

Ruby master - Feature #3112

require "yaml" doesn't use psych as default

04/08/2010 11:14 AM - usa (Usaku NAKAMURA)

Status:	Closed	
Priority:	Normal	
Assignee:		
Target version:	2.0.0	
Description =begin why? =end		
Related issues: Related to Ruby master - Feature #2340: Removing YAML/Syck Rejected 11/06/2009		

History

#1 - 04/08/2010 02:06 PM - naruse (Yui NARUSE)

=begin

= Before

require 'yaml' and you can use YAML by syck.

= History

[#2340](#) Removing YAML/Syck.

<http://redmine.ruby-lang.org/issues/show/2340>

After imported psych, it is found that test-all conflicts.

To avoid this, yamler is introduced.

By yamler people can switch syck or psych (default is syck).

= Current

There is 3 element:

- lib/yaml.rb
- ext/syck
- ext/psych

= Q&A

Why psych is bundled:

- To review psych

Why psych is not default:

- it is not confirmed that it can be default in 1.9.2
- it has some incompatibilities showed following.

Why syck is still bundled:

- for machines which libyaml is not installed
- for Windows

Incompatibilities are following, can you explain why it can't compatible, Aaron?

ext/syck/lib/syck.rb: YAML.generic_parser is deprecated, switch to psych"

ext/syck/lib/syck.rb: YAML.resolver is deprecated

ext/syck/lib/syck.rb: YAML.emitter is deprecated

ext/syck/lib/syck.rb: YAML.each_document is deprecated

ext/syck/lib/syck.rb: YAML.each_node is deprecated

ext/syck/lib/syck.rb: YAML.parse_documents is deprecated, use load_stream

ext/syck/lib/syck.rb: YAML.add_ruby_type is deprecated, use add_domain_ty

ext/syck/lib/syck.rb: YAML.add_private_type is deprecated, use add_domain

ext/syck/lib/syck.rb: YAML.detect_implicit is deprecated

ext/syck/lib/syck.rb: YAML.tagurize is deprecated

ext/syck/lib/syck.rb: YAML.transfer is deprecated

ext/syck/lib/syck.rb: YAML.try_implicit is deprecated

ext/syck/lib/syck.rb: YAML.read_type_class is deprecated

ext/syck/lib/syck.rb: YAML.object_maker is deprecated
ext/syck/lib/syck.rb: YAML.quick_emit is deprecated
ext/syck/lib/syck/basenode.rb: select is deprecated
ext/syck/lib/syck/basenode.rb: select!() is deprecated
ext/syck/lib/syck/basenode.rb: search() is deprecated
ext/syck/lib/syck/basenode.rb: at() is deprecated
ext/syck/lib/syck/basenode.rb: match_path is deprecated
ext/syck/lib/syck/basenode.rb: match_segment is deprecated
ext/syck/lib/syck/basenode.rb: children_with_index is deprecated, use
ext/syck/lib/syck/encoding.rb: YAML.escape is deprecated
ext/syck/lib/syck/encoding.rb: YAML.unescape is deprecated
ext/syck/lib/syck/stream.rb: edit is deprecated
ext/syck/lib/syck/stringio.rb: yaml/stringio is deprecated
ext/syck/lib/syck/ypath.rb: YAML::YPath is deprecated

=end

#2 - 04/08/2010 02:34 PM - usa (Usaku NAKAMURA)

=begin

Why psych is not default:

- it is not confirmed that it can be default in 1.9.2

You wrote that "It's decided by Yugui." at [ruby-core:28828].
Didn't it mean to make psych the default YAML engine?

=end

#3 - 04/09/2010 02:41 PM - naruse (Yui NARUSE)

=begin

First of all, people want compatibility.

Although Ruby sometimes breaks compatibility, it is important.

We should keep compability until it has enough reason.

Yes! Many of these methods are exposing implementation specific details for syck. For example, the "detect_implicit" method detects whether a YAML string is an implicit or explicit string. "try_implicit" which attempts to use "detect_implicit". "object_maker" calls allocate on a class and sets instance variables based on a hash. These are clearly methods which should never be exposed as a public API.

An API is implementation specific is not enough reason.
Implementation can return some value as if it works.
YAML.detect_implicit seems such one.

If an API is depend on syck implementation and hard to emulate,
such API can be removed.

It will trouble some users and it is sorry, but it is unavoidable.

Other methods are deprecated because they are simply duplicates. For example, YAML.load_documents and YAML.each_document do exactly the same thing. YAML.each_node and YAML.parse_documents do exactly the same thing.

Duplication is not enough reason to break compatibility.

Yes, your preference should be respect.

You can remove them with some migrating process.

Usual Ruby's process of removing API is:

- mark the API as deprecated
- release some versions
- remove it

On top of that, most of these methods are untested, which makes writing 100% compatible software impossible. All of these methods could be added to Psych, but it wouldn't be guaranteed that they work the same way as Syck.

Of course 100% is impossible.
What we need is almost compatible.
"almost" means what people use.

I understood from [ruby-core:28749], that Psych didn't need to have full compatibility. The best way I could think to warn users and maintain compatibility is to default to the old parser and let users choose to use the new one.

If psych doesn't have enough compatibility to replace in 1.9.2,
we need some migration path.

I am happy to add most of these to Psych, *if* Psych is going to be the default parser. I do not want to support these methods long term though. Please give me a clear direction, and I will make it happen.

If psych has enough compatibility, it can be a default parser in 1.9.2.
As far as I know, YAML.quick_emit breaks some applications like RubyGems.

Yeah, what is "enough compatibility" is the problem.
I think, Rails and its dependency can run can be a test.

--
NARUSE, Yui
naruse@airemix.jp

=end

#4 - 04/09/2010 02:43 PM - naruse (Yui NARUSE)

=begin
2010/4/8 Usaku NAKAMURA redmine@ruby-lang.org:

Issue [#3112](#) has been updated by Usaku NAKAMURA.

Why psych is not default:

- it is not confirmed that it can be default in 1.9.2

You wrote that "It's decided by Yugui." at [ruby-core:28828].
Didn't it mean to make psych the default YAML engine?

What I wanted to say is the direction replacing syck with psych is already decided.
When it will be done is not intended; 1.9.2 or 1.9.3 or later.

Sorry for misleading.

--
NARUSE, Yui
naruse@airemix.jp

=end

#5 - 04/10/2010 07:11 PM - naruse (Yui NARUSE)

=begin
I found Aaron added YAML.quick_emit to Psych, thank.

So I tried test-all with setting psych as default yaml impl as following:

```
diff --git a/lib/yaml.rb b/lib/yaml.rb
index 9b5a9b2..0151973 100644
--- a/lib/yaml.rb
+++ b/lib/yaml.rb
@@ -40,4 +40,4 @@ module Psych
ENGINE = YAML::ENGINE
end
```

```
-YAML::ENGINE.yamler = engine
+YAML::ENGINE.yamler = 'psych'
```

```
diff --git a/test/psych/helper.rb b/test/psych/helper.rb
index 61049d6..256fe60 100644
--- a/test/psych/helper.rb
+++ b/test/psych/helper.rb
@@ -55,9 +55,3 @@ module Psych
end
```

```
require 'psych'
-
-# FIXME: remove this when syck is removed
-o = Object.new
-a = o.method(:psych_to_yaml)
-b = o.method(:to_yaml)
-raise "psych should define to_yaml" unless a == b
```

And I ran this and following result:
make RUBYOPT=-w TESTS='-v -x test/psych' test-all

2) Failure:

```
test_to_yaml(Psych::TestArray) [/home/naruse/ruby/test/yaml/test_array.rb:11]:
<[{:a=>"b"}, "foo"]> expected but was
<[["taguri", "!ruby/object:Array"], ["to_yaml_style", 1]]>.
```

3) Failure:

```
test_to_yaml(Psych::TestHash) [/home/naruse/ruby/test/yaml/test_hash.rb:11]:
<{:a=>"b"}> expected but was
<{"taguri"=>"!ruby/object:Hash", "to_yaml_style"=>1}>.
```

4) Failure:

```
test_to_yaml(Psych::TestOmap) [/home/naruse/ruby/test/yaml/test_omap.rb:30]:
Expected !omap/ to match "--- !ruby/object:Psych::Omap\ntaguri: !'!ruby/object:Psych::Omap\nto_yaml_style: 1\n".
```

5) Failure:

```
test_to_yaml(Psych::TestSet) [/home/naruse/ruby/test/yaml/test_set.rb:13]:
Expected !set/ to match "--- !ruby/object:Psych::Set\ntaguri: !'!ruby/object:Psych::Set\nto_yaml_style: 1\n".
```

6) Error:

```
test_load(Psych::TestStruct):
Psych::SyntaxError: couldn't parse YAML at line 2 column 0
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse_stream'
/home/naruse/obj/ruby/.ext/common/psych.rb:118:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:105:in load'
/home/naruse/ruby/test/yaml/test_struct.rb:23:in `test_load'
```

7) Error:

```
test_to_yaml(Psych::TestSymbol):
TypeError: can't define singleton
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in extend_object'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in extend'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in quick_emit'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:15:in into_yaml'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:199:in to_yaml'
/home/naruse/ruby/test/yaml/test_symbol.rb:7:in test_to_yaml'
```

11) Failure:

```
test_execute_field(TestGemCommandsSpecificationCommand)
[/home/naruse/ruby/test/rubygems/test_gem_commands_specification_command.rb:83]:
Expected "foo", not "".
```

12) Failure:

```
test_write(TestGemConfigFile) [/home/naruse/ruby/test/rubygems/test_gem_config_file.rb:224]:
install.
Expected "--wrappers", not nil.
```

13) Failure:

```
test_write_from_hash(TestGemConfigFile) [/home/naruse/ruby/test/rubygems/test_gem_config_file.rb:257]:
backtrace.
```

Expected true, not false.

14) Failure:

```
test_sign_in_with_other_credentials_doesnt_overwrite_other_keys(TestGemGemcutterUtilities)
[/home/naruse/ruby/test/rubygems/test_gem_gemcutter_utilities.rb:64]:
Expected "a5fdbb6ba150cbb83aad2bb2fed64cf040453903", not nil.
```

15) Failure:

```
test_sign_in_with_host(TestGemGemcutterUtilities) [/home/naruse/ruby/test/rubygems/test_gem_gemcutter_utilities.rb:38]:
Expected "a5fdbb6ba150cbb83aad2bb2fed64cf040453903", not nil.
```

16) Failure:

```
test_sign_in(TestGemGemcutterUtilities) [/home/naruse/ruby/test/rubygems/test_gem_gemcutter_utilities.rb:26]:
Expected "a5fdbb6ba150cbb83aad2bb2fed64cf040453903", not nil.
```

17) Error:

```
test_to_yaml_platform_legacy(TestGemSpecification):
TypeError: allocator undefined for NilClass
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in allocate'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:175:in visit_Psych_Nodes_Mapping'
/home/naruse/obj/ruby/.ext/common/psych/visitors/visitor.rb:7:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:16:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in block in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in map'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:175:in visit_Psych_Nodes_Mapping'
/home/naruse/obj/ruby/.ext/common/psych/visitors/visitor.rb:7:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:16:in accept'
/home/naruse/obj/ruby/.ext/common/psych/nodes/node.rb:25:into_ruby'
/home/naruse/obj/ruby/.ext/common/psych.rb:106:in load'
/home/naruse/ruby/test/rubygems/test_gem_specification.rb:886:intest_to_yaml_platform_legacy'
```

18) Error:

```
test_to_yaml_fancy(TestGemSpecification):
TypeError: allocator undefined for NilClass
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in allocate'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:175:in visit_Psych_Nodes_Mapping'
/home/naruse/obj/ruby/.ext/common/psych/visitors/visitor.rb:7:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:16:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in block in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in map'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:175:in visit_Psych_Nodes_Mapping'
/home/naruse/obj/ruby/.ext/common/psych/visitors/visitor.rb:7:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:16:in accept'
/home/naruse/obj/ruby/.ext/common/psych/nodes/node.rb:25:into_ruby'
/home/naruse/obj/ruby/.ext/common/psych.rb:106:in load'
/home/naruse/ruby/test/rubygems/test_gem_specification.rb:867:intest_to_yaml_fancy'
```

19) Error:

```
test_to_yaml(TestGemSpecification):
TypeError: allocator undefined for NilClass
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in allocate'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:175:in visit_Psych_Nodes_Mapping'
/home/naruse/obj/ruby/.ext/common/psych/visitors/visitor.rb:7:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:16:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in block in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in map'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:217:in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:175:in visit_Psych_Nodes_Mapping'
/home/naruse/obj/ruby/.ext/common/psych/visitors/visitor.rb:7:in accept'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:16:in accept'
/home/naruse/obj/ruby/.ext/common/psych/nodes/node.rb:25:into_ruby'
/home/naruse/obj/ruby/.ext/common/psych.rb:106:in load'
/home/naruse/ruby/test/rubygems/test_gem_specification.rb:858:intest_to_yaml'
```

20) Error:

```
test_path_ok_gh_user(TestGemUninstaller):
TypeError: allocator undefined for NilClass
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in allocate'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:216:in revive'
/home/naruse/obj/ruby/.ext/common/psych/visitors/to_ruby.rb:175:in visit_Psych_Nodes_Mapping'
```



```
/home/naruse/obj/ruby/.ext/common/openssl/buffering.rb:336:in `close'  
/home/naruse/obj/ruby/.ext/common/openssl/buffering.rb:336:in `sysclose'  
/home/naruse/obj/ruby/.ext/common/openssl/buffering.rb:336:in `close'
```

```
30) Error:  
test_changes_after_commit_are_discarded(YAMLStoreTest):  
NameError: uninitialized constant Psych::Store  
/home/naruse/ruby/test/yaml/test_yamlstore.rb:7:in `setup'
```

```
31) Error:  
test_changes_are_not_written_on_abort(YAMLStoreTest):  
NameError: uninitialized constant Psych::Store  
/home/naruse/ruby/test/yaml/test_yamlstore.rb:7:in `setup'
```

```
32) Error:  
test_data_should_be_loaded_correctly_when_in_readonly_mode(YAMLStoreTest):  
NameError: uninitialized constant Psych::Store  
/home/naruse/ruby/test/yaml/test_yamlstore.rb:7:in `setup'
```

```
33) Error:  
test_data_should_be_loaded_correctly_when_in_readwrite_mode(YAMLStoreTest):  
NameError: uninitialized constant Psych::Store  
/home/naruse/ruby/test/yaml/test_yamlstore.rb:7:in `setup'
```

```
34) Error:  
test_opening_new_file_in_readonly_mode_should_result_in_empty_values(YAMLStoreTest):  
NameError: uninitialized constant Psych::Store  
/home/naruse/ruby/test/yaml/test_yamlstore.rb:7:in `setup'
```

```
35) Error:  
test_opening_new_file_in_readwrite_mode_should_result_in_empty_values(YAMLStoreTest):  
NameError: uninitialized constant Psych::Store  
/home/naruse/ruby/test/yaml/test_yamlstore.rb:7:in `setup'
```

```
36) Error:  
test_writing_inside_readonly_transaction_raises_error(YAMLStoreTest):  
NameError: uninitialized constant Psych::Store  
/home/naruse/ruby/test/yaml/test_yamlstore.rb:7:in `setup'
```

```
37) Failure:  
test_akira(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1180]:  
<{"A"=>"A", "B"=>"B"}> expected but was  
<{"taguri"=>"!ruby/object:Hash", "to_yaml_style"=>1}>.
```

```
38) Failure:  
test_ambiguous_comments(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:204]:  
<"Call the method #dave"> expected but was  
<"">.
```

```
39) Failure:  
test_basic_strings(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:79]:  
<"x"> expected but was  
<"">.
```

```
40) Failure:  
test_circular_references(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1252]:  
<"._.l.l.."> expected but was  
<["["taguri", "!ruby/object:Array", ["to_yaml_style", 1]]>.
```

```
41) Error:  
test_document(YAML_Unit_Tests):  
NameError: uninitialized constant Psych::Stream  
/home/naruse/ruby/test/yaml/test_yaml.rb:1146:in `test_document'
```

```
42) Failure:  
test_emitting_indicators(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1136]:  
<"Hi, from Object 1. You passed: please, pretty please"> expected but was  
<"">.
```

```
43) Failure:  
test_empty_map_key(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1291]:  
<[[]]> expected but was  
<["taguri", "to_yaml_style"]>.
```

```
44) Error:
```



```
test_numeric_cycle(YAML_Unit_Tests):
TypeError: can't define singleton
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in extend_object'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in extend'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in quick_emit'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:15:into_yaml'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:370:in to_yaml'
/home/naruse/ruby/test/yaml/test_yaml.rb:38:in assert_cycle'
/home/naruse/ruby/test/yaml/test_yaml.rb:1277:in `test_numeric_cycle'
```

45) Failure:
test_range_cycle(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1231]:
<"a".."z"> expected but was
.

46) Failure:
test_ranges(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1061]:
expected but was
.

47) Error:
test_ruby_complex(YAML_Unit_Tests):
TypeError: can't define singleton method "encode_with" for Complex
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:12:in singleton_method_added'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:12:in undefine_method'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:12:in quick_emit'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:15:into_yaml'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:425:in to_yaml'
/home/naruse/ruby/test/yaml/test_yaml.rb:22:in assert_to_yaml'
/home/naruse/ruby/test/yaml/test_yaml.rb:1124:in `test_ruby_complex'

48) Error:
test_ruby_rational(YAML_Unit_Tests):
TypeError: can't define singleton method "encode_with" for Rational
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:12:in singleton_method_added'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:12:in undefine_method'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:12:in quick_emit'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:15:into_yaml'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:405:in to_yaml'
/home/naruse/ruby/test/yaml/test_yaml.rb:22:in assert_to_yaml'
/home/naruse/ruby/test/yaml/test_yaml.rb:1112:in `test_ruby_rational'

49) Failure:
test_ruby_regexp(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1047]:
<{"simple"=>/a.b/,
"complex"=>/^A"((?:[l|v|+])*)"/,
"case-insensitive"=>/George McFly/i}> expected but was
<{"taguri"=>"!ruby/object:Hash", "to_yaml_style"=>1}>.

50) Failure:
test_ruby_struct(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1084]:
<{#,
#>}> expected but was
<[["taguri", "!ruby/object:Array"], ["to_yaml_style", 1]]>.

51) Failure:
test_spec_anchors_and_aliases(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:244]:
<[{"arrival"=>"EDI", "departure"=>"LAX", "fareref"=>"DOGMA", "currency"=>"GBP"},
{"arrival"=>"MEL", "departure"=>"SYD", "fareref"=>"MADF", "currency"=>"AUD"},
{"arrival"=>"MCO", "departure"=>"JFK", "fareref"=>"DFSF", "currency"=>"USD"}]> expected but was
<[["taguri", "!ruby/object:Array"], ["to_yaml_style", 1]]>.

52) Error:
test_spec_application_family(YAML_Unit_Tests):
Psych::SyntaxError: couldn't parse YAML at line 0 column 29
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse_stream'
/home/naruse/obj/ruby/.ext/common/psych.rb:118:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:105:in load'
/home/naruse/ruby/test/yaml/test_yaml.rb:33:in assert_parse_only'
/home/naruse/ruby/test/yaml/test_yaml.rb:792:in test_spec_application_family'

53) Failure:
test_spec_builtin_literal_blocks(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:910]:
<{"both are equal to"=>" This has no newline."}>

```

"is equal to"=>
"The \ ' \" characters may be\nfreely used. Leading white\n space is significant.\n\nLine breaks are significant.\nThus this value contains one\nempty
line and ends with a\nsingle line break, but does\nnot start with one.\n",
"also written as"=>" This has no newline.",
"indented and chomped"=>" This has no newline.",
"empty"=>""",
"literal"=>
"The \ ' \" characters may be\nfreely used. Leading white\n space is significant.\n\nLine breaks are significant.\nThus this value contains one\nempty
line and ends with a\nsingle line break, but does\nnot start with one.\n"}> expected but was
<{"empty"=>""",
"literal"=>
"The \ ' \" characters may be\nfreely used. Leading white\n space is significant.\n\nLine breaks are significant.\nThus this value contains one\nempty
line and ends with a\nsingle line break, but does\nnot start with one.\n",
"is equal to"=>
"The \ ' \" characters may be\nfreely used. Leading white\n space is significant.\n\nLine breaks are significant.\nThus this value contains one\nempty
line and ends with a\nsingle line break, but does\nnot start with one.\n",
"indented and chomped"=>" This has no newline.",
"also written as"=>" This has no newline.",
"both are equal to"=>" This has no newline."}>.

```

```

54) Error:
test_spec_domain_prefix(YAML_Unit_Tests):
Psych::SyntaxError: couldn't parse YAML at line 1 column 26
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse_stream'
/home/naruse/obj/ruby/.ext/common/psych.rb:118:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:105:in load'
/home/naruse/ruby/test/yaml/test_yaml.rb:33:in assert_parse_only'
/home/naruse/ruby/test/yaml/test_yaml.rb:647:in test_spec_domain_prefix'

```

```

55) Error:
test_spec_float_explicit(YAML_Unit_Tests):
Psych::SyntaxError: couldn't parse YAML at line 4 column 17
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse_stream'
/home/naruse/obj/ruby/.ext/common/psych.rb:118:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:105:in load'
/home/naruse/ruby/test/yaml/test_yaml.rb:33:in assert_parse_only'
/home/naruse/ruby/test/yaml/test_yaml.rb:810:in test_spec_float_explicit'

```

```

56) Error:
test_spec_private_types(YAML_Unit_Tests):
NoMethodError: undefined method parse_documents' for Psych:Module
/home/naruse/ruby/test/yaml/test_yaml.rb:693:in test_spec_private_types'

```

```

57) Error:
test_spec_root_fold(YAML_Unit_Tests):
Psych::SyntaxError: couldn't parse YAML at line 5 column 0
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse_stream'
/home/naruse/obj/ruby/.ext/common/psych.rb:118:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:105:in load'
/home/naruse/ruby/test/yaml/test_yaml.rb:589:in `test_spec_root_fold'

```

```

58) Failure:
test_spec_simple_implicit_map(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:126]:
<{"hr"=>65, "avg"=>0.278, "rbi"=>147}> expected but was
<{"taguri"=>"!ruby/object:Hash", "to_yaml_style"=>1}>.

```

```

59) Failure:
test_spec_simple_implicit_sequence(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:115]:
<["Mark McGwire", "Sammy Sosa", "Ken Griffey"]> expected but was
<[["taguri", "!ruby/object:Array"], ["to_yaml_style", 1]]>.

```

```

60) Failure:
test_spec_simple_map_with_nested_sequences(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:140]:
<{"american"=>["Boston Red Sox", "Detroit Tigers", "New York Yankees"],
"national"=>["New York Mets", "Chicago Cubs", "Atlanta Braves"]}> expected but was
<{"taguri"=>"!ruby/object:Hash", "to_yaml_style"=>1}>.

```

```

61) Failure:
test_spec_simple_sequence_with_nested_map(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:159]:
<{"name"=>"Mark McGwire", "hr"=>65, "avg"=>0.278},
{"name"=>"Sammy Sosa", "hr"=>63, "avg"=>0.288}> expected but was
<[["taguri", "!ruby/object:Array"], ["to_yaml_style", 1]]>.

```

```
62) Error:
test_spec_url_escaping(YAML_Unit_Tests):
Psych::SyntaxError: couldn't parse YAML at line 1 column 25
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:147:in parse_stream'
/home/naruse/obj/ruby/.ext/common/psych.rb:118:in parse'
/home/naruse/obj/ruby/.ext/common/psych.rb:105:in load'
/home/naruse/ruby/test/yaml/test_yaml.rb:33:in assert_parse_only'
/home/naruse/ruby/test/yaml/test_yaml.rb:725:in test_spec_url_escaping'
```

```
63) Error:
test_symbol_cycle(YAML_Unit_Tests):
TypeError: can't define singleton
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in extend_object'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in extend'
/home/naruse/obj/ruby/.ext/common/psych/deprecated.rb:10:in quick_omit'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:15:in into_yaml'
/home/naruse/obj/ruby/.ext/common/syck/rubytypes.rb:199:in to_yaml'
/home/naruse/ruby/test/yaml/test_yaml.rb:38:in assert_cycle'
/home/naruse/ruby/test/yaml/test_yaml.rb:1262:in test_symbol_cycle'
```

```
64) Failure:
test_time_now_cycle(YAML_Unit_Tests) [/home/naruse/ruby/test/yaml/test_yaml.rb:1220]:
expected but was
.
```

```
65) Error:
test_yaml_parsing(YAML_Unit_Tests):
NameError: uninitialized constant Psych::YPath
/home/naruse/ruby/test/yaml/test_yaml.rb:42:in assert_path_segments'
/home/naruse/ruby/test/yaml/test_yaml.rb:1165:in test_yaml_parsing'
```

```
--
NARUSE, Yui naruse@airemix.jp
```

```
=end
```

#6 - 04/11/2010 09:20 AM - naruse (Yui NARUSE)

```
=begin
(2010/04/11 8:48), Aaron Patterson wrote:
```

I am surprised about the rubygems failures. With the following patch, I am able to run the rubygems tests, like this:

```
$ GEM_HOME=/tmp GEM_PATH=/tmp make test-all TESTS=rubygems
```

```
diff --git a/lib/yaml.rb b/lib/yaml.rb
index 9b5a9b2..0151973 100644
--- a/lib/yaml.rb
+++ b/lib/yaml.rb
@@ -40,4 +40,4 @@ module Psych
ENGINE = YAML::ENGINE
end
```

```
-YAML::ENGINE.yamler = engine
+YAML::ENGINE.yamler = 'psych'
diff --git a/test/rubygems/gemutilities.rb b/test/rubygems/gemutilities.rb
index bb94750..cea1dbe 100644
--- a/test/rubygems/gemutilities.rb
+++ b/test/rubygems/gemutilities.rb
@@ -8,13 +8,7 @@ else
require 'rubygems'
end
require 'fileutils'
-begin
```

- gem 'minitest', '>= 1.3.1'
- require 'minitest/unit' -rescue Gem::LoadError
- warn "Install minitest gem >= 1.3.1"
- raise -end +require 'minitest/unit' require 'tmpdir' require 'uri' require 'rubygems/package'

Bundled minitest is 1.6.0(-dev?) after r27076, so why this is needed...

```
diff --git a/test/rubygems/test_gem_specification.rb b/test/rubygems/test_gem_specification.rb
index 5979890..337073f 100644
--- a/test/rubygems/test_gem_specification.rb
+++ b/test/rubygems/test_gem_specification.rb
@@ -1,5 +1,6 @@
require_relative 'gemutilities'
require 'stringio'
+require 'date'
require 'rubygems/specification'
```

```
class TestGemSpecification < RubyGemTestCase
```

Some of the Syck YAML tests (like `test_spec_application_family` and `test_spec_url_escaping`) contain invalid YAML, so Psych will raise an exception. Do `diff test/yaml/test_yaml.rb test/psych/test_yaml.rb` to see the differences.

Yeah, I know some of them are intended.
I want to confirm taht all of them are from invalid YAML.

I will make `test/yaml/*` use the syck engine.

OK, it may have to move to `test/syck`.

I can't run `make test-all` on Snow Leopard because of the problem I mentioned in [ruby-core:29087]. I will try on linux.

Can you make a ticket in Redmine for [ruby-core:29087]?

For workaround you can run `test-all` like following:
`make RUBYOPT=-w TESTS='-v -x "test/dr|test/socket"' test-all`

--
NARUSE, Yui naruse@airemix.jp

=end

#7 - 04/11/2010 10:23 AM - drbrain (Eric Hodel)

```
=begin
On Apr 10, 2010, at 17:58, Aaron Patterson wrote:
diff --git a/test/rubygems/gemutilities.rb b/test/rubygems/gemutilities.rb
```

```
index bb94750..cea1dbe 100644
--- a/test/rubygems/gemutilities.rb
+++ b/test/rubygems/gemutilities.rb
@@ -8,13 +8,7 @@ else
require 'rubygems'
end
require 'fileutils'
-begin
```

- `gem 'minitest', '>= 1.3.1'`
- `require 'minitest/unit' -rescue Gem::LoadError`
- `warn "Install minitest gem>= 1.3.1"`
- `raise -end +require 'minitest/unit' require 'tmpdir' require 'uri' require 'rubygems/package'`

Bundled minitest is 1.6.0(-dev?) after r27076, so why this is needed...

I'm not sure. I've applied this to trunk with Eric's permission.

Won't the bundled minitest will only be available after install?

=end

#8 - 04/11/2010 01:59 PM - naruse (Yui NARUSE)

```
=begin
(2010/04/11 10:23), Eric Hodel wrote:
```

On Apr 10, 2010, at 17:58, Aaron Patterson wrote:
diff --git a/test/rubygems/gemutilities.rb b/test/rubygems/gemutilities.rb

```
index bb94750..cea1dbe 100644
--- a/test/rubygems/gemutilities.rb
+++ b/test/rubygems/gemutilities.rb
@@ -8,13 +8,7 @@ else
require 'rubygems'
end
require 'fileutils'
-begin

  • gem 'minitest', '>= 1.3.1'
  • require 'minitest/unit' -rescue Gem::LoadError
  • warn "Install minitest gem>= 1.3.1"
  • raise -end +require 'minitest/unit' require 'tmpdir' require 'uri' require 'rubygems/package'
```

Bundled minitest is 1.6.0(-dev?) after r27076, so why this is needed...

I'm not sure. I've applied this to trunk with Eric's permission.

Won't the bundled minitest will only be available after install?

Ah, yes, so this can be a bug of Gem's load path.
Also note that this can be avoided by install before test-all.

--
NARUSE, Yui naruse@airemix.jp

=end

#9 - 04/11/2010 02:04 PM - naruse (Yui NARUSE)

=begin
(2010/04/11 8:48), Aaron Patterson wrote:

```
diff --git a/test/rubygems/test_gem_specification.rb b/test/rubygems/test_gem_specification.rb
index 5979890..337073f 100644
--- a/test/rubygems/test_gem_specification.rb
+++ b/test/rubygems/test_gem_specification.rb
@@ -1,5 +1,6 @@
require_relative 'gemutilities'
require 'stringio'
+require 'date'
require 'rubygems/specification'

class TestGemSpecification < RubyGemTestCase
```

This is from the difference of loaded libraries between syck and psych.
So this is considered as compatibility problem and this is fixed by psych.

Example is following but where "require 'date' is is upon you.

```
diff --git a/ext/psych/lib/psych/deprecated.rb b/ext/psych/lib/psych/deprecated.rb
index 5a96e91..c0c9abe 100644
--- a/ext/psych/lib/psych/deprecated.rb
+++ b/ext/psych/lib/psych/deprecated.rb
@@ -1,3 +1,5 @@
+require 'date'
+
module Psych
module DeprecatedMethods # :nodoc:
attr_accessor :taguri
```

--
NARUSE, Yui naruse@airemix.jp

=end

#10 - 04/14/2010 10:22 AM - naruse (Yui NARUSE)

=begin
Current syck is mixed old YAML and YAML::Syck.
This breaks compatibility like following:
<http://pc12.2ch.net/test/read.cgi/tech/1265467681/871>

It also says require 'yaml/syck' should work.

People may require 'yaml' and use YAML::Syck, but we should give up such case.
=end

#11 - 05/06/2010 05:00 PM - usa (Usaku NAKAMURA)

=begin
Hello,

In message "[ruby-core:30044] Re: [Bug #3112] require 'yaml' doesn't use psych as default"
on May.06,2010 15:46:00, aaron@tenderlovmaking.com wrote:

Bug #3112: require "yaml" doesn't use psych as default
<http://redmine.ruby-lang.org/issues/show/3112>

Author: Usaku NAKAMURA
Status: Open, Priority: Normal
Category: lib, Target version: 1.9.2
ruby -v: ruby 1.9.2dev (2010-04-08 trunk 27255)

why?

Rails runs with Psych as default. Our applications at work run well
with Psych as default.

I would like to make Psych default YAML parser for 1.9.2. What do you
think?

IMO, if Psych is enough compatible with Syck, Psych should be
the default YAML parser, like the way of the decision performed
about ext/fiddle and ext/dl recently.

However, naruse has already explained the intention of yugui at
[ruby-core:29377].
I don't want to disobey the branch/release manager about this
matter :)

Regards,
--
U.Nakamura usa@garbagecollect.jp

=end

#12 - 05/18/2010 08:30 PM - yugui (Yuki Sonoda)

- Target version changed from 1.9.2 to 2.0.0

=begin
I think syck works more or less correctly unlike dl2. So I want to decide it conservatively. I don't want to switch the default yaml engine to psych.
=end

#13 - 05/19/2010 03:22 AM - wycats (Yehuda Katz)

=begin
Yugui,

I am worried about possible encoding issues that can come in Syck that will
be hard to fix. For instance, Aaron was able to fix Psych to honor
default_internal in just a few days, and there is no real maintainer for
Syck.

Since Aaron is the maintainer for Psych, and there is no maintainer for
Syck, I think it makes sense to make Psych the default, with the yamler=
option for backward compatibility.

Yehuda Katz
Architect | Engine Yard
(ph) 718.877.1325

On Tue, May 18, 2010 at 3:30 PM, Yuki Sonoda redmine@ruby-lang.org wrote:

Issue [#3112](#) has been updated by Yuki Sonoda.

Target version changed from 1.9.2 to 1.9.x

I think syck works more or less correctly unlike dl2. So I want to decide

it conservatively. I don't want to switch the default yamll engine to psych.

<http://redmine.ruby-lang.org/issues/show/3112>

<http://redmine.ruby-lang.org>

Yugui, I am worried about possible encoding issues that can come in Syck that will be hard to fix. For instance, Aaron was able to fix Psych to honor default_internal in just a few days, and there is no real maintainer for Syck.

Since Aaron is the maintainer for Psych, and there is no maintainer for Syck, I think it makes sense to make Psych the default, with the yamll= option for backward compatibility.

Yehuda Katz Architect | Engine Yard (ph) 718.877.1325

On Tue, May 18, 2010 at 3:30 PM, Yuki Sonoda <redmine@ruby-lang.org> wrote:

Issue [#3112](#) has been updated by Yuki Sonoda.

Target version changed from 1.9.2 to 1.9.x

I think syck works more or less correctly unlike dl2. So I want to decide it conservatively. I don't want to switch the default yamll engine to psych.

<http://redmine.ruby-lang.org/issues/show/3112>

<http://redmine.ruby-lang.org>

=end

#14 - 05/19/2010 03:32 AM - tenderlovmaking (Aaron Patterson)

=begin

On Tue, May 18, 2010 at 08:30:14PM +0900, Yuki Sonoda wrote:

Issue [#3112](#) has been updated by Yuki Sonoda.

Target version changed from 1.9.2 to 1.9.x

I think syck works more or less correctly unlike dl2. So I want to decide it conservatively. I don't want to switch the default yamll engine to psych.

I'm afraid that if we use Syck as default, no one will see errors from invalid YAML files. If we make Psych as default, people will see errors with their YAML files but *have the option* to move back to Syck.

Are we going to remove Syck in 1.9.3? If we don't make Psych default in 1.9.2, it seems to me that we have to wait until 1.9.4 to remove Syck.

Since Syck has no active maintainer, it seems bad to wait so long to remove it.

--

Aaron Patterson

<http://tenderlovmaking.com/>

Attachment: (unnamed)

=end

#15 - 05/24/2010 08:29 AM - wycats (Yehuda Katz)

=begin

Bump.

I think the lack of a Syck maintainer, yet making Syck the default in 1.9.2 needs to be addressed. 1.9.2 is probably going to be the de-facto release for some time. What's the rationale for keeping an unmaintained major component when a maintained version is available in the standard library?

Yehuda Katz
Architect | Engine Yard
(ph) 718.877.1325

On Tue, May 18, 2010 at 11:31 AM, Aaron Patterson <aaron@tenderlovmaking.com> wrote:

On Tue, May 18, 2010 at 08:30:14PM +0900, Yuki Sonoda wrote:

Issue [#3112](#) has been updated by Yuki Sonoda.

Target version changed from 1.9.2 to 1.9.x

I think syck works more or less correctly unlike dl2. So I want to decide it conservatively. I don't want to switch the default yaml engine to psych.

I'm afraid that if we use Syck as default, no one will see errors from invalid YAML files. If we make Psych as default, people will see errors with their YAML files but *have the option* to move back to Syck.

Are we going to remove Syck in 1.9.3? If we don't make Psych default in 1.9.2, it seems to me that we have to wait until 1.9.4 to remove Syck.

Since Syck has no active maintainer, it seems bad to wait so long to remove it.

--

Aaron Patterson
<http://tenderlovmaking.com/>

Bump. I think the lack of a Syck maintainer, yet making Syck the default in 1.9.2 needs to be addressed. 1.9.2 is probably going to be the de-facto release for some time. What's the rationale for keeping an unmaintained major component when a maintained version is available in the standard library?

Yehuda Katz Architect | Engine Yard (ph) 718.877.1325

On Tue, May 18, 2010 at 11:31 AM, Aaron Patterson <aaron@tenderlovmaking.com> wrote:

On Tue, May 18, 2010 at 08:30:14PM +0900, Yuki Sonoda wrote:

> Issue [#3112](#) has been updated by Yuki Sonoda.

>

> Target version changed from 1.9.2 to 1.9.x

>

> I think syck works more or less correctly unlike dl2. So I want to decide it conservatively. I don't want to switch the default yaml engine to psych.

I'm afraid that if we use Syck as default, no one will see errors from invalid YAML files. If we make Psych as default, people will see errors with their YAML files but *have the option* to move back to Syck.

Are we going to remove Syck in 1.9.3? If we don't make Psych default in 1.9.2, it seems to me that we have to wait until 1.9.4 to remove Syck.

Since Syck has no active maintainer, it seems bad to wait so long to remove it.

--

Aaron Patterson
<http://tenderlovmaking.com/>

=end

#16 - 08/27/2010 09:09 PM - anshul (Anshul Khandelwal)

=begin

Any updates on this? Is there some way I can ensure that my code is using the bundled psych?

Anshul

=end

#17 - 08/29/2010 07:24 AM - tenderlovmaking (Aaron Patterson)

=begin

On Fri, Aug 27, 2010 at 09:09:27PM +0900, Anshul Khandelwal wrote:

Issue [#3112](#) has been updated by Anshul Khandelwal.

Any updates on this? Is there some way I can ensure that my code is using the bundled psych?

Psych is an "opt-in" experience for 1.9.2. I'm not sure when we'll make it the default YAML object.

There are two ways you can use Psych. The first way is to just require psych and reference the Psych constant:

```
require 'psych'

Psych.load '--- hello world!'
Psych.dump { :goodbye => 'cruel world' }
```

The second way is to change the YAML engine:

```
require 'yaml'

YAML::ENGINE.yamler = 'psych'
YAML.load '--- hello world!'
YAML.dump { :goodbye => 'cruel world' }
```

Setting the engine will replace the YAML constant with the Psych constant. Using the first example will absolutely guarantee using Psych. The second example cannot guarantee usage because other libraries could possibly set the engine back to "syck" without your knowledge.

--

Aaron Patterson

<http://tenderlovmaking.com/>

Attachment: (unnamed)

=end

#18 - 08/29/2010 04:11 PM - nobu (Nobuyoshi Nakada)

=begin

(10/08/29 7:24), Aaron Patterson wrote:

On Fri, Aug 27, 2010 at 09:09:27PM +0900, Anshul Khandelwal wrote:
There are two ways you can use Psych. The first way is to just require psych and reference the Psych constant:

```
require 'psych'

Psych.load '--- hello world!'
Psych.dump { :goodbye => 'cruel world' }
```

The second way is to change the YAML engine:

```
require 'yaml'

YAML::ENGINE.yamler = 'psych'
YAML.load '--- hello world!'
YAML.dump { :goodbye => 'cruel world' }
```

And the third way is to require psych before yaml:

```
require 'psych'
require 'yaml'

# p YAML::ENGINE.yamler # => 'psych'
YAML.load '--- hello world!'
YAML.dump { :goodbye => 'cruel world' }
```

--

Nobu Nakada

=end

#19 - 06/26/2011 02:16 PM - naruse (Yui NARUSE)

- Status changed from Open to Closed