

## Ruby trunk - Feature #13020

### Zlib.gzip and Zlib.gunzip

12/09/2016 10:37 PM - naruse (Yui NARUSE)

<b>Status:</b>	Closed	
<b>Priority:</b>	Normal	
<b>Assignee:</b>		
<b>Target version:</b>		
<b>Description</b>		
I added Zlib.deflate/inflate [Feature <a href="#">#4180</a> ] before, but writing/reading gzip is still too complex. It should have shorthand method.		
<b>Related issues:</b>		
Related to Ruby trunk - Bug #13021: `Zlib.gunzip` modifies argument String		<b>Closed</b>

#### Associated revisions

##### Revision 583b6dc3 - 12/09/2016 10:45 PM - naruse (Yui NARUSE)

Zlib.gzip and Zlib.gunzip [Feature #13020]

Encode and Decode gzip data without creating files.

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@57035 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

##### Revision 57035 - 12/09/2016 10:45 PM - naruse (Yui NARUSE)

Zlib.gzip and Zlib.gunzip [Feature #13020]

Encode and Decode gzip data without creating files.

##### Revision 57035 - 12/09/2016 10:45 PM - naruse (Yui NARUSE)

Zlib.gzip and Zlib.gunzip [Feature #13020]

Encode and Decode gzip data without creating files.

##### Revision 57035 - 12/09/2016 10:45 PM - naruse (Yui NARUSE)

Zlib.gzip and Zlib.gunzip [Feature #13020]

Encode and Decode gzip data without creating files.

##### Revision 57035 - 12/09/2016 10:45 PM - naruse (Yui NARUSE)

Zlib.gzip and Zlib.gunzip [Feature #13020]

Encode and Decode gzip data without creating files.

##### Revision ba568c0e - 12/20/2016 04:02 PM - naruse (Yui NARUSE)

Zlib.gzip uses kwargs instead of argc [Feature #13020]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@57126 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

##### Revision 57126 - 12/20/2016 04:02 PM - naruse (Yui NARUSE)

Zlib.gzip uses kwargs instead of argc [Feature #13020]

##### Revision 57126 - 12/20/2016 04:02 PM - naruse (Yui NARUSE)

Zlib.gzip uses kwargs instead of argc [Feature #13020]

##### Revision 57126 - 12/20/2016 04:02 PM - naruse (Yui NARUSE)

Zlib.gzip uses kwargs instead of argc [Feature #13020]

##### Revision 57126 - 12/20/2016 04:02 PM - naruse (Yui NARUSE)

Zlib.gzip uses kwargs instead of argc [Feature #13020]

**Revision 0adce993 - 12/20/2016 05:15 PM - nobu (Nobuyoshi Nakada)**

rdoc for r57126 [ci skip]

- ext/zlib/zlib.c: [DOC] update as keyword arguments. [Feature #13020]

git-svn-id: svn+ssh://ci.ruby-lang.org/ruby/trunk@57129 b2dd03c8-39d4-4d8f-98ff-823fe69b080e

**Revision 57129 - 12/20/2016 05:15 PM - nobu (Nobuyoshi Nakada)**

rdoc for r57126 [ci skip]

- ext/zlib/zlib.c: [DOC] update as keyword arguments. [Feature #13020]

**Revision 57129 - 12/20/2016 05:15 PM - nobu (Nobuyoshi Nakada)**

rdoc for r57126 [ci skip]

- ext/zlib/zlib.c: [DOC] update as keyword arguments. [Feature #13020]

**Revision 57129 - 12/20/2016 05:15 PM - nobu (Nobuyoshi Nakada)**

rdoc for r57126 [ci skip]

- ext/zlib/zlib.c: [DOC] update as keyword arguments. [Feature #13020]

**Revision 57129 - 12/20/2016 05:15 PM - nobu (Nobuyoshi Nakada)**

rdoc for r57126 [ci skip]

- ext/zlib/zlib.c: [DOC] update as keyword arguments. [Feature #13020]

**History**

---

**#1 - 12/09/2016 10:45 PM - naruse (Yui NARUSE)**

- Status changed from Open to Closed

Applied in changeset r57035.

---

Zlib.gzip and Zlib.gunzip [Feature [#13020](#)]

Encode and Decode gzip data without creating files.

**#2 - 12/10/2016 12:21 AM - normalperson (Eric Wong)**

[naruse@airemix.jp](mailto:naruse@airemix.jp) wrote:

I added Zlib.deflate/inflate [Feature [#4180](#)] before, but writing/reading gzip is still too complex. It should have shorthand method.

I like the convenience, but I think encouraging use of potentially large String buffers is dangerous.

It causes memory problems and even opens up DoS attacks.

So, I think it would be better if it used IO-like object instead, to discourage slurping.

Maybe make it like IO.copy\_stream:

```
Zlib.gzip(src_io, dst_io)
Zlib.gunzip(src_io, dst_io)
```

Slurpers can still use StringIO if they want to.

In general, I am against non-streaming APIs and want to encourage processing data in predictable sizes to avoid fragmentation and OOM.

**#3 - 12/10/2016 02:00 AM - znz (Kazuhiro NISHIYAMA)**

- Related to Bug #13021: `Zlib.gunzip` modifies argument String added

#4 - 12/19/2016 06:02 AM - naruse (Yui NARUSE)

Eric Wong wrote:

[naruse@airemix.jp](mailto:naruse@airemix.jp) wrote:

I added Zlib.deflate/inflate [Feature #4180] before, but writing/reading gzip is still too complex. It should have shorthand method.

I like the convenience, but I think encouraging use of potentially large String buffers is dangerous.

It causes memory problems and even opens up DoS attacks.

So, I think it would be better if it used IO-like object instead, to discourage slurping.

Maybe make it like IO.copy\_stream:

```
Zlib.gzip(src_io, dst_io)
Zlib.gunzip(src_io, dst_io)
```

Slurpers can still use StringIO if they want to.

In general, I am against non-streaming APIs and want to encourage processing data in predictable sizes to avoid fragmentation and OOM.

There's already Zlib.deflate/inflate, which are memory based API.

And gzip also uses deflate/inflate.

Note that gzip has the length of original data at the end of data. (isize=origsize / (2<sup>32</sup>))

Anyway IO.copy\_stream like API sounds reasonable.

Current API needs to change interface to use keyword arguments to avoid using 2nd argument...