

Kerbal Space Program - Bug #9723

Opening list of saves is extremely slow

05/12/2016 09:02 PM - gamerprogrammer

Status:	Closed	Start date:	05/12/2016
Severity:	Low	% Done:	100%
Assignee:			
Category:	Application		
Target version:	1.2.2		
Version:	1.1.2	Language:	English (US)
Platform:	Linux, OSX, Windows	Mod Related:	No
Expansion:			
Description			
<ol style="list-style-type: none">1. Have 119 saves2. Launch KSP3. Load profile4. You will see KSP5. Open menu6. Select "Load Save..."7. Wait 43 seconds (this is really annoying)8. You will see list of saves (with little ammount of information - there is nothing to load that long)			
Expected result:			
<ol style="list-style-type: none">7. Immediate opening			
This was before I have 100 saves. Opening in flight with ALT+F9 works pretty good (it opens immediately). Consider optimization of loading saves and caching results.			
Environment:			
KSP: 1.1.2 (Unix) - Unity: 5.2.4f1 - OS: Mac OS X 10.11.4			
Filter Extensions - 2.5			
USI Tools - 0.7.2.1			
B9 Part Switch - 1.0.1			
Community Resource Pack - 0.5.1			
Firespitter - 7.2.1			
Interstellar Fuel Switch - 1.27			
RasterPropMonitor - 0.25.1			
KAS - 0.5.6.7			
KIS - 1.2.7.6			
KSP-AVC Plugin - 1.1.6.1			
Kerbal Joint Reinforcement - 3.1.6			
HyperEdit - 1.5			
NearFutureConstruction - 0.6			
NearFutureElectrical - 0.7			
NearFuturePropulsion - 0.7			
NearFutureSolar - 0.6			
NearFutureSpacecraft - 0.4.4			
Kerbal Planetary Base Systems - 1.0.9			
Procedural Parts - 1.2			
DeepFreeze Continued... - 0.22.1			
RCS Sounds - 5.0			
SCANSat - 1.1.6			
Interstellar Fuel Switch - 1.23			
SpaceY Expanded - 1.1.8			
SpaceY Lifters - 1.12.2			
Kerbal Alarm Clock - 3.6.1			
TweakScale - 2.2.7.2			
USI Core - 0.2.2.1			
USI Exploration Pack - 0.5.2.1			

Karbonite - 0.7.2.1
Karbonite Plus - 0.6.2.1
Karibou - 0.2.2.1
Kolonization Core - 0.2.2.1
USI-LS - 0.4.2.1
USI Survivability Pack - 0.5.2.1
UKS - 0.40.2.1
USI Alcubierre Drive - 0.4.2.1
KSP Interstellar Extended - 1.8.12

History

#1 - 05/29/2016 04:52 AM - gamerprogrammer

This doesn't rely on count of saves, but on total size of saves. E.g. I have 200 saves, each of 12MB size (but if I simply compress 12MB file with ZIP, it takes less than 1MB space). So I don't use the menu from KSC. As a workaround I made a vessel near KSC, that is immediately visible when I load my persistent.sfs save. When I load the game I switch to that vessel and open menu with Mod+F9.

I suggest you to not parse the whole file when loading just amount of funds, science, and other info.

I also suggest you to compress save files. Compressing 1.35GB saves folder took 5 seconds on my computer and gave 75MB zip file (just with standard ZIP utility). However, 1.35GB is too little space in 2016, so you can ignore this suggestion freely.

But optimization of loading saves' list must be implemented! Now I just do not use this feature because of lags.

#2 - 07/17/2016 09:56 AM - TriggerAu

- Status changed from New to Needs Clarification

#3 - 08/15/2016 02:28 AM - warwick

I get the same experience and am happy to provide any clarification required. It basically looks like the "Load Save..." dialog (weird button name, by the way) loads the whole SFS file (in order to give the extra info for that dialog), whereas the Alt-F9 dialog just lists the filenames (and so pops up immediately).

Even with just a few dozen save files, using "Load Save..." is painfully slow.

#4 - 11/17/2016 03:18 PM - Padishar

- Status changed from Needs Clarification to Updated

- % Done changed from 0 to 10

This is even worse in 1.2.x as Alt-F9 now opens the same slow dialog.

Basically, the dialog is currently loading the complete game state from every save just to work out a few things for the load dialog. It is doing this because the info it uses is currently spread through the file. Caching the values for each file would be one option but it would still need to do the work the first time and would need to be careful to detect changes to the files. I think it would be better to modify the save format to include a copy of all the required info in a new node near the start of the file (e.g. a SAVEINFO node as the first child node of the GAME node) and then extract it for the dialog using an optimised routine. This would also remove the huge amount of garbage churn that occurs. The whole file is currently loaded into a string[] and then processed into a List<string[]> and then that's used to build the tree of ConfigNode objects, each of which contains a List<ConfigNode> containing the child nodes and a List<ConfigNode.Value> that contains an object for every value in the node. This is a lot of objects/memory even for a single save file, but when repeated for a couple of hundred files it must be enormous.

The new routine would only ever need to load a small chunk of the start of the file (1 or 2k would be more than enough for the data that's currently needed even including all the values in the GAME node) and it could also reuse the same buffer for each file. It could search the buffer for the SAVEINFO and then for the closing brace at the end of it and pass just that part to a ConfigNode for parsing. An alternative could be to put the values directly in the GAME node and search the buffer for the PARAMETERS node and terminate the GAME node at that point for fast parsing.

#5 - 11/21/2016 09:23 AM - Padishar

- Platform Linux, Windows added

#7 - 12/07/2016 03:53 AM - JPLRepo

- Status changed from Updated to Ready to Test

- Target version set to 1.2.2

- % Done changed from 10 to 80

This should be fixed in 1.2.2. Please test.

#8 - 02/23/2017 10:26 PM - Firzen

It's sadly definitely not yet fixed in 1.2.2 Linux. At least with saves imported from previous version. Loading list of saves still takes ages, depending on total size of them as mentioned.

#9 - 06/22/2017 08:37 PM - Squelch

- *Status changed from Ready to Test to Resolved*
- *% Done changed from 80 to 100*

#10 - 06/22/2017 08:39 PM - Squelch

- *Status changed from Resolved to Closed*

#11 - 06/28/2017 06:12 PM - gamerprogrammer

God, I didn't think that you fixed this issue. KSP team is very cool and was always cool. Sorry for my offtopic.