
Subject: Re: Pop3 class and reference examples for U++

Posted by [Oblivion](#) on Wed, 16 Apr 2014 01:00:11 GMT

[View Forum Message](#) <> [Reply to Message](#)

Hello Mirek,

I see that you recently added ParseMessageIDs() method to the InetMessage class.

The code block in that function is something I encounter often. Especially when parsing some text (you can even see it in the Pop3::GetTimestamp()).

Maybe we should put that code under String utilities and change the code accordingly?

So I wrote the following Slice() functions which you can also come by in some other C++/Java frameworks.

```
int Slice(const String& s, String& d, const String& delim1, const String& delim2, int pos = 0)
{
    int b = -1, e = -1;
    if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, b += delim1.GetLength())) == -1)
        return -1;
    d = s.Mid(b, e - b);
    return e += delim2.GetLength();
}
```

```
int Slice(const WString& s, WString& d, const WString& delim1, const WString& delim2, int pos = 0)
{
    int b = -1, e = -1;
    if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, b += delim1.GetLength())) == -1)
        return -1;
    d = s.Mid(b, e - b);
    return e += delim2.GetLength();
}
```

```
int Slice(const String& s, String& d, int delim1, int delim2, int pos = 0)
{
    int b = -1, e = -1;
    if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, ++b)) == -1)
        return -1;
    d = s.Mid(b, e - b);
    return ++e;
}
```

```
int Slice(const WString& s, WString& d, int delim1, int delim2, int pos = 0)
{
    int b = -1, e = -1;
    if((b = s.Find(delim1, pos)) == -1 || (e = s.Find(delim2, ++b)) == -1)
        return -1;
    d = s.Mid(b, e - b);
}
```

```
    return ++e;
}
```

```
String Slice(const String& s, const String& delim1, const String& delim2)
{
    String r;
    return (Slice(s, r, delim1, delim2) == -1) ? String::GetVoid() : r;
}
```

```
WString Slice(const WString& s, const WString& delim1, const WString& delim2)
{
    WString r;
    return (Slice(s, r, delim1, delim2) == -1) ? WString::GetVoid() : r;
}
```

```
String Slice(const String& s, int delim1, int delim2)
{
    String r;
    return (Slice(s, r, delim1, delim2) == -1) ? String::GetVoid() : r;
}
```

```
WString Slice(const WString& s, int delim1, int delim2)
{
    WString r;
    return (Slice(s, r, delim1, delim2) == -1) ? WString::GetVoid() : r;
}
```

They are really useful for parsing text. I already patched my local copy of upp (with its api reference doc in "String utility functions" section).

By the way, I can upload the patched files if you have time to review them and also approve them.

Regards.
