

Ruby - Feature #20306

Add rb_free_at_exit_p

02/26/2024 08:12 PM - peterzhu2118 (Peter Zhu)

Status:	Closed	
Priority:	Normal	
Assignee:	peterzhu2118 (Peter Zhu)	
Target version:		
Description		
GitHub PR: https://github.com/ruby/ruby/pull/10104		
From ticket #20290, I found that C extensions could use ruby_vm_at_exit to register hooks to free memory at shutdown. However, they cannot determine whether they should free all memory during shutdown to mirror the behavior of Ruby when RUBY_FREE_AT_EXIT is set.		
This ticket proposes a new API called rb_free_at_exit_p that returns true when RUBY_FREE_AT_EXIT is set, and false otherwise.		

Associated revisions

Revision 83618f2cfa004accdd1514de7dcbba291aa7e831 - 03/14/2024 12:33 PM - peterzhu2118 (Peter Zhu)

[Feature #20306] Implement ruby_free_at_exit_p

ruby_free_at_exit_p is a way for extensions to determine whether they should free all memory at shutdown.

Revision 83618f2cfa004accdd1514de7dcbba291aa7e831 - 03/14/2024 12:33 PM - peterzhu2118 (Peter Zhu)

[Feature #20306] Implement ruby_free_at_exit_p

ruby_free_at_exit_p is a way for extensions to determine whether they should free all memory at shutdown.

Revision 83618f2c - 03/14/2024 12:33 PM - peterzhu2118 (Peter Zhu)

[Feature #20306] Implement ruby_free_at_exit_p

ruby_free_at_exit_p is a way for extensions to determine whether they should free all memory at shutdown.

History

#1 - 03/06/2024 02:19 AM - kjtsanaktisidis (KJ Tsanaktisidis)

This might be one of those "everything wants to run after everything else" situations, but...

ruby_vm_run_at_exit_hooks is called before rb_objspace_free_objects during shutdown; that means extension dfree methods might be called after the atexit hooks they registered. My sense is that normally extensions will want that the other way around? If a C library offers a kind of global shutdown function, it's normally invalid to call any of the library's functions after that. E.g. for libxml which was mentioned in the previous ticket: <https://gnome.pages.gitlab.gnome.org/libxml2/devhelp/libxml2-parser.html>

xmlCleanupParser ()

void xmlCleanupParser (void)

This function name is somewhat misleading. It does not clean up parser state, it cleans up memory allocated by the library itself. It is a cleanup function for the XML library. It tries to reclaim all related global memory allocated for the library processing. It doesn't deallocate any document related memory. One should call xmlCleanupParser() only when the process has finished using the library and all XML/HTML documents built with it. See also xmlInitParser() which has the opposite function of preparing the library for operations. WARNING: if your application is multithreaded or has plugin support calling this may crash the application if another thread or a plugin is still using libxml2. It's sometimes very hard to guess if libxml2 is in use in the application, some libraries or plugins may use it without notice. In case of doubt abstain from calling this function or do it just before calling exit() to avoid leak reports from valgrind !

On the other hand, a C extension might want to do something with its Ruby objects before they get freed by rb_objspace_free_objects.....

Do we actually need two sets of atexit hooks here?

#2 - 03/06/2024 04:27 PM - peterzhu2118 (Peter Zhu)

ruby_vm_run_at_exit_hooks is called before rb_objspace_free_objects during shutdown

rb_objspace_free_objects should NOT be freeing objects from native extensions, so this is not a problem. Objects from native extensions should have been freed earlier in rb_objspace_call_finalizer. rb_objspace_free_objects is only there to free objects that cannot be freed until the very end (e.g. thread, mutex, main Ractor).

#3 - 03/06/2024 11:25 PM - kjtsanaktsidis (KJ Tsanaktsidis)

Oh I didn't realise that - leveraging ruby_vm_at_exit makes complete sense then.

Big +1 from me on this proposal!

#4 - 03/14/2024 09:43 AM - mame (Yusuke Endoh)

- Status changed from Open to Assigned
- Assignee set to peterzhu2118 (Peter Zhu)

Discussed at the dev meeting, and [@matz \(Yukihiro Matsumoto\)](#) said "go ahead".

#5 - 03/14/2024 12:33 PM - peterzhu2118 (Peter Zhu)

- Status changed from Assigned to Closed

Applied in changeset [git|83618f2cfa004accdd1514de7dcbba291aa7e831](#).

[Feature [#20306](#)] Implement ruby_free_at_exit_p

ruby_free_at_exit_p is a way for extensions to determine whether they should free all memory at shutdown.