



**ISO/IEC JTC 1/SC 2/WG 3
7-bit and 8-bit codes and their extension
SECRETARIAT : ELOT**

DOC TYPE :	Proposal for corrigendum / revision
TITLE :	Proposal for a corrigendum / revision of ISO/IEC 2022 to allow, in addition to control characters, the registration of two column graphic character sets in columns 08 and 09
SOURCE :	ELOT, the Greek National Standards Body
PROJECT:	JTC 1.02.02
STATUS :	Proposal
ACTION ID :	ACT
DUE DATE :	-
DISTRIBUTION :	P, O and L Members of ISO/IEC JTC 1/SC 2 WG Conveners, Secretariats WG 3 Members ISO/IEC JTC 1 Secretariat ISO/IEC ITTF
MEDIUM :	P, DOC file
NO OF PAGES :	5

Contact 1: Secretariat ISO/IEC JTC 1/SC 2/WG 3 ELOT Mrs K.Velli (acting)
Acharnon 313, 111 45 Kato Patissia, ATHENS – GREECE
Tel: +30 1 22 80 001 Fax : +30 1 22 86 219 E-mail : kkb@elot.gr

Contact 2 : Convenor ISO/IEC JTC 1/SC 2/WG 3 Mr E.Melagrakis
Acharnon 313, 111 45 Kato Patissia, ATHENS – GREECE
Tel: +30 1 22 80 001 Fax : +30 1 22 86 219 E-mail: eam@elot.gr

The Greek Standards Body and its National Technical Committee on Character Sets would like to submit the following proposal to ISO/IEC/JTC1/SC2 and to SC2/WG3.

The proposal is to make a corrigendum / revision of ISO/IEC 2022 that will allow two-column graphic character sets (in addition to control characters) to be coded in the so-called C1 area of an 8-bit coding environment.

Rationale to use columns 08 and 09 of an 8-bit coding table for both control and graphic characters.

In the 8-bit environment, the character sets defined were based on some well known (at least to Standardizers) International Standards, like for example ISO 8859, ISO 2022, ISO 4873, ISO 2375 etc. Many character sets, compatible with the requirements and specifications of these Standards were registered for use in 8-bit coding environments. Unfortunately, despite the supposed good work of the Standardizers, the actual implementations in the market almost never were fully compatible with the specifications of the coding standards.

One major difference between the actual coded character sets that are used in various Operating Systems and the specifications of the International Standards was the use of the columns 08 and 09 of the coding table. In International Standards these columns are used only to codify control characters, while in most of the Operating Systems of today, these columns are occupied by graphic character sets.

The Greek National Technical Committee strongly believes that the use of columns 08 and 09 for control characters only has a historical background and is not based on actual technological needs. Indeed, the 8-bit character set coding environment was treated in International Standards as a two times 7-bit codification scheme and not as an independent coding environment. This idea of simply doubling the 7-bit codification scheme to produce an 8-bit codification scheme was never accepted by the major vendors. Because of that, the 8-bit coding standards were never "purely" implemented in Operating Systems. The major vendors used the 08 and 09 columns mainly for graphic characters.

As the years went by, there were many efforts to persuade vendors to use International Standards instead of inventing their own, especially in character set coding. Until today, vendors accepted International Coding Standards as much as these Standards did not create any unnecessary obstacle to technology. The approach in character codification was that the International Standards began to be used widely, but unnecessary empty coding positions, like columns 08 and 09, were filled according to the needs of the applications.

It is a major question for the Standardizers: Shall we go on with the approach that we had all these years and simply close our eyes to what industry has done so far and what users have accepted? The Greek National Technical Committee ELOT TE74 strongly believes that it is time to include into the International Standards what is already for many years in the market.

The C1 area is an excellent area to put some additional characters we all need.

It was quickly realized that the introduction of the new European Monetary Sign, the Euro, affects, amongst others, the character sets used in most of the systems of Europe and even worldwide. Many countries would like to have the Euro Sign available in their systems for processing.

The Euro Sign is already codified in a 16-bit coding environment (i.e. in ISO/IEC 10646-1) and thus, those able to use an Operating System that supports the 16-bit coding environment have already a solution, at least for the codification of the character. However, this is not the case for many systems of today that can only support an 8-bit environment. The Euro Sign is not available in any of the 8-bit

systems, and since they still form a very big percentage of the systems available in the market, a quick solution must be found.

We only have two solutions for this problem. Let us examine them.

In the first solution, we choose to be compatible with the existing architecture of the International Standards on 8-bit coding. We only code graphic characters into 94 or 96 character sets, the G-sets. Thus, we are left with the choice either to change most of the existing 8-bit coded character sets or prepare totally new character sets that will include the Euro Sign (and perhaps some other characters we may need), as totally new G-sets registrations.

If we search for a position to code Euro in the existing G-sets (e.g. in the existing 8859 parts), we will immediately realize how difficult it is to find a common position in every code page. Of course, to find a common and unused position is impossible. If, on the other hand, we substitute a character with another, then we risk creating problems for users.

If we produce totally new G-sets adding Euro and possibly a few other characters, then we re-invent most of the 8859 structure. There should be full support from the industry on this choice to provide the new code tables in systems, extensive experimentation on the conversions needed etc. What we actually do in this case is changing what we have done to a complete new environment. Data produced with the old code tables will not be directly readable with the new ones.

As a conclusion, either choice, in the first solution, presents some serious problems:

It is not easy or economical or even feasible to change well-established and implemented character sets. Who is going to sacrifice the investments made for many years and risk the integrity of stored data and go through painful conversion procedures just to get a couple of additional characters?

On the other hand, no-one can imagine that vendors would like to invest on producing totally new 8-bit character sets, with the additional risk to create problems to their own applications, just to offer these few additional characters. Will the users accept not to read directly their old data and instead having to convert them, even at the cost of losing information?

The Greek National Technical Committee strongly favors a second solution, which is the adaptation of the existing coding architecture to the needs of the market and to the actual way vendors have coded graphic characters.

The proposal is to amend the existing coding architecture of ISO/IEC 2022 to allow for the registration of two-column graphic character sets in the C1 area (columns 08 and 09). In the same area, the user should be given the opportunity to invoke either a C1_control character set or a GS (Graphic Special)_graphic character set, a two column graphic character set.

This proposal leaves the existing structure of G sets unchanged and gives us a number of significant advantages and possibilities:

- a. There is no need to change anything in well-established and widely used graphic character sets. Instead, using columns 08 and 09 one has the chance to add some characters that are needed for a specific application, to an existing coding structure.
- b. There can be a direct and immediate registration of two column graphic character sets already available in the market. Most of the major vendors already occupy the column 08 and 09 area with graphic characters. Giving them a chance to register what they are using already is another major step towards bridging the gap between the character set standards and the established situation in the market.

According to our research, many of the problems of the past can be solved if we use the column 08 and 09 area for both graphic and control characters. The Greek National Technical Committee EL07 TE74 has identified and checked a number of issues that can be solved if this proposal is accepted. Amongst many others, we can mention the inclusion of the Euro Sign, special signs for bidirectionality, accents

ISO/IEC JTC 1/SC 2/WG 3 N 431

for polytonic Greek, compatibility with existing local market standards, possibility to convert to ITU character sets with non-advancing characters etc.

ADDENDUM. (1998-03-12)

A response from the Greek National Technical Committee to comments delivered by the Canadian Standards Body (Mr. Alain La Bonte).

The responses are given point by point.

C1. The Canadian Standards Body (CSB) claims that populating the C1 area is not the panacea to solving data interchange in the 8-bit world.

R1. The Greek Standards Body (GSB) agrees.

C2. CSB points that several PC-DOS and OS/2 code pages are full, utilizing the C1 area and even part of the C0 area.

R2. With this comment, CSB implies that in these code pages there is no room for additional characters, like for example Euro, or the characters needed by certain languages, like French. What is obvious is that industry has, since many years, taken steps to utilize the C1 area for purposes other than coding control characters and standardization simply prefers to ignore that fact.

Industry should be given a chance to register what it already uses.

If some of the existing attempts to populate the C1 area are not good enough, then we should have an instrument to register what indeed is good enough for our needs.

C3. CSB: Far-east PC-codes use this area for a valid first byte of a two-byte code.

R3. GSB: This is yet another example that the C1 area has been used for non-control characters. This use of the C1 area should also be preserved. Not only this use is not in conflict with what we are proposing but it is another very good reason why this change should happen as soon as possible.

C4. CSB: Two control characters -SS2 and SS3- belonging to the C1 area are used in certain instances.

R4. GSB: It seems to us that the misunderstanding from the CSB was that they thought that our proposal was to stop using the C1 area for control characters and instead use it for graphic characters only. On the contrary: We favor the use of this area for BOTH control and graphic characters, each set being designated and invoked properly.

C5. CSB: While Esc yy can be used for C1 controls, this is not used by some popular terminals or by ODA.

R5. GSB: The same answer as in Comment 4 applies. Nobody would like to dictate a change to the core software of these applications. They can still use what they use now.

C6. CSB: The available C1 space is grossly inadequate to meet the real needs of the full LATIN set -- 6937 and corresponding Teletex recommendation from ITU.

R6. GSB: That is true, but nobody attempts to do so in our proposal. We just give the existing structure a chance to code some more urgently needed characters, whose use was not clearly foreseen in the past. We are tempted by the comment to say, that since the 8859 series have stopped being a purely graphic one byte code, we can use the C1 area to code any strange animal we can imagine: non-advancing characters or signs of bidirectionality. That could preserve the puristic approach of 8859, if we would like to preserve it.

C7. CSB: As to the IBM EBCDIC to ISO-8 compatibility, IBM has gone through a lot of careful disciplined approach to maintain the equivalence in space between the controls and graphics of ISO-8 (4873) and EBCDIC SBCS codes. There are places in EBCDIC-based software, where code position values less than x3F are discarded as invalid Graphic Characters.

R7. GSB: The most popular 8-bit coded character set in Greece, for many years and till recently, was not the International Standard ISO/IEC 8859/7, but something that IBM had invented and called it IBM 437G. Even now, IBM in Greece utilizes many internal code pages, all of which have some Greek characters in the C1 area. All of these IBM code pages, that even now exist in PCs, in Unix, in AIX, in OS2 and even in Windows, work in harmony with any of the IBM mainframes all over Greece. If this was not true, then 90% of the Greek Banks will not function properly, or all of them will need special conversion software. GSB, having in its experts members from the IT in Banks sector and IBMers, can assure CSB that this does not happens.

C8. CSB: the requirements that are to be met must be spelled out clearly.

R8. GSB: We agree. We have identified a number of sections and paragraphs in ISO/IEC 2022 that need to be amended. The designation and invocation mechanism should be defined as a priority. We think that while the corrigendum or revision of ISO/IEC 2022 will take place, everybody should be encouraged to prepare or even submit the two column graphic character sets available for registration.

C9. A real need covered by 8859-15: French OEs and Y diaeresis. 8859-15 corrects historical mistakes for Finnish also.

R9. The GSB and its National Technical Committee have nothing against the concept of 8859-15. We share much of the sentiments of the editor of 8859-15. The GSB will not stand against the adoption of this Standard, once the majority adopts the position that it is needed.

One thing that should be admitted by everybody, though, is that correcting historical mistakes is one thing and adding Euro or other symbols is another.

The GSB will cordially vote in favor of a standard that is supposed to correct historical mistakes.

The inclusion of the Euro Sign is another thing. The Euro Sign is not needed only by the languages covered by 8859-15 but also by many others. Therefore, changing most, if not all, the parts of 8859 to accommodate Euro is not the solution. The GSB strongly believes that its proposal presents the best solution for Euro in the 8-bit world.