

---

Subject: compressed serialization

Posted by [nixnixnix](#) on Wed, 09 Sep 2009 00:56:34 GMT

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

---

Is it possible to combine Z compression (or any other kind of compression) with the built in serialization functions such as LoadFromFile and StoreToFile? If so, is there an example of this please?

If its not possible to do that, is it possible to write and read compressed serialized files without writing the uncompressed version to disk and with uncompressed file sizes in the hundreds of megabytes or more?

Nick

---

---

Subject: Re: compressed serialization

Posted by [Novo](#) on Wed, 09 Sep 2009 03:59:19 GMT

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

---

IMO, that is not possible.

LoadFromFile/StoreToFile just open a FileStream and call Load/Store using this stream. In order to use compressed serialized files you need to have compression/decompression algorithms, which behave as streams, and bind compression streams with regular streams in a filter-like way. IMO that is not possible right now.

---

---

Subject: Re: compressed serialization

Posted by [Mindtraveller](#) on Wed, 09 Sep 2009 06:41:27 GMT

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

---

Maybe it is possible to unzip file into memory, then load from it as Stream?

---

---

Subject: Re: compressed serialization

Posted by [Novo](#) on Wed, 09 Sep 2009 19:14:29 GMT

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

---

Mindtraveller wrote on Wed, 09 September 2009 02:41 Maybe it is possible to unzip file into memory, then load from it as Stream?

That is definitely possible You just need to unzip one stream into another. Who cares about memory these days ...

---

---

Subject: Re: compressed serialization

Posted by [Mindtraveller](#) on Thu, 10 Sep 2009 01:01:48 GMT

---

You may create some proxy class like MyClassZippedSerializer which will compress/decompress your inner MyClass serialized data then save it to file. Using this technique you may even partially read/write your serialized data to make memory consumption close to theoretical minimum.

---