Subject: Re: 16 bits wchar
Posted by cbpporter on Thu, 04 Oct 2007 11:15:09 GMT
View Forum Message <> Reply to Message

OK, we should leave than Stream the way you intended. It serves it's purpose well without extra buffers and I don't want 20 variants of Stream and assorted with different kinds of buffers (like in Java).

So I am going to concentrate on CharSet and String. I created a function to check if an UTF-8 sequence is correct or not. I know that you have such a function (I even reused most of it), but we use different versions of Unicode. Mine is compliant (or will be) with changes after November 2003, while yours is older.

I tested it a little and going to try to find some test data so I can fully debug it, but it looks something like this:

```
bool utf8check5(const char *_s, int len)
{
 const byte *s = (const byte *)_s;
 const byte *lim = s + len;
 int codePoint = 0;
 while(s < lim) {
  word code = (byte)*s++;
  if(code >= 0x80) {
   if(code < 0xC2)
    return false;
   else
   if(code < 0xE0) {
    if(s >= lim || *s < 0x80 || *s >= 0xc0)
     return false;
    codePoint = ((code - 0xC0) << 6) + *s - 0x80;
    if(codePoint < 0x80 || codePoint > 0x07FF)
     return false;
    s++;
   }
   else
   if(code < 0xF0) {
    if(s + 1 >= lim ||
       s[0] < 0x80 || s[0] >= 0xc0 ||
       s[1] < 0x80 || s[1] >= 0xc0)
        return false;
    codePoint = ((code - 0xE0) << 12) + ((s[0] - 0x80) << 6) + s[1] - 0x80;
    if(codePoint < 0x0800 || codePoint > 0xFFFF)
     return false;
    s += 2;
   }
   else
   if(code < 0xF5) {
```

```
    if(s + 2 >= lim ||
       s[0] < 0x80 || s[0] >= 0xc0 ||
       s[1] < 0x80 || s[1] >= 0xc0 ||
       s[2] < 0x80 || s[2] >= 0xc0)
        return false;
    codePoint = ((code - 0xf0) << 18) + ((s[0] - 0x80) << 12) + ((s[1] - 0x80) << 6) + s[2] - 0x80;
    if(codePoint < 0x010000 || codePoint > 0x10FFFF)
     return false;
    s += 3;
   }
   else
    return false;
  }
 }
 return true;
}
```