



## RURBAN / Storable-3.05 / ChangeLog

```
1      Sun Jan 29 11:36:43 2017 +0100 Reini Urban <rurban@cpanel.net>
2          Version 3.05
3
4          * Protect against classname len overflow on the stack
5            and 2x on the heap with retrieve_bless and retrieve_hook.
6            A serious security issue with malcrafted storable files or buffers,
7            but p5p accepts no CVE on Storable attacks. See RT #130635
8            (reported by JD).
9          * Fix NULL ptr SEGVs with retrieve_code and retrieve_other.
10         See RT #130098 (reported and fixed by JD)
11         * Fix wrong huge LOBJECT support, broken since 3.00c.
12         Repro with `export PERL_TEST_MEMORY=8`
13         * Fix the few remaining 2-arg open calls.
14         * Portability and backport fixes back to 5.6.2
15
16      Sat Jan 7 09:01:29 2017 +0100 Reini Urban <rurban@cpanel.net>
17          Version 3.04c
18
19          * fix printf types and warnings, esp. for 32bit use64bitint
20          * Change sv_setpvn(..., "...", ...) to sv_setpvs(..., "...")
21
22      Tue Jul 26 11:49:33 2016 +1000 Tony Cook <tony@develop-help.com>
23          Version 3.03c
24
25          * remove . from @INC when loading optional modules
26
27      Sun Nov 20 18:06:45 2016 +0100 Reini Urban <rurban@cpanel.net>
28          Version 3.02c
29
30          * Fix -Wc++11-compat warnings, fix -Wchar-subscripts
31
32      Fri Sep 16 01:32:59 2016 +0200 Reini Urban <rurban@cpanel.net>
33          Version 3.01c
34
35          * Added warn_security("Movable-Type CVE-2015-1592 Storable metasploit
36            when detecting the third destructive metasploit vector,
37            thawing bless \"mt-config.cgi\", \"CGITempFile\".
38
39      Thu Mar 31 17:10:27 2016 +0200 Reini Urban <rurban@cpanel.net>
40          Version 3.00c
41
42          * Added support for u64 strings, arrays and hashes >2G
43            via a new LOBJECT tag. This is for 32bit systems and lengths
44            between 2GB and 4GB (I32-U32), and 64bit (>I32).
45          * Bumped STORABLE_BIN_MINOR and STORABLE_BIN_WRITE_MINOR from 10 to
46          * fix parallel tests, use unique filenames.
47          * fixed 2 instances of 2arg open,
48          * added optional flag arguments to skip tie and bless on retrieve/
49          * added SECURITY WARNING and Large data support to docs
50          * compute CAN_FLOCK at compile-time
51          * reformat everything consistently
```

[56](#) Wed Jul 2 16:25:25 IST 2014 Abhijit Menon-Sen <ams@toroid.org>  
[57](#) Version 2.51  
[58](#)  
[59](#) \* [perl #121928] Fix memory leak for dclone inside freeze hook  
[60](#) (Alex Solovey)  
[61](#) \* Do not call DESTROY for empty objects  
[62](#) (Vladimir Timofeev)  
[63](#) \* Other bugfixes  
[64](#)

[65](#) Sat Jul 13 18:34:27 IST 2013 Abhijit Menon-Sen <ams@toroid.org>  
[66](#) Version 2.45  
[67](#)  
[68](#) \* [perl #118829] Memory leaks in STORABLE\_attach  
[69](#) (Vladimir Timofeev)  
[70](#) \* [perl #118139] Don't SEGV during global destruction  
[71](#) (Nicholas Clark, report/test from Reini Urban)  
[72](#) \* Added security warnings section (Steffen Mueller)  
[73](#) \* Update INSTALLDIRS to favour installation in 'site'  
[74](#) (James E Keenan)  
[75](#)

[76](#) Tue 11 Sep 06:51:11 IST 2012 Abhijit Menon-Sen <ams@toroid.org>  
[77](#) Version 2.39  
[78](#)  
[79](#) Various bugfixes, including compatibility fixes for older  
[80](#) versions of Perl and vstring handling.  
[81](#)

[82](#) Sun 3 Jul 09:10:11 IST 2011 Abhijit Menon-Sen <ams@toroid.org>  
[83](#) Version 2.29  
[84](#)  
[85](#) Various bugfixes, notably including preventing nfreeze from  
[86](#) incorrectly stringifying integers.  
[87](#)

[88](#) Fri 3 Dec 14:12:32 GMT 2010 David Leadbeater <dgl@dgl.cx>  
[89](#) Version 2.25  
[90](#)  
[91](#) Support for serializing coderefs containing UTF-8.  
[92](#)

[93](#) Fri Nov 12 10:52:19 IST 2010 Abhijit Menon-Sen <ams@toroid.org>  
[94](#)  
[95](#) Version 2.24  
[96](#)  
[97](#) Performance improvement for overloaded classes from Benjamin  
[98](#) Holzman.  
[99](#)

[100](#) Fri Nov 12 10:36:22 IST 2010 Abhijit Menon-Sen <ams@toroid.org>  
[101](#)  
[102](#) Version 2.23  
[103](#)  
[104](#) Release the latest version from the Perl repository.  
[105](#)

[106](#) Thu Aug 6 10:55:50 IST 2009 Abhijit Menon-Sen <ams@toroid.org>  
[107](#)  
[108](#) Version 2.21  
[109](#)  
[110](#) Includes hints/hpux.pl that was inadvertently left out of 2.20.  
[111](#)

[112](#) Mon May 18 09:38:20 IST 2009 Abhijit Menon-Sen <ams@toroid.org>  
[113](#)

[118](#)  
[119](#) (Version 2.19 was released with 5.8.9.)  
[120](#)  
[121](#) Thu Nov 22 13:24:18 IST 2007 Abhijit Menon-Sen <ams@toroid.org>  
[122](#)  
[123](#) Version 2.18  
[124](#)  
[125](#) Compile fixes for older Perls. (No functional changes.)  
[126](#)  
[127](#) Sat Nov 17 02:12:12 IST 2007 Abhijit Menon-Sen <ams@toroid.org>  
[128](#)  
[129](#) Version 2.17  
[130](#)  
[131](#) Various broken tests fixed. (No functional changes.)  
[132](#)  
[133](#) Sat Mar 31 06:11:06 IST 2007 Abhijit Menon-Sen <ams@toroid.org>  
[134](#)  
[135](#) Version 2.16  
[136](#)  
[137](#) 1. Fixes to Storable::dclone, read\_magic, retrieve\_lscalar  
[138](#) 2. Storable 0.1 compatibility  
[139](#) 3. Miscellaneous compile/leak/test/portability fixes  
[140](#)  
[141](#) Mon May 23 22:48:49 IST 2005 Abhijit Menon-Sen <ams@wiw.org>  
[142](#)  
[143](#) Version 2.15  
[144](#)  
[145](#) Minor changes to address a couple of compile problems.  
[146](#)  
[147](#) Mon Apr 25 07:29:14 IST 2005 Abhijit Menon-Sen <ams@wiw.org>  
[148](#)  
[149](#) Version 2.14  
[150](#)  
[151](#) 1. Store weak references  
[152](#) 2. Add STORABLE\_attach hook.  
[153](#)  
[154](#) Thu Jun 17 12:26:43 BST 2004 Nicholas Clark <nick@ccl4.org>  
[155](#)  
[156](#) Version 2.13  
[157](#)  
[158](#) 1. Don't change the type of top level overloaded references to RV  
[159](#) they are perfectly correct as PVMG  
[160](#) 2. Storable needs to cope with incoming frozen data that happens to  
[161](#) utf8 encoded.  
[162](#)  
[163](#) Wed Mar 17 15:40:29 GMT 2004 Nicholas Clark <nick@ccl4.org>  
[164](#)  
[165](#) Version 2.12  
[166](#)  
[167](#) 1. Add regression tests for the auto-require of STORABLE\_thaw  
[168](#) 2. Add auto-require of modules to restore overloading (and tests)  
[169](#) 3. Change to no context (should give speedup with ithreads)  
[170](#)  
[171](#) Sat Mar 13 20:11:03 GMT 2004 Nicholas Clark <nick@ccl4.org>  
[172](#)  
[173](#) Version 2.11  
[174](#)  
[175](#) 1. Storing restricted hashes in canonical order would SEGV. Fixed.

[180](#) 4. These changes allow a space optimisation for restricted hashes.  
[181](#)  
[182](#) Sat Jan 24 16:22:32 IST 2004 Abhijit Menon-Sen <ams@wiw.org>  
[183](#)  
[184](#) Version 2.10  
[185](#)  
[186](#) 1. Thread safety: Storable::CLONE/init\_perlinterp() now create  
[187](#) a new Perl context for each new ithread.  
[188](#) (From Stas Bekman and Jan Dubois.)  
[189](#) 2. Fix a tag count mismatch with \$Storable::Deparse that caused  
[190](#) all back-references after a stored sub to be off-by-N (where  
[191](#) N was the number of code references in between).  
[192](#) (From Sam Vilain.)  
[193](#) 3. Prevent CODE references from turning into SCALAR references.  
[194](#) (From Slaven Rezic.)  
[195](#)  
[196](#) Sat Jan 3 18:49:18 GMT 2004 Nicholas Clark <nick@ccl4.org>  
[197](#)  
[198](#) Version 2.09  
[199](#)  
[200](#) Fix minor problems with the CPAN release  
[201](#) 1: Make Storable.xs work on 5.8.2 and later (already in the core)  
[202](#) 2: Ship the linux hints file  
[203](#) 3: Ship Test::More for the benefit of Perls pre 5.6.2  
[204](#) 4: Correct Makefile.PL to only install in core for 5.8.0 and later  
[205](#)  
[206](#) Sat Sep 6 01:08:20 IST 2003 Abhijit Menon-Sen <ams@wiw.org>  
[207](#)  
[208](#) Version 2.08  
[209](#)  
[210](#) This release works around a 5.8.0 bug which caused hashes to not  
[211](#) be marked as having key flags even though an HEK had HEK\_WASUTF8  
[212](#) set. (Note that the only reasonable solution is to silently drop  
[213](#) the flag from the affected key.)  
[214](#)  
[215](#) Users of RT 3 who were seeing assertion failures should upgrade.  
[216](#) (Perl 5.8.1 will have the bug fixed.)  
[217](#)  
[218](#) Mon May 5 10:24:16 IST 2003 Abhijit Menon-Sen <ams@wiw.org>  
[219](#)  
[220](#) Version 2.07  
[221](#)  
[222](#) Minor bugfixes (self-tied objects are now correctly stored, as  
[223](#) are the results of additions larger than INT\_MAX).  
[224](#)  
[225](#) Mon Oct 7 21:56:38 BST 2002 Nicholas Clark <nick@ccl4.org>  
[226](#)  
[227](#) Version 2.06  
[228](#)  
[229](#) Remove qr// from t/downgrade.t so that it will run on 5.004  
[230](#) Mention \$File::Spec::VERSION a second time in t/forgive.t so that  
[231](#) runs without warnings in 5.004 (this may be a 5.00405 bug I'm worki  
[232](#) round)  
[233](#) Fix t/integer.t initialisation to actually generate 64 bits of 9c  
[234](#) Fix comparison tests to use eval to get around 64 bit IV conversion  
[235](#) issues on 5.6.x, following my t/integer.t ^ precedence bug found by  
[236](#) Rafael Garcia-Suarez  
[237](#) Alter t/malice.t to work with Test/More.pm in t/, and skip individu

[242](#) Remove prerequisite of Test::More from Makefile.PL  
[243](#) Ship Test::Builder, Test::Simple and Test::More in t  
[244](#)  
[245](#) Thu Oct 3 08:57:22 IST 2002 Abhijit Menon-Sen <ams@wiw.org>  
[246](#)  
[247](#) Version 2.05  
[248](#)  
[249](#) Adds support for CODE references from Slaven Rezic  
[250](#) <slaven.rezic@berlin.de>.  
[251](#)  
[252](#) Fri Jun 7 23:55:41 BST 2002 Nicholas Clark  
[253](#)  
[254](#) Version 2.04  
[255](#)  
[256](#) Bug fix from Radu Greab <radu@netsoft.ro> (plus regression test)  
[257](#) to fix a recently introduced bug detected by Dave Rolsky.  
[258](#) Bug was that for a non threaded build, the class information was  
[259](#) being lost at freeze time on the first object with a STORABLE\_freeze  
[260](#) hook. Consequentially the object was not blessed at all when thawed  
[261](#) (The presence (or lack) of STORABLE\_thaw was irrelevant; this was  
[262](#) a store-time data lost bug, caused by failure to initialize internal  
[263](#) context)  
[264](#) The bug was introduced as development perl change 16442 (on  
[265](#) 2002/05/07), so has been present since 2.00.  
[266](#) Patches to introduce more regression tests to reduce the chance of  
[267](#) a reoccurrence of this sort of goof are always welcome.  
[268](#)  
[269](#) Thu May 30 20:31:08 BST 2002 Nicholas Clark <nick@ccl4.org>  
[270](#)  
[271](#) Version 2.03 Header changes on 5.6.x on Unix where IV is long long  
[272](#)  
[273](#) 5.6.x introduced the ability to have IVs as long long. However,  
[274](#) Configure still defined BYTEORDER based on the size of a long.  
[275](#) Storable uses the BYTEORDER value as part of the header, but  
[276](#) doesn't explicitly store sizeof(IV) anywhere in the header.  
[277](#) Hence on 5.6.x built with IV as long long on a platform that  
[278](#) uses Configure (ie most things except VMS and Windows) headers  
[279](#) are identical for the different IV sizes, despite the files  
[280](#) containing some fields based on sizeof(IV)  
[281](#)  
[282](#) 5.8.0 is consistent; all platforms have BYTEORDER in config.h  
[283](#) based on sizeof(IV) rather than sizeof(long). This means that  
[284](#) the value of BYTEORDER will change from (say) 4321 to 87654321  
[285](#) between 5.6.1 and 5.8.0 built with the same options to Configure  
[286](#) on the same machine. This means that the Storable header will  
[287](#) differ, and the two versions will wrongly thing that they are  
[288](#) incompatible.  
[289](#)  
[290](#) For the benefit of long term consistency, Storable now  
[291](#) implements the 5.8.0 BYTEORDER policy on 5.6.x. This means that  
[292](#) 2.03 onwards default to be incompatible with 2.02 and earlier  
[293](#) (ie the large 1.0.x installed base) on the same 5.6.x perl.  
[294](#)  
[295](#) To allow interworking, a new variable  
[296](#) \$Storable::interwork\_56\_64bit is introduced. It defaults to  
[297](#) false. Set it to true to read and write old format files. Don't  
[298](#) use it unless you have existing stored data written with 5.6.x  
[299](#) that you couldn't otherwise read, or you need to interwork with

[304](#)[305](#)

Version 2.02

[306](#)[307](#)

Rewrite Storable.xs so that the file header structure for write\_magic is built at compile time, and check\_magic attempts to the header in blocks rather than byte per byte. These changes make the compiled extension 2.25% smaller, but are not significant enough to give a noticeable speed up.

[308](#)[309](#)[310](#)[311](#)[312](#)[313](#)

Thu May 23 22:50:41 BST 2002 Nicholas Clark &lt;nick@ccl4.org&gt;

[314](#)[315](#)

Version 2.01

[316](#)[317](#)

- New regression tests integer.t
- Add code to safely store large unsigned integers.
- Change code not to attempt to store large integers (ie > 32 bits) in network order as 32 bits.

[318](#)[319](#)[320](#)[321](#)[322](#)

*\*Never\** underestimate the value of a pathological test suite carefully crafted with maximum malice before writing a line of real code. It prevents crafty bugs from stowing away in your released code. It's much less embarrassing to find them before you ship. (Well, never underestimate it if you ever want to work for me)

[323](#)[324](#)[325](#)[326](#)[327](#)[328](#)

Fri May 17 22:48:59 BST 2002 Nicholas Clark &lt;nick@ccl4.org&gt;

[329](#)

Version 2.0, binary format 2.5 (but writes format 2.4 on pre 5.7.3)

[330](#)[331](#)[332](#)

The perl5 porters have decided to make sure that Storable still builds on pre-5.8 perls, and make the 5.8 version available on CPAN. The VERSION is now 2.0, and it passes all tests on 5.005\_03, 5.6.1 and 5.6.1 with threads. On 5.6.0 t/downgrade.t fails tests 34 and 35 due to a bug in 5.6.0 - upgrade to 5.6.1.

[333](#)[334](#)[335](#)[336](#)[337](#)[338](#)

Jarkko and I have collated the list of changes the perl5 porters have made from the perl5 Changes file:

[339](#)[340](#)[341](#)

- data features of upcoming perl 5.8.0 are supported: Unicode hash keys (Unicode hash values have been supported since Storable 1.0) and "restricted hashes" (readonly hashes and hash entries)
- a newer version of perl can now be used to serialize data which is not supported in earlier perls: Storable will attempt to do the right thing for as long as possible, croaking only when safe data conversion simply isn't possible. Alternatively earlier perls can opt to have a lossy downgrade data instead of croaking
- when built with perls pre 5.7.3 this Storable writes out files with binary format 2.4, the same format as Storable 1.0.8 onwards. This should mean that this Storable will inter-operate seamlessly with any Storable 1.0.8 or newer on perls pre 5.7.3
- dclone() now works with empty string scalar objects
- retrieving of large hashes is now more efficient
- more routines autosplit out of the main module, so Storable should load slightly more quickly
- better documentation
- the internal context objects are now freed explicitly, rather than relying on thread or process exit
- bugs fixed in debugging trace code affecting builds made with 64 bit IVs

[342](#)[343](#)[344](#)[345](#)[346](#)[347](#)[348](#)[349](#)[350](#)[351](#)[352](#)[353](#)[354](#)[355](#)[356](#)[357](#)[358](#)[359](#)[360](#)[361](#)

[366](#) Unix's file command about Storable files

[367](#)

[368](#) We plan to keep Storable on CPAN in sync with the Perl core, so

[369](#) if you encounter bugs or other problems building or using Storable,

[370](#) please let us know at perl5-porters@perl.org

[371](#) Patches welcome!

[372](#)

[373](#) Sat Dec 1 14:37:54 MET 2001 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[374](#)

[375](#) This is the LAST maintenance release of the Storable module.

[376](#) Indeed, Storable is now part of perl 5.8, and will be maintained

[377](#) as part of Perl. The CPAN module will remain available there

[378](#) for people running pre-5.8 perls.

[379](#)

[380](#) Avoid requiring Fcntl upfront, useful to embedded runtimes.

[381](#) Use an eval {} for testing, instead of making Storable.pm

[382](#) simply fail its compilation in the BEGIN block.

[383](#)

[384](#) store\_fd() will now correctly autoflush file if needed.

[385](#)

[386](#) Tue Aug 28 23:53:20 MEST 2001 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[387](#)

[388](#) Fixed truncation race with lock\_retrieve() in lock\_store().

[389](#) The file has to be truncated only once the exclusive lock is held.

[390](#)

[391](#) Removed spurious debugging messages in .xs file.

[392](#)

[393](#) Sun Jul 1 13:27:32 MEST 2001 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[394](#)

[395](#) Systematically use "=over 4" for POD linters.

[396](#) Apparently, POD linters are much stricter than would

[397](#) otherwise be needed, but that's OK.

[398](#)

[399](#) Fixed memory corruption on croaks during thaw(). Thanks

[400](#) to Claudio Garcia for reproducing this bug and providing the

[401](#) code to exercise it. Added test cases for this bug, adapted

[402](#) from Claudio's code.

[403](#)

[404](#) Made code compile cleanly with -Wall (from Jarkko Hietaniemi).

[405](#)

[406](#) Changed tagnum and classnum from I32 to IV in context. Also

[407](#) from Jarkko.

[408](#)

[409](#) Thu Mar 15 01:22:32 MET 2001 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[410](#)

[411](#) Last version was wrongly compiling with assertions on, due

[412](#) to an edit glitch. That did not cause any problem (apart from

[413](#) a slight performance loss) excepted on Win\* platforms, where the

[414](#) assertion code does not compile.

[415](#)

[416](#) Sat Feb 17 13:37:37 MET 2001 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[417](#)

[418](#) Version 1.0.10.

[419](#)

[420](#) Forgot to increase version number at previous patch (there were

[421](#) two of them, which is why we jump from 1.0.8 to 1.0.10).

[422](#)

[423](#) Sat Feb 17 13:35:00 MET 2001 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[428](#)

Now bless objects ASAP at retrieve time, which is meant to fix two bugs:

[429](#)[430](#)[431](#)[432](#)[433](#)[434](#)[435](#)[436](#)[437](#)[438](#)[439](#)[440](#)[441](#)[442](#)[443](#)[444](#)[445](#)[446](#)[447](#)[448](#)[449](#)[450](#)[451](#)[452](#)

Wed Jan 3 10:43:18 MET 2001 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[453](#)

Removed spurious 'clean' entry in Makefile.PL.

[454](#)[455](#)[456](#)[457](#)[458](#)[459](#)[460](#)[461](#)[462](#)[463](#)[464](#)[465](#)[466](#)[467](#)[468](#)[469](#)

Sun Nov 5 18:23:48 MET 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[470](#)

Version 1.0.6.

[471](#)[472](#)[473](#)[474](#)[475](#)[476](#)[477](#)[478](#)[479](#)

Thu Oct 26 19:14:38 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.com>

[480](#)

Version 1.0.5.

[481](#)[482](#)[483](#)[484](#)[485](#)

Documented that store() and retrieve() can return undef. That is, the error reporting is not always made via exceptions, as the paragraph on error reporting was implying.

Auto requires module of blessed ref when STORABLE\_thaw misses.

[490](#) Just check \$^O, in t/lock.t: there's no need to pull the whole  
[491](#) Config module for that.  
[492](#)

[493](#) Mon Oct 23 20:03:49 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[494](#)  
[495](#) Version 1.0.4.  
[496](#)  
[497](#) Protected calls to flock() for DOS platform: apparently, the  
[498](#) flock/fcntl emulation is reported to be broken on that  
[499](#) platform.  
[500](#)  
[501](#) Added logcarp emulation if they don't have Log::Agent, since  
[502](#) we now use it to carp when lock\_store/lock\_retrieve is used  
[503](#) on DOS.  
[504](#)

[505](#) Fri Sep 29 21:52:29 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[506](#)  
[507](#) Version 1.0.3.  
[508](#)  
[509](#) Avoid using "tainted" and "dirty" since Perl remaps them via  
[510](#) cpp (i.e. #define). This is deeply harmful when threading  
[511](#) is enabled. This concerned both the context structure and  
[512](#) local variable and argument names. Brrr..., scary!  
[513](#)

[514](#) Thu Sep 28 23:46:39 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[515](#)  
[516](#) Version 1.0.2.  
[517](#)  
[518](#) Fixed spelling in README.  
[519](#)  
[520](#) Added lock\_store, lock\_nstore, and lock\_retrieve (advisory locking)  
[521](#) after a proposal from Erik Haugan <erik@solbors.no>.  
[522](#)  
[523](#) Perls before 5.004\_04 lack newSVpvvn, added remapping in XS.  
[524](#)  
[525](#) Fixed stupid typo in the t/utf8.t test.  
[526](#)

[527](#) Sun Sep 17 18:51:10 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[528](#)  
[529](#) Version 1.0.1, binary format 2.3.  
[530](#)  
[531](#) Documented that doubles are stored stringified by nstore().  
[532](#)  
[533](#) Added Salvador Ortiz Garcia in CREDITS section, He identified  
[534](#) a bug in the store hooks and proposed the right fix: the class  
[535](#) id was allocated too soon. His bug case was also added to  
[536](#) the regression test suite.  
[537](#)  
[538](#) Now only taint retrieved data when source was tainted. A bug  
[539](#) discovered by Marc Lehmann.  
[540](#)  
[541](#) Added support for UTF-8 strings, a contribution of Marc Lehmann.  
[542](#) This is normally only activated in post-5.6 perls.  
[543](#)

[544](#) Thu Aug 31 23:06:06 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[545](#)  
[546](#) First official release Storable 1.0, for inclusion in perl 5.7.0.  
[547](#) The license scheme is now compatible with Perl's.



[552](#) The next version of Storable will be 0.8, and will be integrated  
[553](#) into the Perl core (development branch).  
[554](#)  
[555](#) Dispatch tables were moved upfront to relieve some compilers,  
[556](#) especially on AIX and Windows platforms.  
[557](#)  
[558](#) Merged 64-bit fixes from perl5-porters.  
[559](#)  
[560](#) Mon Aug 14 09:22:04 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[561](#)  
[562](#) Added a refcnt dec in retrieve\_tied\_key(): sv\_magic() increases  
[563](#) the refcnt on the mg\_ptr as well.  
[564](#)  
[565](#) Removed spurious dependency to Devel::Peek, which was used for  
[566](#) testing only in t/tied\_items.t. Thanks to Conrad Heiney  
[567](#) <conrad@fringehead.org> for spotting it first.  
[568](#)  
[569](#) Sun Aug 13 22:12:59 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[570](#)  
[571](#) Marc Lehmann kindly contributed code to add overloading support  
[572](#) and to handle references to tied variables.  
[573](#)  
[574](#) Rewrote leading blurb about compatibility to make it clearer what  
[575](#) "backward compatibility" is about: when I say 0.7 is backward  
[576](#) compatible with 0.6, it means the revision 0.7 can read files  
[577](#) produced by 0.6.  
[578](#)  
[579](#) Mention new Clone(3) extension in SEE ALSO.  
[580](#)  
[581](#) Was wrongly optimizing for "undef" values in hashes by not  
[582](#) fully recursing: as a result, tied "undef" values were incorrectly  
[583](#) serialized.  
[584](#)  
[585](#) Sun Jul 30 12:59:17 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[586](#)  
[587](#) First revision of Storable 0.7.  
[588](#)  
[589](#) The serializing format is new, known as version 2.0. It is fully  
[590](#) backward compatible with 0.6. Earlier formats are deprecated and  
[591](#) have not even been tested: next version will drop pre-0.6 format.  
[592](#)  
[593](#) Changes since 0.6@11:  
[594](#)  
[595](#) - Moved interface to the "beta" status. Some tiny parts are still  
[596](#) subject to change, but nothing important enough to warrant an "a  
[597](#) status any longer.  
[598](#)  
[599](#) - Slightly reduced the size of the Storable image by factorizing  
[600](#) object class names and removing final object storage notification  
[601](#) to a redesign of the blessed object storing.  
[602](#)  
[603](#) - Classes can now redefine how they wish their instances to be ser  
[604](#) and/or deep cloned. Serializing hooks are written in Perl code.  
[605](#)  
[606](#) - The engine is now fully re-entrant.  
[607](#)  
[608](#) Sun Apr 2 23:47:50 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co  
[609](#)

[614](#)[615](#)[616](#)[617](#)[618](#)

I'm expecting this revision to be the last of the 0.6 series. Unless it does not work with perl 5.6, which I don't use yet, and therefore against which I cannot test.

[619](#)

Wed Mar 29 19:55:21 MEST 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co

[620](#)[621](#)

Added note about format incompatibilities with old versions (i.e. pre 0.5@9 formats, which cannot be understood as there was no versioning information in the file by then).

[622](#)[623](#)[624](#)[625](#)

Protect all \$@ variables when eval {} used, to avoid corrupting it when store/retrieve is called within an exception handler.

[626](#)[627](#)[628](#)[629](#)

Mistakenly included "patchlevel.h" instead of <patchlevel.h>, preventing Perl's patchlevel from being included, which is needed starting from 5.6.

[630](#)[631](#)[632](#)

Tue May 12 09:15:15 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenob

[633](#)[634](#)

Fixed shared "undef" bug in hashes, which did not remain shared through store/retrieve.

[635](#)[636](#)[637](#)

Thu Feb 10 19:48:16 MET 2000 Raphael Manfredi <Raphael\_Manfredi@pobox.co

[638](#)[639](#)

added last\_op\_in\_netorder() predicate  
documented last\_op\_in\_netorder()  
added tests for the new last\_op\_in\_netorder() predicate

[640](#)[641](#)[642](#)[643](#)

Wed Oct 20 19:07:36 MEST 1999 Raphael Manfredi <Raphael\_Manfredi@pobox.co

[644](#)[645](#)

Forgot to update VERSION

[646](#)[647](#)

Tue Oct 19 21:25:02 MEST 1999 Raphael Manfredi <Raphael\_Manfredi@pobox.co

[648](#)[649](#)

Added mention of japanese translation for the manual page.

[650](#)[651](#)

Fixed typo in macro that made threaded code not compilable, especially on Win32 platforms.

[652](#)[653](#)[654](#)

Changed detection of older perls (pre-5.005) by testing PATCHLEVEL directly instead of relying on internal symbols.

[655](#)[656](#)

Tue Sep 14 22:13:28 MEST 1999 Raphael Manfredi <Raphael\_Manfredi@pobox.co

[657](#)[658](#)

Integrated "thread-safe" patch from Murray Nesbitt.  
Note that this may not be very efficient for threaded code, see comment in the code.

[660](#)[661](#)[662](#)[663](#)

Try to avoid compilation warning on 64-bit CPUs. Can't test it, since I don't have access to such machines.

[664](#)[665](#)

Mon Jul 12 14:37:19 METDST 1999 Raphael Manfredi <Raphael\_Manfredi@pobox

[667](#)[668](#)

changed my e-mail to pobox.

[669](#)[670](#)

mentioned it is not thread-safe.

[671](#)



[676](#) Fri Jul 3 13:38:16 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenol  
[677](#)  
[678](#) Updated benchmark figures due to recent optimizations done in  
[679](#) store(): tagnums are now stored as-is in the hash table, so  
[680](#) no surrounding SV is created. And the "shared keys" mode for  
[681](#) hash table was turned off.  
[682](#)  
[683](#) Fixed backward compatibility (wrt 0.5@9) for retrieval of  
[684](#) blessed refs. That old version did something wrong, but the  
[685](#) bugfix prevented correct retrieval of the old format.  
[686](#)  
[687](#) Mon Jun 22 11:00:48 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenol  
[688](#)  
[689](#) Changed benchmark figures.  
[690](#)  
[691](#) Adjust refcnt of tied objects after calling sv\_magic() to avoid  
[692](#) memory leaks. Contributed by Jeff Gresham.  
[693](#)  
[694](#) Fri Jun 12 11:50:04 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenol  
[695](#)  
[696](#) Added workaround for persistent LVALUE-ness in perl5.004. All  
[697](#) scalars tagged as being an lvalue are handled as if they were  
[698](#) not an lvalue at all. Added test for that LVALUE bug workaround.  
[699](#)  
[700](#) Now handles Perl immortal scalars explicitly, by storing &sv\_yes  
[701](#) as such, explicitly.  
[702](#)  
[703](#) Retrieval of non-immortal undef cannot be shared. Previous  
[704](#) version was over-optimizing by not creating a separate SV for  
[705](#) all undefined scalars seen.  
[706](#)  
[707](#) Thu Jun 4 17:21:51 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenol  
[708](#)  
[709](#) Baseline for Storable-0.6@0.  
[710](#)  
[711](#) This version introduces a binary incompatibility in the generated  
[712](#) binary image, which is more compact than older ones by approximativ  
[713](#) 15%, depending on the exact degree of sharing in your structures.  
[714](#)  
[715](#) The good news is that your older images can still be retrieved with  
[716](#) this version, i.e. backward compatibility is preserved. This versio  
[717](#) of Storable can only generate new binaries however.  
[718](#)  
[719](#) Another good news is that the retrieval of data structure is  
[720](#) significantly quicker than before, because a Perl array is used  
[721](#) instead of a hash table to keep track of retrieved objects, and  
[722](#) also because the image being smaller, less I/O function calls are  
[723](#) made.  
[724](#)  
[725](#) Tue May 12 09:15:15 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenol  
[726](#)  
[727](#) Version number now got from Storable.pm directly.  
[728](#)  
[729](#) Fixed overzealous sv\_type() optimization, which would make  
[730](#) Storable fail when faced with an "upgraded" SV to the PVIV  
[731](#) or PVNV kind containing a reference.  
[732](#)  
[733](#) Thu Apr 30 15:11:30 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenol



[738](#) Optimized `sv_type()` to avoid flags checking when not needed, i.e. when their type makes it impossible for them to be refs or tied. This slightly increases throughput by a few percents when refs and tied variables are marginal occurrences in your data.

[739](#)

[740](#)

[741](#)

[742](#)

[743](#) Stubs for XS now use `OutputStream` and `InputStream` file types to make it work when the given file is actually a socket. Perl makes a distinction for sockets in its internal I/O structures by having both a read and a write structure, whereas plain files share the same one.

[744](#)

[745](#)

[746](#)

[747](#)

[748](#)

[749](#) Tue Jun 3 09:41:33 METDST 1997 Raphael Manfredi <Raphael\_Manfredi@grenoble

[750](#)

[751](#) Thanks to a contribution from Benjamin A. Holzman, `Storable` is now able to correctly serialize tied SVs, i.e. tied arrays, hashes and scalars.

[752](#)

[753](#)

[754](#)

[755](#) Thu Apr 9 18:07:51 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenoble

[756](#)

[757](#) I said `SvPOK()` had changed to `SvPOKp()`, but that was a lie...

[758](#)

[759](#) Wed Apr 8 13:14:29 METDST 1998 Raphael Manfredi <Raphael\_Manfredi@grenoble

[760](#)

[761](#) Wrote `sizeof(SV *)` instead of `sizeof(I32)` when portable, which in effect mangled the object tags and prevented portability across 32/64 bit architectures!

[762](#)

[763](#)

[764](#)

[765](#) Wed Mar 25 14:57:02 MET 1998 Raphael Manfredi <Raphael\_Manfredi@grenoble

[766](#)

[767](#) Added code example for `store_fd()` and `retrieve_fd()` in the man page, to emphasize that file descriptors must be passed as globs, not as plain strings.

[768](#)

[769](#)

[770](#)

[771](#) Cannot use SV addresses as tag when using `nstore()` on LP64. This was the cause of problems when creating a storable image on an LP64 machine and retrieving it on an ILP32 system, which is exactly what `nstore()` is meant for...

[772](#)

[773](#)

[774](#)

[775](#)

[776](#) However, we continue to use SV addresses as tags for plain `store()`, because benchmarking shows that it saves up to 8% of the store time, and `store()` is meant to be fast at the expense of lack of portability.

[777](#)

[778](#)

[779](#)

[780](#)

[781](#) This means there will be approximately an 8% degradation of performance for `nstore()`, but it's now working as expected. That cost may vary on your machine of course, since it is solely caused by the memory allocation overhead used to create unique SV tags for each distinct stored SV.

[782](#)

[783](#)

[784](#)

[785](#)

[786](#)

[787](#) Tue Jan 20 09:21:53 MET 1998 Raphael Manfredi <Raphael\_Manfredi@grenoble

[788](#)

[789](#) Don't use any `'_'` in version number.

[790](#)

[791](#) Tue Jan 13 17:51:50 MET 1998 Raphael Manfredi <Raphael\_Manfredi@grenoble

[792](#)

[793](#) Updated version number.

[794](#)

[795](#) added `binmode()` calls for systems where it matters.

[800](#) Wed Nov 5 10:53:22 MET 1997 Raphael Manfredi <Raphael\_Manfredi@grenoble  
[801](#) Fix memory leaks on seen hash table and returned SV refs.  
[802](#)  
[803](#) Storable did not work properly when tainting enabled.  
[804](#)  
[805](#) Fixed "Allocation too large" messages in freeze/thaw and added.  
[806](#) proper regression test in t/freeze.t.  
[807](#)

[808](#) Tue Jun 3 09:41:33 METDST 1997 Raphael Manfredi <Raphael\_Manfredi@grenob  
[809](#)  
[810](#) Updated version number  
[811](#)  
[812](#) Added freeze/thaw interface and dclone.  
[813](#)

[814](#) Fri May 16 10:45:47 METDST 1997 Raphael Manfredi <Raphael\_Manfredi@grenob  
[815](#)  
[816](#) Forgot that AutoLoader does not export its own AUTOLOAD.  
[817](#) I could use  
[818](#)  
[819](#) use AutoLoader 'AUTOLOAD';  
[820](#)  
[821](#) but that would not be backward compatible. So the export is  
[822](#) done by hand...  
[823](#)

[824](#) Tue Mar 25 11:21:32 MET 1997 Raphael Manfredi <Raphael\_Manfredi@grenoble  
[825](#)  
[826](#) Empty scalar strings are now "defined" at retrieval time.  
[827](#)  
[828](#) New test to ensure an empty string is defined when retrieved.  
[829](#)

[830](#) Thu Feb 27 16:32:44 MET 1997 Raphael Manfredi <Raphael\_Manfredi@grenoble  
[831](#)  
[832](#) Updated version number  
[833](#)  
[834](#) Declare VERSION as being used  
[835](#)  
[836](#) Fixed a typo in the PerlIO\_putc remapping.  
[837](#) PerlIO\_read and perlIO\_write inverted size/nb\_items.  
[838](#) (only relevant for pre-perl5.004 versions)  
[839](#)

[840](#) Thu Feb 27 15:58:31 MET 1997 Raphael Manfredi <Raphael\_Manfredi@grenoble  
[841](#)  
[842](#) Updated version number  
[843](#)  
[844](#) Added VERSION identification  
[845](#)  
[846](#) Allow build with perl5.003, which is ante perlIO time  
[847](#)

[848](#) Mon Jan 13 17:53:18 MET 1997 Raphael Manfredi <Raphael\_Manfredi@grenoble  
[849](#)  
[850](#) Random code fixes.  
[851](#)

[852](#) Wed Jan 22 15:19:56 MET 1997 Raphael Manfredi <Raphael\_Manfredi@grenoble  
[853](#)  
[854](#) Updated version number in Makefile.PL.  
[855](#)  
[856](#) Added "thanks to" section to README.  
[857](#)

[861](#)[862](#)[863](#)[864](#)

Added `forgive_me` support to allow `store()` of data structures containing non-storable items like `CODE` refs.