Ruby - Bug #19632

Disable external iterator for frozen enumerator

05/09/2023 01:22 AM - make_now_just (Hiroya Fujinami)

Status:	Closed		
Priority:	Normal		
Assignee:			
Target version:			
ruby -v:		**Backport:**	3.0: UNKNOWN, 3.1: UNKNOWN, 3.2: UNKNOWN

Description

Currently, methods to manipulate an external iterator like #next and #feed can be called even if a receiver of an enumerator is frozen. However, these methods change the state of an external iterator in an enumerator. Therefore, it seems a BUG to me, and these methods should raise FrozenError if the receiver is frozen.

```
e = 3.times.freeze

# Current

e.next # => 1
e.next # => 2

# Expected

e.next # raise FrozenError
```

Two years ago, this issue was mentioned in a comment.

I suggest fixing the following methods to raise FrozenError against a frozen enumerator.

- Enumerator#next
- Enumerator#next_values
- Enumerator#peek
- Enumerator#peek_values
- Enumerator#feed
- Enumerator#rewind

Also, even if an enumerator is frozen, it does not affect other methods for internal iterators.

Associated revisions

Revision 3e64cf60b5162bb5dad772f300b7f6346e5f83f9 - 10/25/2023 07:32 AM - make_now_just (Hiroya Fujinami)

Fix [Bug #19632]: Disable external iterator for frozen enumerator (#7791)

- Fix [Bug #19632]: Disable external iterator for frozen enumerator

Currently, methods to manipulate an external iterator like #next and #feed can be called even if a receiver of an enumerator is frozen. However, these methods change the state of an external iterator in an enumerator. Therefore, it seems a BUG to me, and these methods should raise FrozenError if the receiver is frozen.

This fixed the following methods to raise FrozenError if the receiver is frozen.

- Enumerator#next
- Enumerator#next_values
- Enumerator#peek
- Enumerator#peek_values
- Enumerator#feed
- Enumerator#rewind

- Fix a typo in the document

Thanks @Maumagnaguagno.

Revision 3e64cf60b5162bb5dad772f300b7f6346e5f83f9 - 10/25/2023 07:32 AM - make_now_just (Hiroya Fujinami)

Fix [Bug #19632]: Disable external iterator for frozen enumerator (#7791)

- Fix [Bug #19632]: Disable external iterator for frozen enumerator

Currently, methods to manipulate an external iterator like #next
and #feed can be called even if a receiver of an enumerator is
frozen. However, these methods change the state of an external
iterator in an enumerator. Therefore, it seems a BUG to me, and
these methods should raise FrozenError if the receiver is frozen.

This fixed the following methods to raise FrozenError if the receiver is
frozen.

- Enumerator#next
- Enumerator#next_values
- Enumerator#peek
- Enumerator#peek_values
- Enumerator#feed
- Enumerator#rewind

- Fix a typo in the document

Thanks @Maumagnaguagno.

Revision 3e64cf60 - 10/25/2023 07:32 AM - make_now_just (Hiroya Fujinami)

Fix [Bug #19632]: Disable external iterator for frozen enumerator (#7791)

- Fix [Bug #19632]: Disable external iterator for frozen enumerator

Currently, methods to manipulate an external iterator like #next
and #feed can be called even if a receiver of an enumerator is
frozen. However, these methods change the state of an external
iterator in an enumerator. Therefore, it seems a BUG to me, and
these methods should raise FrozenError if the receiver is frozen.

This fixed the following methods to raise FrozenError if the receiver is
frozen.

- Enumerator#next
- Enumerator#next_values
- Enumerator#peek
- Enumerator#peek_values
- Enumerator#feed
- Enumerator#rewind

- Fix a typo in the document

Thanks @Maumagnaguagno.

History

#1 - 05/09/2023 01:26 AM - make_now_just (Hiroya Fujinami)

A Pull Request for this is created. https://github.com/ruby/ruby/pull/7791

#2 - 05/09/2023 01:41 AM - nobu (Nobuyoshi Nakada)

Sounds reasonable.

#3 - 05/13/2023 02:14 PM - matz (Yukihiro Matsumoto)

peek does not seem to modify the enumerator? Should we prohibit it (or not)?

#4 - 05/14/2023 12:41 AM - make_now_just (Hiroya Fujinami)

If a result value is not stored, peek invokes the iterator and stores it in its internal state. For that reason, I consider peek is also a mutable method like next.

If you call peek in advance and then call freeze, peek will not change the internal state. IMHO, this case is exceptional and should still raise a FrozenError for simplicity. However, I would follow matz's decision.

#5 - 05/14/2023 10:53 AM - Eregon (Benoit Daloze)

+1. And yes peek can have side effect so should be considered mutating method.

#6 - 10/25/2023 07:32 AM - make_now_just (Hiroya Fujinami)

- Status changed from Open to Closed

Applied in changeset git|3e64cf60b5162bb5dad772f300b7f6346e5f83f9.

Fix [Bug #19632]: Disable external iterator for frozen enumerator (#7791)

- Fix [Bug #19632]: Disable external iterator for frozen enumerator

Currently, methods to manipulate an external iterator like #next
and #feed can be called even if a receiver of an enumerator is
frozen. However, these methods change the state of an external
iterator in an enumerator. Therefore, it seems a BUG to me, and
these methods should raise FrozenError if the receiver is frozen.

This fixed the following methods to raise FrozenError if the receiver is
frozen.

- Enumerator#next
- Enumerator#next_values
- Enumerator#peek
- Enumerator#peek_values
- Enumerator#feed
- Enumerator#rewind

- Fix a typo in the document

Thanks @Maumagnaguagno.