Warning for Fixnum#size to use RbConfig::SIZEOF['long']

01/10/2015 11:42 AM - akr (Akira Tanaka)

Status: Open
Priority: Normal
Assignee:
Target version:

Description
How about add a warning for Fixnum#size?

```
% ./ruby -e 'p 0.size'
-e:1: warning: Use RbConfig::SIZEOF['long'] instead of Fixnum#size
8
```
Currently it returns sizeof(long) which is 4 or 8.

However it is implementation detail and it may difficult to define it in Ruby implementations other than CRuby.

There is a way to obtain sizeof(long): RbConfig::SIZEOF['long']
It works since Ruby 2.1.

```
% ruby -rrbconfig/sizeof -e "p RbConfig::SIZEOF['long']"
8
```
So the warning is a migration path to use RbConfig::SIZEOF['long'].

Note that this may affect many code. So this issue may irritate many people.
fixnum-size-search.txt is a search result in gems.

Related issues:
Related to Ruby master - Feature #7517: Fixnum::MIN,MAX

Closed

History
#1 - 01/10/2015 01:03 PM - cremno (cremno phobia)
I took a look at fixnum-size-search.txt and using RbConfig::SIZEOF['long'] instead usually isn't the correct replacement. In many lines the size of long in bytes isn't of interest, the min- and maximum value of a fixnum is (see feature request #7517).

#2 - 01/10/2015 01:19 PM - akr (Akira Tanaka)
- Related to Feature #7517: Fixnum::MIN,MAX added

#3 - 01/11/2015 04:23 PM - akr (Akira Tanaka)
RbConfig::SIZEOF['long'] is a drop-in replacement but Fixnum::MAX is not.

Also, I feel Fixnum is a implementation detail.

Files
fixnum-size-search.txt 7.03 KB 01/10/2015 akr (Akira Tanaka)
fixnum-size-warning.patch 2.03 KB 01/10/2015 akr (Akira Tanaka)