L2/04-365

Title: **The case against encoding the Koalib @-letters**
(Response to Public Review Issue #40: Encoding of Latin Capital and Small Letter "At")

Source: Doug Ewell
Individual member, Unicode Consortium

Status: Individual contribution

Action: For consideration by UTC

Date: 2004-10-20

Public Review Issue #40 concerns the proposed addition of two Latin letters—a duplicate of U+0040 COMMERCIAL AT but with letter-like properties, plus its uppercase counterpart—to the Unicode repertoire, to support the Latin-based orthography of the Koalib language spoken in northern Sudan.

The intent of this proposal is to provide support for writing a minority language, which is certainly in the best spirit of the Universal Character Set. However, encoding these particular characters, especially the lowercase letter, could cause serious problems involving Internet security, significantly outweighing the stated advantages.

The @ sign has been in common use for centuries, originally in Spain and Portugal as a symbol denoting a unit of weight, and subsequently as a logogram meaning "at" or more specifically "at the price of." But even this widespread usage has been outstripped by the explosive growth of Internet e-mail, which has used the @ sign since 1972 in the sense of "at" to separate the user name from the domain name, as in "somebody@somewhere.com".

The association of @ with the Internet has been so prominent that companies have begun using it to present a "modern," globally interconnected image. For example, the Swatch company uses a bold red @ sign as the symbol for its "Swatch Internet Time" concept. It has been used in similar marketing contexts with increasing regularity, typically to replace the letter "a" in ordinary words. It has not generally been thought of as part of an actual spelling system, and had not been proposed for encoding as such until the Koalib proposal.

Koalib is one of more than 100 minority languages in a country where Standard Arabic is dominant (Ethnologue). The number of Koalib speakers is estimated at just over 44,000. The literacy rate in Sudan as a whole is estimated to be 20 to 27 percent; if this figure can be applied to the Nubian Desert region where Koalib is spoken, the number of literate Koalib speakers can be roughly estimated at anywhere from 9,000 to 12,000.

By contrast, one source (www.internetworldstats.com) estimates the number of worldwide Internet users—many of whom are e-mail users, who understand the special

meaning of the @ sign—at more than 812 *million*. In Africa alone, by far the region of the world with the *least* Internet usage per capita, there are estimated to be almost 13 million Internet users, more than *one thousand times* the number of literate Koalib speakers. The potential confusion of encoding two different @ signs could seriously affect the African population the proposal attempts to help.

The potential for intentional misuse of the proposed characters for "spoofing," or using look-alike characters in an attempt to mislead readers, should not be underestimated. Bulk e-mail lists are often used to entice recipients to click on a link that may lead to a Web site or e-mail address quite different from that promised. A long-enough URI with a mixture of "real" @ signs and Koalib @-letters would confuse even careful readers, and would be a boon to spammers, some of whom might concentrate their efforts on comparatively inexperienced Internet users in less-developed regions of Africa whose fonts support these characters. Again, the Koalib @-letters could become a burden and a security risk even to the users they are intended to benefit.

The recent successful efforts to establish standards for internationalized domain names, expanding beyond the unaccented Latin alphabet, mean that the potential for user confusion is even greater when a character as fundamental to Internet naming conventions as @ is duplicated as an orthographic letter.

At least one contributor has suggested that the proposed lowercase letter could be rendered differently from the standard @ sign, in hopes of reducing the confusion. But "recommended" glyphs are not normative, and the greater likelihood is that type designers would create an @-letter that matches the overall design of the alphabet—just as they do now with the existing @ sign. Furthermore, in the examples accompanying the proposal, the lowercase @-letter is rendered using the simpler, one-loop "ɑ" shape even as the rest of the text uses a Times-like face with two-loop "a's". There is clearly no attempt here to make the Koalib @-letter look like anything other than the familiar @ sign. (Even the proposed names, "LATIN CAPITAL (and SMALL) LETTER AT," show the true identity of these "letters" as overloaded symbols.)

The uppercase @-letter might not cause the same degree of trouble as its lowercase counterpart, because it is not an exact visual duplicate of any existing symbol. Encoding this letter may not be as problematic as encoding the lowercase version. There is still a real possibility that users will confuse the uppercase letter for COMMERCIAL AT, especially in addresses that are otherwise all-capitals (*i.e.* "SOMEBODY@SOMEWHERE.COM"). This type of capitalization is especially common among users of on-line services like AOL, who are often not experienced enough to recognize "slightly different" characters and might regard an uppercase Koalib @-letter as a trendy glyph variant of U+0040.

Not only would encoding the Koalib @-letters cause genuine security problems, but perhaps equally importantly, it would undercut the credibility and reputation of UTC and WG2 for being sensitive to security problems. A section on "confusable" characters was added to *The Unicode Standard, Version 4.0*, acknowledging the existence of such characters and explaining the problems they can cause, even in legacy character sets. The Koalib @-letters are perhaps the most egregious example ever of gratuitous

confusability; adding them to the Universal Character Set would send security experts scrambling to their keyboards to proclaim that UTC and WG2 "just don't get it" with respect to security. The actual effect on the acceptance of Unicode in security-sensitive applications would be hard to predict.

The literate Koalib-speaking community can be served adequately by continuing to encode the lowercase @-letter using U+0040, as was undoubtedly done in producing the two religious works in Kenya during the 1990s, and most likely in other printed and typewritten works. (Koalib apparently uses the @-letters only for words borrowed from Arabic; see http://www.language-museum.com/k/koalib.htm for another sample of religious text in Koalib that contains no @-letters.) Alternatively, the existing Unicode characters U+24B6 Ⓐ and U+24D0 ⓐ could be used to write Koalib, as suggested on the public Unicode mailing list.

It is true that the character properties of these alternative characters are not consistent with other orthographic letters, but this affects only a small set of text-processing operations, such as word-breaking and spell-checking, which are unlikely to be implemented widely for Koalib *even if the proposed @-letters are approved.* The use of the @-letter in identifiers in markup languages such as XML would also be prohibited. A restriction like this may be a small price to pay to prevent a "letter" that looks exactly like the @ sign from appearing in XML identifiers.

Providing the characters needed by computer users worldwide, including speakers of minority languages, has always been a fundamental goal of the Universal Character Set. However, the disadvantages of encoding the Koalib @-letters, in terms of confusion and security risk, far outweigh their advantages. I strongly encourage the Unicode Technical Committee and WG2 to reject the proposal to encode the Koalib @-letters in the Universal Character Set.