

## Ruby master - Feature #17045

### ObjectSpace.dump\_all should allocate as little as possible in the GC heap

07/23/2020 08:52 AM - byroot (Jean Boussier)

<b>Status:</b>	Closed
<b>Priority:</b>	Normal
<b>Assignee:</b>	
<b>Target version:</b>	
<b>Description</b>	
For context I'm working on a <a href="#">heap profiler</a> . In short the use case is like this (pseudo code):	
<pre>GC.disable ObjectSpace.dump_all(output: file1) # run the user code ObjectSpace.dump_all(output: file2) compute_the_diff_and_report_statistics(file1, file2)</pre>	
Ideally I would need ObjectSpace.dump_all to not modify the GC heap at all so that an empty user code would report an empty diff.	
However as showcased <a href="#">in this test case</a> , dump_all(output: <#File:/path.json>) currently allocates 4 objects:	
<ul style="list-style-type: none"><li>• The File instance passed as output: is re-opened by rb_io_stdio_file in dump_output and I don't quite understand why.</li><li>• The scan_args in dump_all allocates a Hash instance. Would using the new Primitive interface avoid that?</li><li>• Another hash is allocated but I'm unsure where it comes from.</li><li>• An IMEMO "imemo_type"=&gt;"callcache" is allocated. Surprisingly it can be avoided by calling ObjectSpace.dump_all(**opts)</li></ul>	
Could any of these be eliminated?	

#### Associated revisions

##### Revision fbba6bd4 - 09/15/2020 04:18 PM - byroot (Jean Boussier)

Parse ObjectSpace.dump\_all / dump arguments in Ruby to avoid allocation noise

[Feature #17045] ObjectSpace.dump\_all should allocate as little as possible in the GC heap

Up until this commit ObjectSpace.dump\_all allocates two Hash because of rb\_scan\_args.

It also can allocate a File because of rb\_io\_get\_write\_io.

These allocations are problematic because dump\_all dumps the Ruby heap, so it should try modify as little as possible what it is observing.

#### History

##### #1 - 07/24/2020 08:19 PM - jeremyevans0 (Jeremy Evans)

- Backport deleted (2.5: UNKNOWN, 2.6: UNKNOWN, 2.7: UNKNOWN)

- Tracker changed from Bug to Feature

##### #2 - 09/10/2020 12:34 PM - byroot (Jean Boussier)

I opened a PR with a patch for this: <https://github.com/ruby/ruby/pull/3530>

##### #3 - 09/15/2020 04:18 PM - byroot (Jean Boussier)

- Status changed from Open to Closed

Applied in changeset [git|fbba6bd4e3dff7a61965208fecae908f10c4edbe](https://github.com/ruby/ruby/commit/fbba6bd4e3dff7a61965208fecae908f10c4edbe).

Parse ObjectSpace.dump\_all / dump arguments in Ruby to avoid allocation noise

[Feature [#17045](#)] ObjectSpace.dump\_all should allocate as little as possible in the GC heap

Up until this commit ObjectSpace.dump\_all allocates two Hash because of rb\_scan\_args.

It also can allocate a File because of `rb_io_get_write_io`.

These allocations are problematic because `dump_all` dumps the Ruby heap, so it should try modify as little as possible what it is observing.