Subject: Funny way how to speed string equality comparison
Posted by mirek on Fri, 05 Jan 2007 11:30:15 GMT
View Forum Message <> Reply to Message

```
#ifdef STRING_EXPERIMENTAL
#define EQ1(i)  (a[i] ^ b[i])
#define EQ2(i)  (*(word *)(a + i) ^ *(word *)(b + i))
#define EQ4(i)  (*(dword *)(a + i) ^ *(dword *)(b + i))
#endif

bool String::IsEqual(const String& s) const
{
#ifdef STRING_EXPERIMENTAL
 int l = GetLength();
 if(s.GetLength() != GetLength()) return false;
 const char *a = ptr;
 const char *b = s;
 switch(l) {
 case  0: return true;
 case  1: return a[0] == b[0];
 case  2: return *(word *)a == *(word *)b;
 case  3: return (EQ2(0) | EQ1(2)) == 0;
 case  4: return *(dword *)a == *(dword *)b;
 case  5: return (EQ4(0) | EQ1(4)) == 0;
 case  6: return (EQ4(0) | EQ2(4)) == 0;
 case  7: return (EQ4(0) | EQ2(4) | EQ1(7)) == 0;
 case  8: return (EQ4(0) | EQ4(4)) == 0;
 case  9: return (EQ4(0) | EQ4(4) | EQ1(8)) == 0;
 case 10: return (EQ4(0) | EQ4(4) | EQ2(8) | EQ1(10)) == 0;
 case 11: return (EQ4(0) | EQ4(4) | EQ2(8) | EQ1(10)) == 0;
 case 12: return (EQ4(0) | EQ4(4) | EQ4(8)) == 0;
 case 13: return (EQ4(0) | EQ4(4) | EQ4(8) | EQ1(12)) == 0;
 case 14: return (EQ4(0) | EQ4(4) | EQ4(8) | EQ2(12)) == 0;
 }
#endif
 return B::IsEqual(s, s.GetCount());
}
```

Speedsup map oriented benchmarks by 10%....

Mirek