



Forum: Modules

Topic: UNICODE : trouver une solution

Subject: Re: UNICODE : trouver une solution

Posté par: Chan

Contribution le : 14/10/2005 20:13:03

Le problème de l'UNICODE n'est pas simplement un problème d'affichage au niveau de WinDev. L'unicode est codé sur 2 octets, l'ansi sur 1 octet. Apparemment, les opérations sur chaîne dans windev, travaillent sur 1 octet.

Si on veut afficher par exemple du français et du chinois, on est obligé de coder en 2 octets. En unicode, il n'y a pas d'alphabet d'origine, les 2 octets peuvent coder tous les alphabets de la planète sans changer de code page.

Dans ses chaînes de caractère, windev ne sait pas travailler sur 2 octets. Même avec les fonctions ChangeAlphabet, les fonctions de recherche qui sont basées sur l'octet ne fonctionnent pas correctement.

Par exemple: si on passe en alphabet chinois simplifié (code GB), les idéogrammes sont alors codés sur 2 octets, les caractères latins sur 1 octet.

Supposons que l'on aie une chaîne de deux caractères chinois, soit par exemple en hexa: A3B67EF4 et que l'on cherche par la fonction "contient" par requête ou tri, le caractère 3B67 ou B67E ou 67EF et bien Windev dira que la chaîne initiale contient ces caractères alors que c'est faux. Toute la gestion des chaînes de caractère de plus d'un octet est à revoir et à forciore celle de l'unicode.

La gestion de l'unicode n'est pas un problème anodin.

La Chine vient de réaliser un standard, le GB18030 qui correspond par une table code à code au standard unicode, et leur intention est de ne pas autoriser à terme sur leur territoire les logiciels non conforme à ce standard. Le marché chinois, ce n'est pas rien.

Mais le reste du monde fait de même, l'unicode va être le standard du codage des chaînes de caractère.

Unicode est le codage sur WindowsCE, je n'ai pas encore testé comment se comporte Windev Mobile émulé sur WindowsXP.

Je ne suis pas chinois, mais travaille avec des chinois et ce problème je pense est à nos portes. Tous les logiciels pouvant afficher du chinois sur lesquels j'ai travaillé gère correctement l'unicode, même le bloc note de windows le fait !