

OrientDB Leaks

True experiences about OrientDB

Saturday, June 20, 2015

Why you should avoid OrientDB

Up until now I've been a silent bystander regarding constant OrientDB issues but recent events changed my mind and I've decided to intervene so others can learn from my experience.

TL;DR: OrientDB is like a book with a great cover and some interesting first chapters but once you reach the middle of the book you realize you've been mistaken and it's not all it promises to be. If you don't want to read the whole story scroll down to the conclusion. (if you don't read the whole post please refrain from posting comments).

3rd quarter of 2014: decision to use OrientDB

In the end of Q3 2014 my team agreed a graphDB would be a perfect fit for a new project and we started evaluating options. OrientDB came up on top in terms of features and because of its permissive license. Investigating a little bit deeper revealed a few stability issues (for example: [Continual frustration with OrientDB durability](#) and ["With 1.7.x, they \[issues\] have been mostly around stability"](#)) but we thought v2.0 would fix all these issues... We've done a proof of concept, all looked good and we jumped with both feet adopting OrientDB. This would be a decision we would later regret. Below is a timeline of events and issues that changed our mind about the product.

February: several small OrientDB issues

5/6 months after we began using OrientDB we started realizing that stability/robustness problems subsided. By now we had reported several issues (in years of experience with mainstream SQL DBs I never had to report a single issue!). None of the issues was huge but small bugs like [LIMIT required with ORDER BY and SKIP #2743](#) (still unresolved after 10 months) made us waste countless hours debugging OrientDB.

February 17-22: OrientDB CEO accuses competitor of not implementing a GraphDB

One could expect a company's CEO to have character and show *fair play* towards other competitors. What one wouldn't expect is a GraphDB creator going to StackOverflow and accusing their competitor's solution of not implementing a GraphDB but that did happen twice in [What factors to consider when choosing a Multi-model DBMS?](#):

I'm the founder of OrientDB (so it's biased) and I can say that OrientDB is a multi-model DBMS at the engine level, while ArangoDB and FoundationDB just implemented layers on top of it. It's like using Hibernate on top of Oracle thinking that you have a ODBMS. My 0,02. – [Lvca](#) Feb 17 at 20:41

2 I'm the other founder of ArangoDB and therefore also biased. Luca, I fear you got this one wrong: ArangoDB is designed and built as a multi-model DB from its inception. All three data models together with complete support in the API and query language are implemented in the DB engine as first class citizens and with high-performance C++ code. It is untrue that graphs and key/value are only implemented as layers on top of the document store. – [weinberger](#) Feb 18 at 1:17

Hey weinberger, a GraphDB is defined as "index-free adjacency". ArangoDB uses a Hash Index to cross relationship, so it's not "index-free", sorry. It's rather a JOIN. It's like traversing relationships with a RDBMS. – [Lvca](#) Feb 19 at 0:43

2 Who made up this definition? This is more marketing than anything else. When we perform a traversal, it's not a JOIN. What you wrote it not true. [j.mp/1EOt8gk](#) – [weinberger](#) Feb 25 at 5:31

March 22: allegedly 3 companies drop OrientDB

The source of this comes from a Hacker News Article <https://news.ycombinator.com/item?id=9253488>, where anonwarnings anonymously reveals some OrientDB problems. Some highlights are:

anonwarnings: OrientDB is snakeoil and anyone deploying it has not done their due diligence. The real world performance is very different to their claims and many features just don't work. Do yourself a favor and read through their bug tracker some time, you'll be scared.

Ivca (CEO of Orient Technologies): An account created 3 hours ago with just this comment? Smell like a troll.

anonwarnings: We spent a lot of time and money trying to get OrientDB working and we ran into so many problems that it was impossible for us to continue with it. It is buggy beyond belief, distributed mode is deeply flawed, transactions are not transactional, it's super easy to hang the database and the explain keyword straight up lies, the backup strategy is non-existent, the documentation incredibly poor and riddled with typos and outright mistakes. When we finally thought we'd got over these issues, we imported our data and the system ground to a halt, just weeks before a major product launch. [...]

Also, look at this list of issues closed as "invalid/wontfix", it seems like you're a lot keener to close issues than to actually fix them - <https://github.com/orienttechnologies/orientdb/issues?utf8=%E2%9C%93&q=is%3Aissue+label%3Ainvalid>

Ivca: Ok, I understand who you are, but it's not fair saying the problems was on OrientDB product: we have

About Me



OrientDB Leaks

a disgruntled OrientDB user

[View my complete profile](#)

Blog Archive

▼ 2015 (4)

► July (1)

▼ June (3)

[OrientDB's lack of transparency](#)

[Why you should avoid OrientDB](#)

[About OrientDB Leaks](#)

Twitter

Follow [@orientdb_leaks](#)

Labels

- [Transparency](#) (3)
- [OrientDB Issues](#) (1)

a lot of clients running in production without such problems. [...]
Ok, so at this point you wasn't a client and you never hire a real expert?

anonwarnings: I think you're more interested in finding out who I am or who I work for and trying to discredit my claims than you are with addressing the issues I raised [...]
we had a very bad experience with OrientDB and we weren't doing anything out of the ordinary with it. We didn't even really use the graph features much beyond very simple traversals. The consultant that we hired to try and fix the mess was unable to do so, and he recently told me about 2 other companies that had a basically identical experience to us. The thing is just not ready for production, and for small startups making this kind of mistake can be fatal, that's the only reason I'm saying something.

Key takeaway: while anonwarnings is coming publicly with the issues his company experienced with OrientDB, the OrientDB CEO focus in finding out who he is instead of addressing the issues.

March 27: OrientDB distributed instability

In the google group post [Stable version of OrientDb for distributed environment?](#) a couple of users claim OrientDB v2 has instability issues in distributed setup.

April 3: OrientDB transactions broken

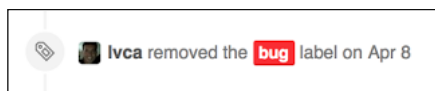
Transactions are a feature that not everyone uses but for those who do use it's usually a mission critical feature that must be reliable. If you are dealing with a financial transaction you wouldn't want to mark a bill as paid if the payment failed or vice-versa. On OrientDB v2.0.6 a bug was reported where transactions failed over the binary transport [Transactional issue in the binary driver \(detected in 'oriento'\) #3868](#). This was later fixed in v2.0.7 but once again our confidence in OrientDB was shaken.

April 8: fixing OrientDB bugs (by hiding them)

I regularly track OrientDB updates and on this day I find that around 150 issues had been updated, an excerpt below:

❶	Invalid keyword error in UPDATE SET clause with JSON	#3273 updated on Apr 8
❷	mismatch size number: out('A').out('B') and out('A').out('B').size()	#2448 updated on Apr 8
❸	no way to return Record from Function.	#2283 updated on Apr 8
❹	(Client=55) console.sh "set echo true" doesn't work	#2588 updated on Apr 8
❺	Query Embedded type is very slow for large dataset	#2256 updated on Apr 8 2.2
❻	Mistake in the RDBMS Wiki	#2666 updated on Apr 8

I'm curious and start clicking on dozens of these issues to see what was modified and what do I find?



Yeap, I didn't go through all of them but all the issues I clicked had their bug label removed. You can check by yourself, here are a few: [#2462](#), [#2517](#), [#2768](#), [#2783](#), [#1254](#), [#1145](#), [#2120](#), [#1552](#) for your perusal.

Key takeaway: the OrientDB CEO is willing to hide bugs to make things look better than they really are.

April 9: making OrientDB bugs hard to spot



By default GitHub projects have a bug label that has a bright red background and most project owners don't change this, or use another bright/contrasty color so they stand out among other issues. The rationale is that known bugs are meant to be... known. Any user should be able to spot these easily to avoid finding them during development or worse, production. OrientDB folks contrary to the norm prefer to use a light grey background which is almost indistinguishable from the GitHub's white background.

Key takeaway: the OrientDB team camouflages their bugs.

May 21: null passwords unlock all doors

Remember me saying in the beginning that we'd hope OrientDB v2 would be free of stability issues? Not really, first we find small bugs, then distributed instability, then broken transactions and now a huge security bug where a null password would grant anyone access to OrientDB servers. Check [security issue in binary protocol #4191](#):

micha-nerdlicher commented 29 days ago

Thank you for the confirmation. @lvca So this is pretty serious, right? I narrowed the problem down to Line 464 in OServer.java:

```
if (user != null && (iPassword == null || user.password.equals(iPassword))) {
```

iPassword == null I can't believe this! What kind of backdoor is that? I hope this is by design and the null password should be prevented on a higher level in the binary protocol.

I could expect this sort of issue in a new program but do realise OrientDB was born in 2010.

Key takeaway: OrientDB is not ready for production.

May 30: More OrientDB features - Live Query

So, the big objective for OrientDB v2.1 was to [re-write the SQL parser and the query executor](#). This made lots of sense given these were desperately needed improvements and when this was announced I was again hopeful. For a moment it seemed OrientDB would be on the right track until in [May 30](#) I find out about the new killer feature [Live Query](#). Even today milestone 2.1 has [131 issues open, 24 of them bugs](#) and the OrientDB team is busy adding features that expand the code base and increase the chances of adding more... bugs. Why? Only one thing comes to my mind: marketing. It was precisely because of the wealth of features that I chose OrientDB and if they keep adding them (even if riddled with bugs) it will keep attracting newcomers...

(if you wish to comment please start with *Hey OrientDB Leaks* so I know you read the whole post)

Key takeaway: OrientDB team prefers adding new features than to fix existing bugs.

June 5: OrientDB Node.js driver's author recommends a competitor's solution

In another Hacker News post [Native multi-model can compete with pure document and graph databases](#) [codewithcheese](#) asks:

codewithcheese: How does OrientDB compare to ArangoDB?

phpnode: OrientDB fails to deliver on its promises. It has a load of features but they are poorly thought out and/or broken.

ArangoDB is OrientDB done right, but it's a lot younger.

If you're considering using either, you owe it to yourself to investigate whether postgres's Common Table Expressions can do what you want instead. If you can stick with something more mature like postgres, then you'll be saving yourself a lot of pain.

crudbug: "ArangoDB is OrientDB done right"

How are you backing this ? I am sure Luca from OrientDB will have some comments.

phpnode: It's my opinion based on working with OrientDB a lot and ArangoDB a bit for the last 12+ months. I used to be a big cheerleader for OrientDB, but now I don't recommend it. I'm sure Luca will have some comments but he's interested in selling his product.

Ivca: Guys, IMHO I think Charles Pick (alias phpnode) doesn't deserve such attention. Even if he's trying so hard to start a flame against OrientDB I'd rather like to celebrate the Multi-Model approach. Long life to the Multi-Model approach ;-)

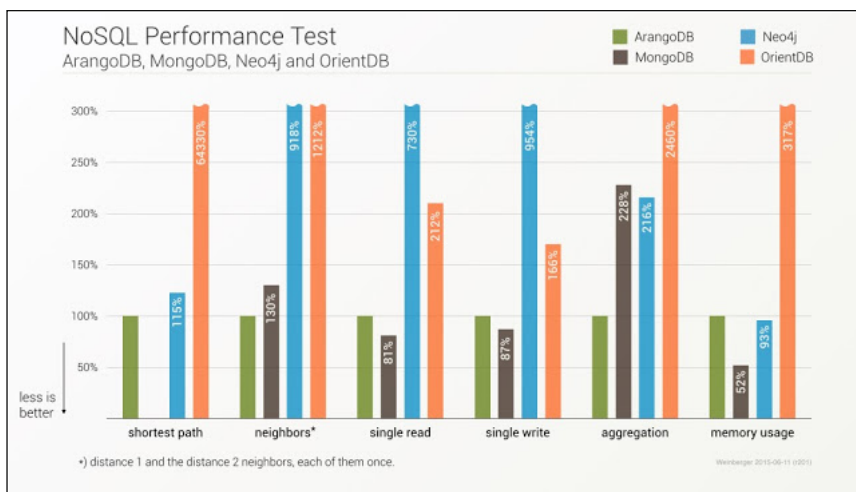
It's interesting that the OrientDB CEO went to the trouble of identifying phpnode by name but forgot to mention that phpnode re-wrote and maintains the official OrientDB Node.js driver [oriento](#) since February 2014 with 380 commits and 186 issues closed. Not only this involved a lot of work and time this was done for free.

Key takeaway: a long time, big supporter and contributor to OrientDB now recommends rival alternatives.

June 11: OrientDB not delivering promises in benchmarks

Note: *this benchmark was done by a competitor vendor and up until now it hasn't been updated to show optimizations made by the OrientDB team. Having said that, the benchmark is open source and can be ran and changed by anyone. The point of including this is not to criticize OrientDB's performance but because this is a precedent for what happens next.*

OrientDB CEO claims [better performance than Neo4J](#), in this [article](#) that doesn't look so clear:



For us these results weren't very significant as OrientDB has had enough performance for our application. Though we couldn't avoid feeling a slight disappointment...

June 16: Welcome to OrientJS, the new Node.JS driver

OrientDB folks, without any agreement or discussion with the OrientDB Node.JS community [fork Oriento into OrientJS](#) claiming:

Unfortunately the original author of "oriento" is not more active on that.

This is not really true and can easily be checked by looking at Oriento [commit logs](#), [issues](#) and [pull requests](#). The original author of Oriento may have stopped contributing with code but he was active at helping users, solving issues and merging code changes from other Oriento contributors, and also from [OrientDB's team](#).

Not only OrientDB turns their back to 31 contributors who freely contributed to OrientDB throughout years, OrientDB does this without a single *thank you*. How ungrateful can a company get?

Key takeaway: OrientDB takes from the community without thanking them, for the sake of... control?

June 17: OrientDB team refuses partnering with Oriento contributors

After the announcement in the previous day, a few users plead OrientDB team to reconsider and focus on a single driver backed by both the existing community and OrientDB team in

<https://github.com/orientechnologies/orientdb/issues/4354#issuecomment-112952745>. A few highlights:

a-unite: I'm the one who doesn't care about labels, but about reality. Oriento was one of the main reasons, why we started with OrientDB. It was and still is the only really usable driver for non Java languages here. [...]

But what is more important - you choose community around the project, not only product (growth is what makes start-up valuable), after all.

dmarcelino: Oriento is more than @phpnode. For instance it currently has 5 owners and 3 collaborators, forking it under a different name and npm package is effectively disowning these people. Forking a project is the last option and it actually goes against the current trend in big FOSS projects: io.js is merging back with node.js and lodash will be merged back to underscore.

Ivca: We kindly asked the transfer of "oriento" project under Orient Technologies umbrella (like we did with other drivers), but @phpnode asked for money...

phpnode: You didn't, you asked for the npm package, not the github projects at all. Since oriento was still being developed until literally today (no, those commits aren't in orientjs) it made no sense for me to transfer the project to people who are not substantially involved in the project development. I did however assure you that any contributions you made would be reviewed and merged quickly, there was no blocker to ongoing collaboration from the oriento side. [...]

Careful now, firstly I didn't ask you for money, I said "make me an offer" which is quite different, but more importantly you seem to be conflating FOSS with "give me your work for free". There's a big difference here. I do not owe you the oriento project, it's something that me and the other collaborators put a lot of effort into, charged you nothing for and your company benefited significantly from. As open source contributors, we rarely benefit financially from our open source work directly, the benefit comes from more indirect links - publishing our work gives us exposure that helps us find new clients, attract developer talent, find more interesting work etc. If you want to claim that project for yourself, then you remove those benefits from the original author, therefore it is reasonable for them to expect something in exchange.

seeden: please take care of us (developers). We need to have a good javascript driver. Ideal solution is to have only one otherwise it will split developers and pull requests too.

Then the CEO does something fairly low:

Ivca: These are the facts: On this topic you said "Anyone is welcome to pick up those projects and continue them, such is the nature of open source, but given that we have no incentive, it won't be us." ([#4354 comment](#)) [...]

The context of "those projects" was an [Oriento re-write](#) and a [new query builder](#) not Oriento itself which was the topic of the conversation...

Ivca:

That said we get the decision to fork and rename the driver to avoid depending by you for all the reasons

above.

StarpTech: @lvca this is not the right way. Your way is a kind of opensource but it signalized the wrong way to work with a community. Where does the hate come from? The prefer way is the community way so create a PR or issue in oriento if you want to change something like contributors of orientdb.

dmarcelino: there is a path open for Orient Technologies to contribute to Oriento and keep the community intact. Can we please try that before resorting to extreme measures as forking?

lucaolivari (President of Orient Technologies): We've been informed of the decision of the actual maintainer to no longer work on the next version of Oriento and that was publicly shared too. Everyone is free to decide wether they want to contribute or not and we respect their choice, but we felt the need to continue development and support.

The reality distortion field kicks in again... as if phpnod had stopped collaborating...

phpnode: I've just added @dmarcelino as a collaborator for oriento on npm - he's a neutral third party that knows the project well. He has the same level of access to oriento that I do. He can publish new releases and has the ability to appoint new collaborators after they've proved themselves capable. If there was ever any blocker to ongoing oriento development, there isn't now.

dmarcelino: @lvca and @lucaolivari, I understand that you guys don't get along with @phpnode, I'm not asking for you to change your opinion about him, but like I said earlier Oriento is more than @phpnode and he clearly has signalled that he is willing to share control over it. He won't and can't hold anybody to ransom so will you reconsider your decision?

And after this the Orient Technologies CEO locks the thread and goes silent.

Key takeaways: OrientDB team does not compromise with the community and fails to conduct proper dialogue. OrientDB is open source but it's not an open community.

June 18: OrientDB instability, again

On a [parallel issue](#) where the previous thread was mentioned another bystander not wanting to get involved reveals the OrientDB problems he's experiencing:

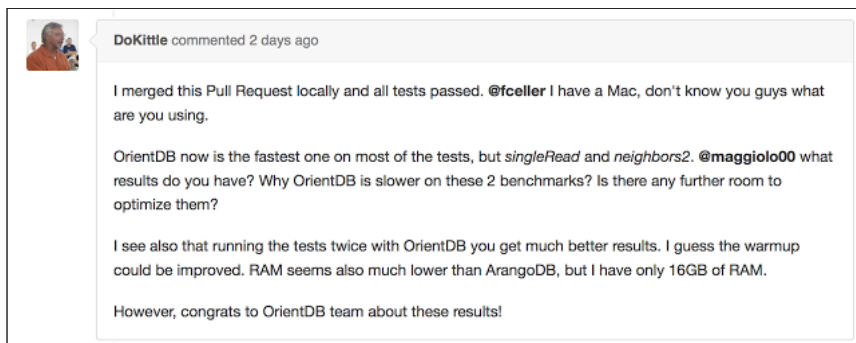
Here are some of my experiences for orientdb database that i have been using for a year.

1. i have realized the current version of orientdb is not stable and promising yet. In that case i really spent a lot of time to debug your stuffs more than doing my own thing. Since here, i really hope we can have most stable version as same as MYSQL in future.
2. A good software should treat testing procedure seriously before release it to market. As for orientdb version releasing that always brings up a lot of issues (etc: bugs) and thus these glitch will cause serious issue to any application especially for the one which involve payment processing.

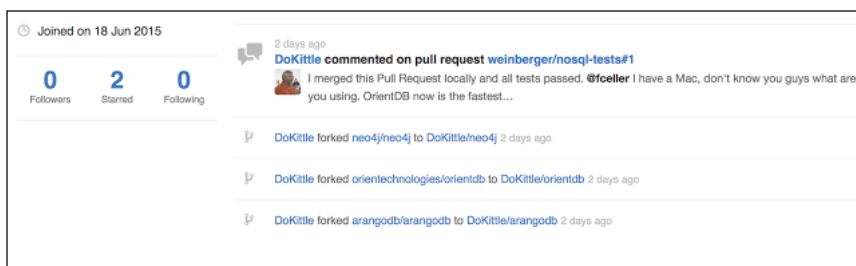
Key takeaway: once more, a user complains of instability and lack of proper testing procedures.

June 18: DoKittle / Don Kattle fake account

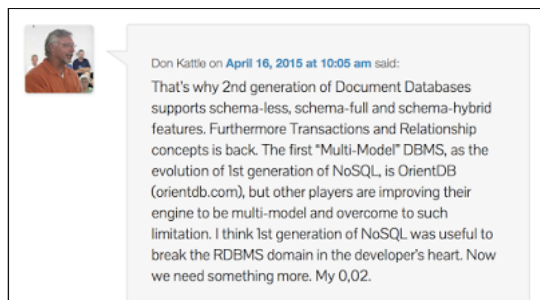
Last but not least I find out that Orient Technologies resorts to a fake account to boost their product. On a [PR to optimize OrientDB results](#) in the benchmark mentioned earlier you can see this:



But if you look at DoKittle's [GitHub profile](#) you notice it was just created for this purpose!



And you can also find this avatar commenting an article named [Why I'm Not Sold on MongoDB](#):



And the funny thing is that this is the photo of [Ron Kittle](#) (former baseball player) addressing [Dyersville city council](#).

All this was uncovered in <https://github.com/weinberger/nosql-tests/pull/1#issuecomment-113470969> where phpnod also notes that Don Kittle ends his comment with "My 0,02" which is [usually used](#) by Orient Technologies CEO.

Key takeaway: OrientDB team is willing to play dirty in order to boost their product

Conclusion

Hopefully you read the whole post and didn't jump straight here. If not please do read the post as the conclusion will make more sense then.

The whole post summarizes my team's experiences and pretty much speaks for itself. We didn't give up because of any specific event or issue but because of all of them. After using OrientDB for more than half year I've reached the following conclusions:

- OrientDB is not robust and is not stable, lots of small issues and some worrying bugs;
- Orient Technologies CEO doesn't treat competitors fairly;
- Paying companies are quitting OrientDB after months of use;
- Orient Technologies CEO prefers to silence and discredit unsatisfied customers than to listen to their issues and dialogue with them;
- Orient Technologies CEO hides bugs and makes them hard to find in GitHub issues so things don't look as bad as they are;
- Did I mention instability?
- OrientDB team prefers adding new features than to fix existing bugs;
- OrientDB Node.js driver's author recommends rival solutions;
- OrientDB team takes from the community without properly recognizing its efforts;
- OrientDB team turned their back to a community who supported them;
- OrientDB is open source but it's not an open community;
- No proper testing procedures: throughout the last 10 months new bugs and regressions keep popping up;
- OrientDB team resorts to a fake account to boost their product.

I fear that the biggest threat to OrientDB is not the product itself but its CEO who again and again has proved to be unreliable. In the end software is as good as the people behind it and OrientDB is severely limited in that regard. **Don't take my word for it, click the links, read, investigate, peruse the OrientDB issues in GitHub, examine the OrientDB google group threads and make your own opinions.**

Unfortunately we are still stuck with OrientDB, we'll need to bear with it until we can find the time and resources to migrate to another product.

Would I come back to OrientDB?

Potentially if:

- The current CEO steps down or comes clean: unfortunately he cannot be trusted and this limits the product's potential;
- Feature freeze for 6 months: OrientDB team needs to address the current issues before moving on to new features;
- Testing, testing, testing: OrientDB team needs to add more tests, implement better test procedures and increase code coverage (I would like to see at least 85% instead of the current [paltry results](#));
- Fix existing problems: nobody wants a database with bugs.

Feel free to comment or asks questions, I'd love to hear feedback (positive or negative) about what me and my team experienced and witnessed. Are you using OrientDB for more than 6 months? Have you experienced issues? Let me know your story in the comments or in the discussions below:

- [Hacker News](#)
- [reddit](#)

UPDATE 1: Orient Technologies CEO addresses this blog post in [5 Wonderful Years on Github – Thanks to the Community!](#).

UPDATE 2: Read my counter reply to the Orient Technologies CEO in [OrientDB's lack of transparency](#).

– a disgruntled OrientDB user

Follow [@orientdb_leaks](#)

Posted by [OrientDB Leaks](#) at 11:28 PM

Recommend this on Google

9 Comments

Labels: [OrientDB Issues](#), [Transparency](#)

Location: Dyersville, IA, USA

9 Comments OrientDB Leaks

 Login Recommend 3  Share

Sort by Best



Join the discussion...

**nitrog7** · a month ago

Was really excited about using it for a couple projects I had. Even more when 2.0 came out and the cluster features. Put the projects on hold for about 9 months and came back. Had a heck of a time migrating my databases to the latest version. Then ran into some bugs with the web admin. Felt a bit uncomfortable to continue with OrientDB if my data could not be properly imported/exported. Ran across your post and I didn't feel like I was going crazy anymore. lol!

Charles Pick is a great dev. Knows what he is doing and put a lot of work into Oriento. I know he was working with OrientDB in a production environment back when he was active with development of the module. When I read in your post that he too switched over, it really made me rethink my decision on continuing with OrientDB.

I do want to mention that when I was working with the NodeJS module early on, Luca was very responsive and was able to get questions and bugs fixed in a timely manner. At the time there was very little documentation. Not sure what's going on now.

All devs love to try out the latest and greatest. Sometimes it grows into something great, other times it falls flat. OrientDB was one of those that is losing its fizz.

Tried using ArangoDB and really enjoy using it. No problems. Was nervous about the AQL syntax but is actually better than pushing SQL to work with document/graph data. The best part of using it is being stress free.

Thanks for the writeup!

-Giraldo

1 ^ v · Reply · Share ›

**Wirone** · 3 months ago

Thank you for sharing this. I was playing with OrientDB for a while as a PoC for next implementation of our product, was optimistic about it, but when I read this... I don't know anymore. I really, really don't like the way Orient Technologies treated Oriento's maintainer and community, I don't like CEO's attitude. Will talk about it with my team.

1 ^ v · Reply · Share ›

**orient** · 5 months ago

Hey OrientDB Leaks,

Storing precious data in Orient DB is unwise. Unfortunately, Orient DB is subject to serious and persistent quality issues that make it impossible to deploy in production environment. It is virtually guaranteed that over time your data will be lost. Eventually some or other bug will appear in a new version which will end up wiping out your db. Our data has been corrupted more times than I can remember. It seems Luca et. al. are more concerned about measuring up to benchmarks and implementing new features, than ensuring basic tenets like data integrity, reliability and durability. Until the people behind it undergo a significant change in attitude / culture, Orient DB will never mature from an experimental project.

1 ^ v · Reply · Share ›

**Brett Coffin** · 6 months ago

I too wasted my time on Orientdb, reporting bugs on bugs and finally gave-up. I am now using Neo4j and could not be happier.

1 ^ v · Reply · Share ›

**orientdb_leaks** Mod → Brett Coffin · 5 months ago

Thanks for sharing **@Brett Coffin**.

^ v · Reply · Share ›

**Sam Williams** · 7 months ago

Hey OrientDB Leaks,

First of all, thank you for opening up about your concerns with OrientDB. We have just started looking into OrientDB for some of our new solutions and your points are helping us to get a balanced view of the state of OrientDB.

To see for myself the issues that you are talking about, I had a quick browse through the invalid/wontfix list that you previously referenced: <https://github.com/orientechno...>

I didn't read all of them, but I picked out a few of the more interesting ones in the first few pages and

saw some responses that I felt were encouraging. For example:

<https://github.com/orientechno...> - (Maybe related to your Mar 27 post?) Initial issue resolved within a day, Luca tagged as a bug, new test cases written

<https://github.com/orientechno...> - Resolved within roughly a week, Luca tagged it as a bug

<https://github.com/orientechno...> - Legitimately closed as it was just asking for someone to look at another ticket that was still open

[see more](#)

1 ^ v · Reply · Share ›



orientdb_leaks Mod ➔ Sam Williams · 6 months ago

Hey **@Sam Williams**,

Thanks for reading through and going a step further and making your own investigation!

I didn't bring up the invalid/wontfix issues, it was part of what anonwarnings said in HackerNews and it was included on the text I quoted from him (which wasn't the reason why I quoted him). Thanks for checking, I'm glad to hear those issues are legit.

The Orient team is usually fairly responsive but unfortunately there is also a flip side to it:

1. Sometimes issues don't get fully resolved and come back in another form;
2. Sometimes they create new issues while fixing something.

In other words, they are quick but not effective, if you look at their github issues count now it reads "1014" and this number has been growing since I started using OrientDB. I believe this is mostly due to poor testing practices and poor code coverage. If you look at their jenkins build (<http://helios.orienttechnologie...>) it shows only 63% line coverage (they actually made progress since I started the blog, when I checked last month it was 58%).

What I brought up was the OrientDB CEO removing the "bug" label on April

[see more](#)

^ v · Reply · Share ›



Izzet Pembeci · 8 months ago

Hey OrientDB Leaks,

I share your feelings and findings towards OrientDB. Thanks for starting this blog. People have the right to know these things before they made a decision to use OrientDB. I can detail my own frustrations later. For now, I'd point these two incidents that support your points:

* Any developer who spent some time with OrientDB will notice that current management prioritizes shiny new features that are not tested well enough above critical bug fixes. Here is an example PR:

<https://github.com/orientechno...>

As one can see, first Lvca tries to include this new feature to 2.1-rc1 at the last moment without any comprehensive tests and no full code overview from OrientDB team. Luckily for users, PR owner is a sensible guy and says it is not ready. Then, Lvca spends some time with the code, makes a failed test to pass in some hackish way. For his credit, he seems to find a more sound way to resolve the issue and delays merging but who knows if there will be a progress as he suggested or suddenly this will be one of many OrientDB features in its current form. May be not the best example of wrong kind of prioritization for a DB product but the whole conversation made me lost confidence.

* May be again a minor point about company's attitude problems but here is what Luca Garulli said

[see more](#)

1 ^ v · Reply · Share ›



orientdb_leaks Mod ➔ Izzet Pembeci · 8 months ago

@Izzet Pembeci, thanks for coming out publicly and sharing your thoughts! Unfortunately @lucagarulli occasionally suppresses uncomfortable messages from wherever he has moderator privileges so it's impossible to have a honest conversation about these topics in OrientDB controlled channels.

"Belittling your competition like that, especially in the FOSS community, is quite irritating to my taste."

- Precisely. OrientDB may be Open Source but it's definitely not an Open Community nor it understands FOSS communities.

"I sincerely want the product to be a success. I think their vision is right. OrientDB can fill some void and become a serious player in the DB world"

- Totally agree. The reason why I went to the trouble of writing this is because I believe there is still hope for OrientDB. Their creators will have to come clean and change their behavior.

If you ever want to share a more complete post, let me know.

^ v · Reply · Share ›

ALSO ON ORIENTDB LEAKS

About OrientDB Leaks

1 comment · 8 months ago

a_nunny_mouse — Hey OrientDB Leaks, Just an FYI your blogpost footer (just below your post and the "follow @orientdb_leaks" ...

WHAT'S THIS?

Deceit, the biggest of OrientDB problems

4 comments · 8 months ago

Javier Lorenzana — Thanks for the article, man. Graph DBs are a viciously competitive market right now, and it's pretty useful to have both ...

 [Subscribe](#)  [Add Disqus to your site](#) [Add Disqus](#) [Add](#)  [Privacy](#)

[Newer Post](#)

[Home](#)

[Older Post](#)

Subscribe to: [Post Comments \(Atom\)](#)