

## Re: Writing binary data in managed C++

**Source:**

<http://www.tech-archive.net/Archive/DotNet/microsoft.public.dotnet.languages.vc/2004-10/0036.html>

---

**From:** Peter Steele (*psteele\_at\_z-force.com*)

**Date:** 10/01/04

Date: Thu, 30 Sep 2004 19:25:51 -0700

John, here's what I came up with. Assuming we have a simple struct as follows:

```
struct  
  
{  
  
    ...  
  
} X;
```

whose members have been initialized in some manner, then the following code can be used to dump the bytes of this structure:

```
FileStream* fs = new FileStream("C:\\data.bin", FileMode::Create,  
FileAccess::Write);
```

```
BinaryWriter* bw = new BinaryWriter(fs);
```

```
char* bytes = (char*)&X;
```

```
char gbytes __gc[] = new char __gc[sizeof X];
```

```
for (int i = 0; i < sizeof X; i++) gbytes[i] = bytes[i];
```

```
bw->Write(gbytes);
```

```
bw->Flush();
```

```
bw->Close();
```

What I don't like about this solution is that it duplicates the entire structure. Can the data be output directly without this duplication?

Peter

"Peter Steele" <psteele@z-force.com> wrote in message  
news:OvgAPS0pEHA.1688@TK2MSFTNGP10.phx.gbl...

> *I have a managed C++ project that contains a simple structure:*

>

> *struct*

> {

> *<multiple members>*

> } *mydata;*

>

> *I want to write the contents of this structure out to disk retaining the*

> *exactly binary data without any additional header information, and of*

> *course be able to later read it back in. What's the best way to accomplish*

> *this?*

>

>