

## Database - Bug #4057

### h2 performance degradation

05/01/2019 10:42 AM - Greg Shah

<b>Status:</b> New	<b>Start date:</b>
<b>Priority:</b> Normal	<b>Due date:</b>
<b>Assignee:</b> Stanislav Lomany	<b>% Done:</b> 0%
<b>Category:</b>	<b>Estimated time:</b> 0.00 hour
<b>Target version:</b>	<b>case_num:</b>
<b>billable:</b> No	
<b>vendor_id:</b> GCD	
<b>Description</b>	
<b>Related issues:</b>	
Related to Database - Support #4702: write temp-table performance test cases	<b>Closed</b>
Related to Database - Support #4701: try to improve H2 transaction commit per...	<b>Closed</b>
Related to Database - Support #6679: H2 general performance tuning	<b>Internal Test</b>

### History

#1 - 05/01/2019 10:43 AM - Greg Shah

- File h2a.png added

From the H2 mailing list:

----- Forwarded Message -----

Subject: [h2] Re: Recent drops in performance

Date: Fri, 26 Apr 2019 07:15:34 -0700 (PDT)

From: Christian MICHON ...

Reply-To: [h2-database@googlegroups.com](mailto:h2-database@googlegroups.com)

To: H2 Database <[h2-database@googlegroups.com](mailto:h2-database@googlegroups.com)>

On Thursday, March 28, 2019 at 8:40:31 AM UTC+1, Evgenij Ryazanov wrote:

Hello.

Could you build a standalone test case that can be used to reproduce such slowdown? I don't see any reasons for it.

Hi,

@Joe: based on your comments, I've built a standalone performance test using JMeter. Not only you were right, but the situation has been around for some time since 1.4.197, maybe even before.

@Evgenij : it's not rocket science to build such scenario, if you need such performance test scenario, I'll share it on github (it's not confidential).

I'm surprised no one ever measured H2 performance in a statistical manner. I'm even more surprised by the results...

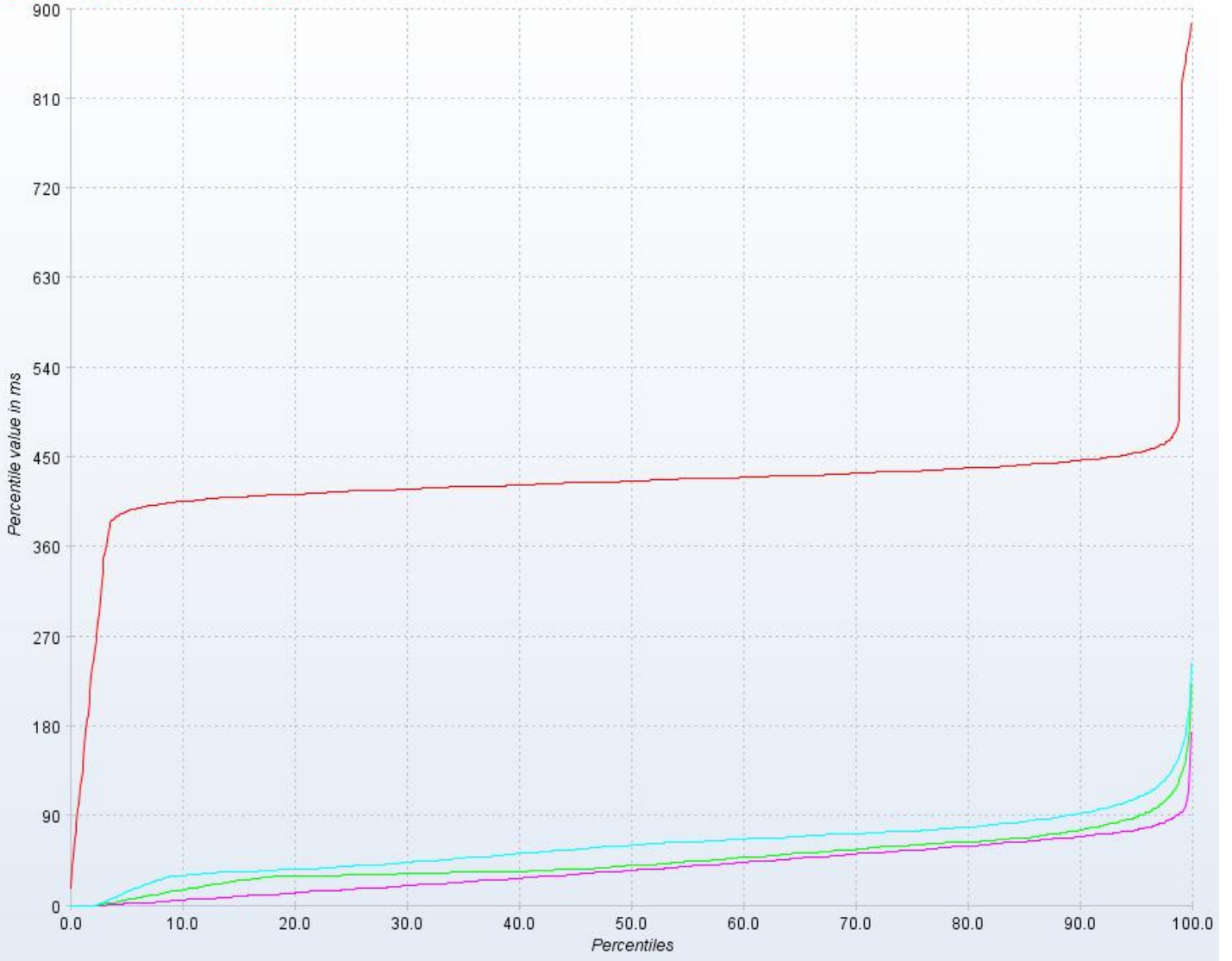
On these releases 1.2.147, 1.3.176, 1.4.196 and 1.4.199, the following scenario ran for 200 virtual users in a memory-only database: perform a complex CTE 1000 times on a indexed taxonomy database, with parametrized input.

This clearly shows a drop not only on latest 1.4.199, but the drop became substantial from 1.4.197 onward.

Below is a percentile graph showing that 1.2.147 (almost 9 years old!) is still the fastest H2 engine on a pure memory database... Followed by 1.3.176, then 1.4.196. Versions from 1.4.197 to 1.4.199 really came up with a new performance issue which can be felt by any of the 200 virtual users.

What dramatically changed between 1.4.196 and 1.4.197?

1.2.147 1.3.176 1.4.196 1.4.199



--  
You received this message because you are subscribed to the Google Groups "H2 Database" group.  
To unsubscribe from this group and stop receiving emails from it, send an email to [h2-database+unsubscribe@googlegroups.com](mailto:h2-database+unsubscribe@googlegroups.com).  
To post to this group, send email to [h2-database@googlegroups.com](mailto:h2-database@googlegroups.com).  
Visit this group at <https://groups.google.com/group/h2-database>.  
For more options, visit <https://groups.google.com/d/optout>.

**#2 - 05/01/2019 10:44 AM - Greg Shah**

From Eric:

We're on (a customized) 1.4.197 (not depicted, but probably somewhere between cyan and red).

**#3 - 05/01/2019 10:45 AM - Greg Shah**

- File *h2c.png* added

More from the mailing list:

----- Forwarded Message -----

Subject: [h2] Re: Recent drops in performance

Date: Fri, 26 Apr 2019 08:19:32 -0700 (PDT)

From: Christian MICHON ...

Reply-To: [h2-database@googlegroups.com](mailto:h2-database@googlegroups.com)

To: H2 Database <[h2-database@googlegroups.com](mailto:h2-database@googlegroups.com)>

On Friday, April 26, 2019 at 4:28:00 PM UTC+2, Evgenij Ryazanov wrote:

Hello.

CTEs are experimental in H2 and have known bugs, recent versions contain some fixes, so this performance drop may be somehow related with them. But older versions of H2 do not support CTEs at all, so I don't understand how you can test a feature that does not exist.

We still need a complete standalone test case for your issue.

CTE was working with v1.2.x from what I recall and tested. I grant that CTE is overkill, but I needed the test to be compute intensive, to test the engine itself.

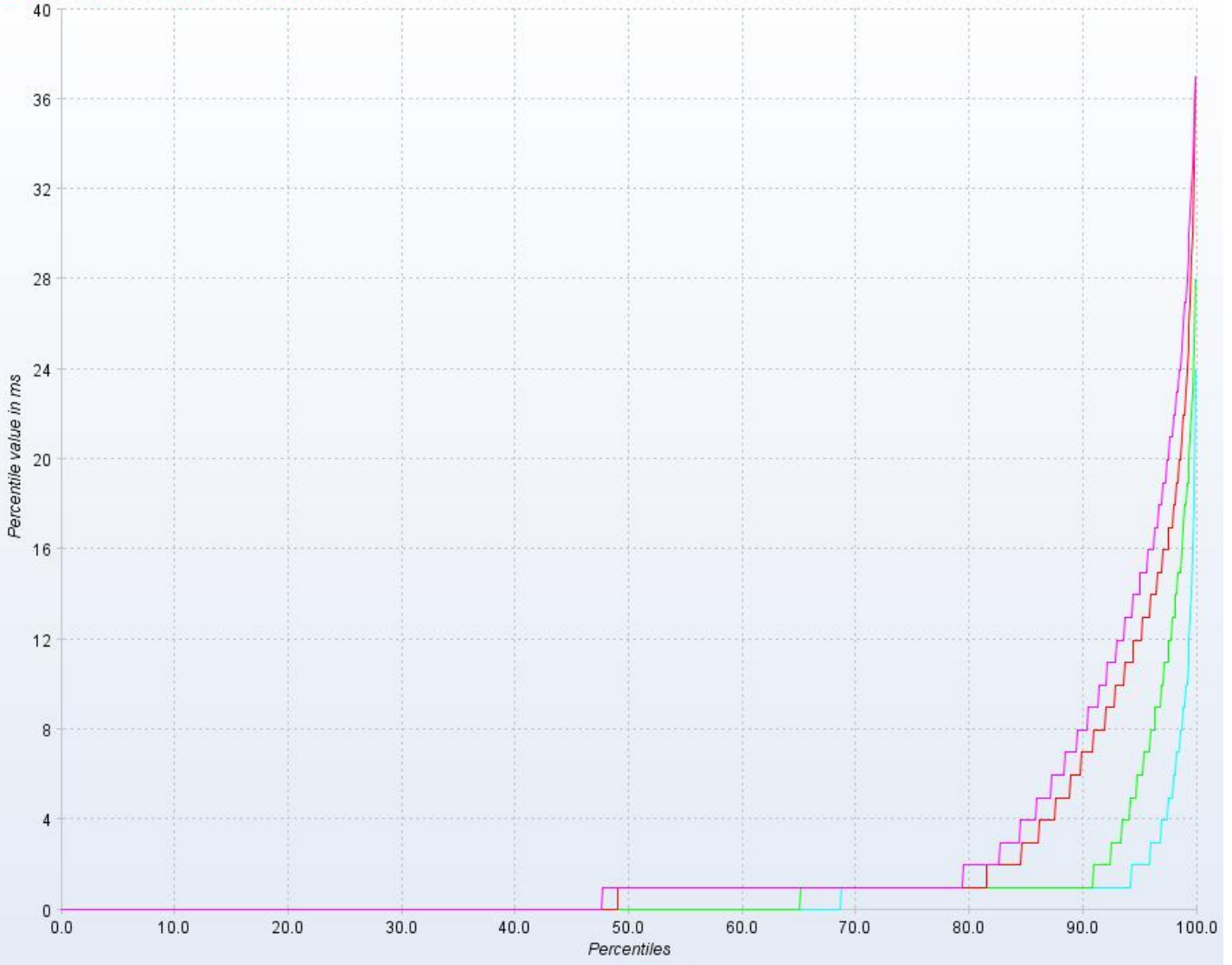
What I re-created just now is purely based on DDL:

```
DROP ALL OBJECTS;  
CREATE TABLE TAXNUM ( TSN CHAR PRIMARY KEY NOT NULL, NAME VARCHAR NOT NULL, "TYPE" VARCHAR NOT NULL, PARENT VARCHAR);
```

I added some pauses ranging from 50ms to 100ms (we call them thinktime in performance testing) and I did only 4 versions, with similar load as described in my earlier email.

Here's the result: 1.4.197 is still slower. 1.4.198 went into errors as I believe the thinktimes are not long enough. I'll fix that.

1.2.147 1.3.176 1.4.196 1.4.197



--

You received this message because you are subscribed to the Google Groups "H2 Database" group.  
To unsubscribe from this group and stop receiving emails from it, send an email to [h2-database+unsubscribe@googlegroups.com](mailto:h2-database+unsubscribe@googlegroups.com).  
To post to this group, send email to [h2-database@googlegroups.com](mailto:h2-database@googlegroups.com).  
Visit this group at <https://groups.google.com/group/h2-database>.  
For more options, visit <https://groups.google.com/d/optout>.

**#4 - 05/01/2019 10:46 AM - Greg Shah**

As an interim measure, can we go back to 1.4.196 (and add our customizations)?

**#5 - 05/01/2019 10:46 AM - Greg Shah**

From Eric:

I'd rather not. 197 has a lot of fixes for regressions in the last few point releases before it.

This is one data point; I'm not sure how accurately it reflects our use cases. Also, it's not clear whether he was using the newer storage engine (mv\_store) in his testing of later releases (which is the default in those releases), or the older (still faster) page store. We're still using page store, just because it was noticeably faster in my testing.

**#6 - 06/29/2020 05:12 PM - Eric Faulhaber**

- Related to Support #4702: write temp-table performance test cases added

**#7 - 06/29/2020 07:07 PM - Eric Faulhaber**

- Related to Support #4701: try to improve H2 transaction commit performance added

**#8 - 06/29/2020 07:16 PM - Eric Faulhaber**

- Assignee set to Stanislav Lomany

Please get the source for the latest version of H2 (1.4.200 at the time of this writing) and port our existing patches to it (see [Building H2](#)). Constantin has a recent patch which is not integrated into the FWD distribution yet; you will need to get that from him.

Using the test cases written for [#4702](#) and branch 4011a\_rebase (if ready) or 4011a (until 4011\_rebase is ready), stress test and profile both customized builds of 1.4.197 and 1.4.200 to see if one performs noticeably better than the other. Report your results here. If there are any functional differences or bugs in either version, please report that as well.

**#9 - 06/29/2020 08:22 PM - Constantin Asofiei**

Eric Faulhaber wrote:

Constantin has a recent patch which is not integrated into the FWD distribution yet; you will need to get that from him.

See [#4011-415](#) for the patch.

**#10 - 07/02/2020 04:01 PM - Adrian Lungu**

I've done a testing effort in comparing 1.4.196, 1.4.197, 1.4.197 patched, 1.4.198 and 1.4.199. I shared the results in [#4701-6](#). It was part of my attempt to see any radical change in regard with transaction commits between versions. I will make use of Apache JMeter and the testing plans I used there to include 1.4.200 in the testing process.

**#11 - 07/08/2020 02:39 PM - Eric Faulhaber**

Stas, when do you think you will have findings, based on the [#4702](#) test cases? Based on the comparison of versions in [#4701](#), I'd like to move to H2 1.4.200, if your results are not contradictory. If they are, we'll need to discuss.

**#12 - 08/17/2022 10:57 AM - Alexandru Lungu**

- Related to Support #6679: H2 general performance tuning added

**#13 - 01/23/2024 04:31 PM - Eric Faulhaber**

Can we close this task?

**Files**

---

h2a.png	27 KB	05/01/2019	Greg Shah
h2c.png	30.8 KB	05/01/2019	Greg Shah