



Compressor Documentation

Release 3.1

/ELSA/MU-14010/V3.1

May 28, 2020

CONTENTS

1	Preamble	1
2	List of functions	3
3	Contents	5
3.1	Index field compression	5
3.2	Object serialize/compression	6
4	Index	7

PREAMBLE

Compressor enables fields compression for arrays/pyTrees.

This module is part of Cassiopee, a free open-source pre- and post-processor for CFD simulations.

To use the module with the Compressor array interface:

```
import Compressor
```

To use the module with the CGNS/Python interface:

```
import Compressor.PyTree as Compressor
```


LIST OF FUNCTIONS

– Index field compression

<code>Compressor.deltaIndex(index, ref)</code>	Return the delta between index and ref.
--	---

– Object serializer/compression

<code>Compressor.pack(a[, method])</code>	Serialize or compress a.
<code>Compressor.unpack(a[, method])</code>	Deserialize or decompress a.

3.1 Index field compression

Compressor.**deltaIndex**(*a*, *ref*)

Compress a list of indices using delta algorithm. The return Delta contains the number of added indices in *a* when compared to *ref*, the list of added indices, the number of suppressed indices, the list of suppressed indices.

Parameters

- **a** (numpy of ints) – input indices
- **ref** (numpy) – compared indices

Returns list of added indices, the number of suppressed indices, list of suppressed indices

Return type (numpy, int, numpy)

- Compression by delta (numpy):

```
# - deltaIndex -
import numpy
import Compressor

# Liste des indexes de reference
indRef = numpy.array([1,2,3,4,5], dtype='int32')

# Liste des indexes a comparer a la reference
index = numpy.array([1,2,3,4], dtype='int32')

delta = Compressor.deltaIndex(index, indRef)
print(delta)
```

3.2 Object serialize/compression

Compressor.`pack`(a)

Serialize/compress a python object a. For now, this is only a general interface to pickle module.

Parameters a (python object) – any python object

Returns serialized stream

- Object serialization (numpy):

```
# - pack -
import Compressor
import Generator.PyTree as G
a = G.cart((0,0,0), (1,1,1), (1000,100,100))
b = Compressor.pack(a)
```

Compressor.`unpack`(a)

Deserialize/decompress a serialized stream b. For now, this is only a general interface to pickle module.

Parameters a (serialized stream) – a serialized stream as produced by pack

Returns python object

- Object deserialization (numpy):

```
# - unpack -
import Compressor
import Generator.PyTree as G
a = G.cart((0,0,0), (1,1,1), (1000,100,100))
b = Compressor.pack(a)
c = Compressor.unpack(b)
```

CHAPTER
FOUR

INDEX

- genindex
- modindex
- search