unshifted", "shifted", and "Level 3").

To select the second group, the "Group 2 select" key is pressed and released before the character key itself is pressed. (This key is optional; it can be replaced by simultaneous pressing of "Shift" + Level 3 select").

To select the third level, "Level 3 Select" (which is called "AltGr" key on most recent keyboards) is pressed together with the wanted key. To make this easier for keys operated with the right hand, an (optional) second such key is employed next to the left "Shift" key. (As this layout maps all Level 3 characters of the secondary group to the Level 3 of the primary group, the Group 2 Select key is not necessary here for the Level 3 characters.)

(The mechanism of the "Group 2 select" key besides a "Level 3 select" key is a technique proven in practice by some keyboard layouts used in Canada. It shall be noted that the "Group 2 select" function can be accomplished also by pressing "Shift" + "Level 3 select" together; thus an ISO/IEC 9995-3 compliant keyboard can be realized using existing hardware layouts.)

Several of the secondary group characters are diacritical marks (which are marked by dotted circles). They are used as "dead keys" in the same way as the keyboard described in Annex 1. These include overstriking diacritics. By the "horizontal stroke" on the K key, several characters like the Croatian/Sámi/Vietnamese Đ/đ (which is different from the Icelandic Ð/ð), Maltese Ħ/ħ, or S-áami Ⱬ/ⱬ/Ŧ/ŧ can be accessed.

It can easily be seen (e.g. by comparison with the layout presented in Annex 1) that the secondary group defined by the revised ISO/IEC 9995-3 contains all characters and diacritical marks needed for Latin-written European languages. Also, it contains several characters needed for good typography (like En-/Em dashes, foot and inch symbols, different quotation marks, etc.), and for transliteration of (e.g.) Arabic and Sanskrit which is needed for scientific and religious purposes.

Moreover, the layout addresses the problem of the non-Latin scripts used in Europe: the Greek and the Cyrillic script. This is done in a way to illustrate the means and the possibilities of the ISO/IEC 9995-9 project ("Multilingual-usage, multiscript keyboard group layouts"), which is currently (October 2009) in an early stage. (In fact, it also addresses the Hebrew script and IPA [the International Phonetic Alphabet], therefore it is called the "Five Script Layout". This abstract naming also implies its design being an example independent of an association to a geographical or cultural area. The special mechanisms provided for IPA, which is important for educational and scientific use, are not subject of MEEK and therefore not discussed here. The inclusion of Hebrew can be explained as Hebrew being part of the European culture by presence of the Jews, but is in fact arbitrary.)

To every letter key of the base US layout, there is associated a uniquely structured set of six characters (two Cyrillic, one Greek, one additional Latin, and two Hebrew). Each of these six positions is associated to a group. This illustrates the main principle of the planned ISO/IEC 9995-9, which associates characters to the 26 keys which are primarily associated to the 26 base letters of the alphabet.

In doing this, the 24 upper-/lowercase letter pairs of the Greek alphabet (plus the final sigma and some special diacritical marks) are associated to these 26 Latin letter keys. When the Greek group is selected, all letter typing occurs in Greek, while all digits and other symbols stay at the same positions on te keyboard. Thus, Greek can be used on any keyboard which employs the switching to the group. (By the way, classic [polytonic] Greek can be typed as well as modern [monotonic] Greek: most of the needed diacritics are mapped to their Latin equivalent, e.g. the koronis [spiritus lenis], is mapped to the "comma above" accent.)

For Cyrillic, this is not as simple. All languages using that script use more than 26 letters; in fact, only considering the languages which are primary languages of a country (like Bulgarian, Russian, Serbian, or Mongolian as an non-European example), 51 upper-/lowercase letter pairs are needed considering only the base letters which cannot represented as a base letter + diacritic pair. Thus, the letters are distributed onto two groups, which are constructed by phonetic, shape, or other pragmatic association to the 26 Latin letters. Whenever a letter of the second group is used, this is done by pressing first the "Special Select" key, which latches to this second group for the next key pressing only. Thus, a mechanism is provided to type Cyrillic names and text on every national keyboard which has the usual 26 Latin letter keys, regardless of the actual arrangement of these, and regardless of (and even without the need to know or even to identify) the language of the names or text (or the national layout of the country of which this language is the primary one).

It has to be pointed out that this is to provide a standardized method to enter any text in Cyrillic script on any keyboard. This method is by no ways intended to replace any national layout of a country using the Cyrillic script.