Subject: Re: Hello, need a helping hand
Posted by cbpporter on Fri, 16 Jul 2010 10:18:54 GMT
View Forum Message <> Reply to Message

jerson wrote on Thu, 15 July 2010 17:46

BTW : what do these statements mean? It's a bit like greek to me.

```
void UWord::SerializeApp(Stream& s)
{
 int version = 1;
 s / version;              <<== this one here
 s % UWordFs()             <<== and here
   % PdfFs();              <<== and here
 if(version >= 1)
  s % lrufile();
}
```

Hi!

Yes, serialization can seem a little bit confusing at first, but this is one my favorite features. It is very easy to serialize large sets of data.

The serialization API uses operator overloading very heavily and "%" is something like "<<" combined with ">>" from standard C++ API (like you use with cout). It is the standard serialization operator. So if you see:
s % foo % bar;
think of it as:
s.serialize(foo).serialize(bar);
This hypothetical serialize method uses chaining and reference parameters, and in pseudo code it has an implementation like this for every basic type:

```
Stream& serialize(Foo& foo) {
  if (stream is loading)
    foo.loadFromStream
  else
    foo.saveToStream
  return stream for chaining;
}
```

This is the reason you don't need to write a separate load and a separate sore serialization function. After you have become more familiar with the syntax, you can look at the implementation if you are curious and you will see that adding support for serializing your own classes is very easy.