

PEV2



Pierre Giraud

PGSession - 21 Novembre 2019

PEV2

TITRE : PEV2

SOUS-TITRE :

DATE: Pierre Giraud

Il était une fois...

PEV2

EXPLAIN

ET SES PLANS D'EXÉCUTION...

Pas facile à lire, hein ?

EXPLAIN.DEPESZ.COM

explain.depesz.com
PostgreSQL's explain analyze made readable

new explain history help about contact login

Result: W5zn : pike_02Oct2019

Settings [Add optimization](#)

#	exclusive	inclusive	rows x	rows	loops	node
1.	2.430	4,942,930.824	↓ 320.0	320	1	<ul style="list-style-type: none"> → Sort (cost=91.649.33..91.649.33 rows=1 width=3.399) (actual time=4.942.930.810..4.942.930.824 rows=320 loops=1) Sort Key: ((ui.lastname)::character varying(50)) COLLATE "en_US", ((ui.firstname "en_US", ui.duplicate_name, ((login.loginname)::character varying(255)) COLLATE (encoding:UTF8)::character varying(50)) COLLATE "en_US", (ex.expensedate varying(255)) COLLATE "en_US", ((e.description)::character varying(255)) COLL (expensetype3.name)::character varying(50)) COLLATE "en_US", ex.billtoctid, ("SELECT" 1::approved::name)::character varying(50)) COLLATE "en_US", (("approved::name)::character varying(50)) COLLATE "en_US", eif.expensestat ("SELECT" 1::approved::name)::character varying(50)) COLLATE "en_US", ((i_1.code)::char "en_US", ((i_1.name)::character varying(255)) COLLATE "en_US", ((k.code)::ch "en_US Sort Method: quicksort Memory: 154kB Buffers: shared hit=2967970075 Initplan (forSort)
2.						
3.	0.003	0.003	↑ 1.0	1	1	<ul style="list-style-type: none"> → Seq Scan on systeminformation (cost=0.00..1.01 rows=1 width=4) (actual tim Buffers: shared hit=1)
4.	0.005	0.005	↑ 1.0	1	1	<ul style="list-style-type: none"> → Seq Scan on systeminformation.systeminformation_1 (cost=0.00..1.01 rows=1 Buffers: shared hit=1)
5.	6.396	4,942,928.386	↓ 320.0	320	1	<ul style="list-style-type: none"> → Nested Loop Semi Join (cost=6.72..91.647.30 rows=1 width=3.399) (actual time=188.986..4.942.928.386 rows=320 loops=1) Join Filter: (ex.userid = userlocation9.userid) Buffers: shared hit=2967970061
6.	4.026	4,942,912.673	↓ 1,331.0	1,331	1	<ul style="list-style-type: none"> → Nested Loop Left Join (cost=6.44..91.646.92 rows=1 width=393) (actual time=188.949..4.942.912.673 rows=1,331 loops=1)

EXPLAIN.DEPESZ.COM

- Existe depuis longtemps,
- Largement reconnu et utilisé,
- Maintenu (développement actif),
- Pratique et simple.

PEV2

EXPLAIN.DEPESZ.COM

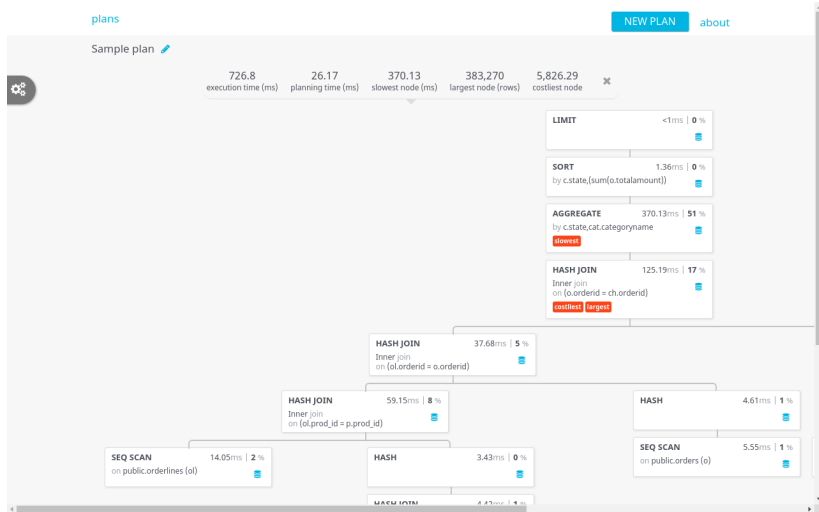
Seulement voilà...

- Pas très sexy,
- Affichage limité,
- Pas intégrable,
- Sur le web.

PEV

POSTGRES EXPLAIN VISUALIZER

tatiyants.com/pev



PEV2

TATIYANTS.COM/PEV

- Plébicité (github 2.3k [?](#)),
- Sexy, moderne,
- Mise en évidence des informations,
- Pas de stockage.

TATIYANTS.COM/PEV

Seulement voilà...

- Abandonné,
- Pas intégrable.

LES BESOINS

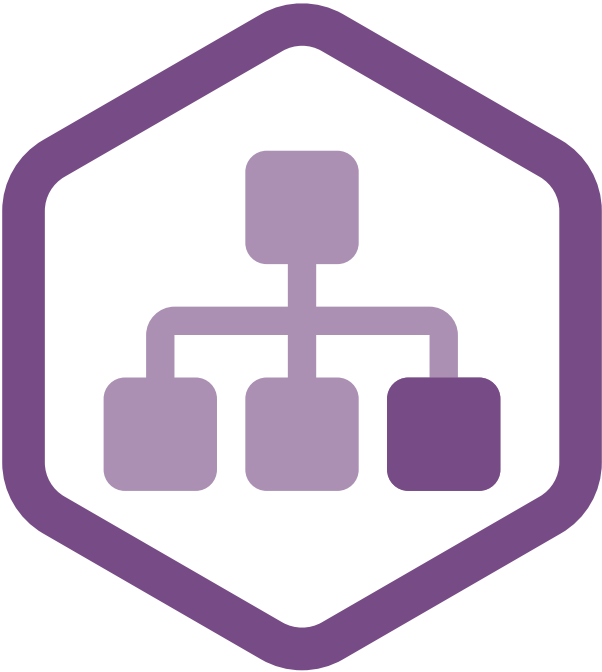
- Intégration dans d'autres outils (*TemBoard, DebugToolbar*)
- Mise à jour (versions de PostgreSQL)
- Ajout fonctionnalités

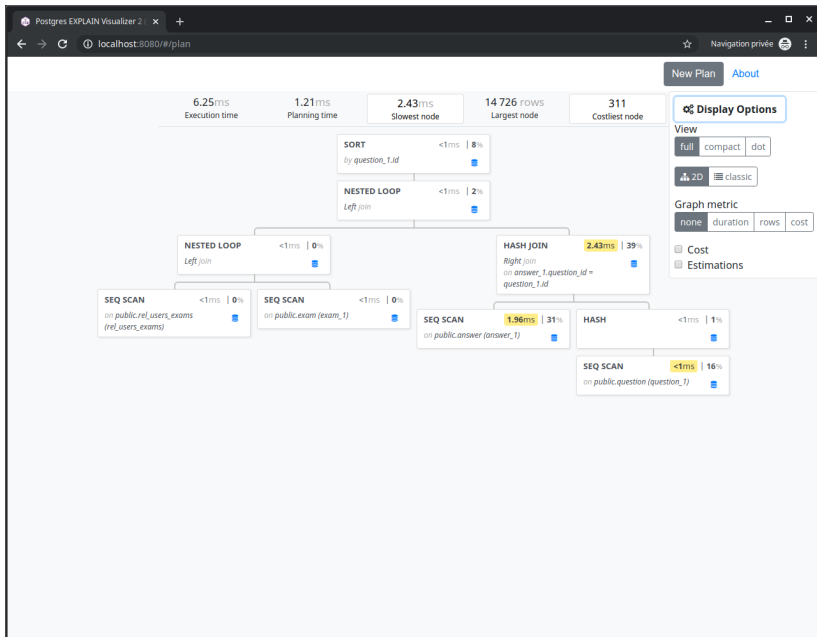
DÉCISION

POC réécriture de PEV

☑ très rapide car code bien écrit

PLEASE WELCOME PEV2!





dalibo.github.io/pev2

PEV2

NOUVELLES FONCTIONNALITÉS & DIFFÉRENCES

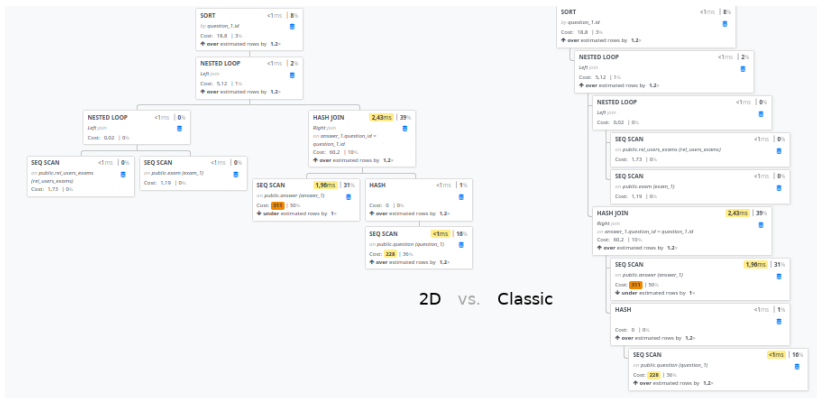
TEXT PARSING

“ Si c'est pas capable de lire un plan au format TEXT, j'm'en servirai pas. ”

Guillaume L.

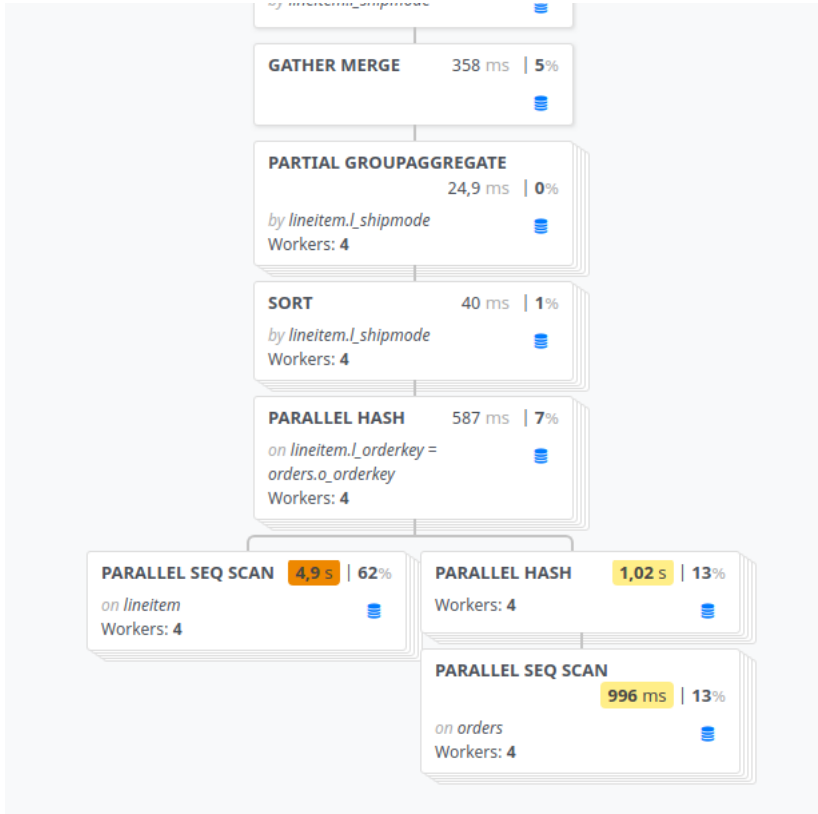
JSON	TEXT
<pre>{ "Plan": { "Node Type": "Aggregate", "Strategy": "Sorted", "Partial Mode": "Simple", "Parallel Aware": false, "Actual Startup Time": 112.534, "Actual Total Time": 144.206, "Actual Rows": 1001, "Actual Loops": 1, "Group Key": ["toto1.c1x2"], "Shared Hit Blocks": 5, "Shared Read Blocks": 0, "Shared Dirty Blocks": 0, "Shared Written Blocks": 0, "Local Hit Blocks": 0, "Local Read Blocks": 0, "Local Dirty Blocks": 0, "Local Written Blocks": 0, "Temp Read Blocks": 0, "Temp Written Blocks": 0, "Plans": [{ "Node Type": "Result", "Parent Relationship": "InitPlan", "Subplan Name": "CTE toto", "Parallel Aware": false, "Actual Startup Time": 0.57, "Actual Total Time": 39.32, "Actual Rows": 1001, "Actual Loops": 1 }] } }</pre>	<pre> + (actual time=133.200..172.325 rows=1001 loops=1) + (actual time=2.204..54.254 rows=1001 loops=1) + (actual time=0.430..0.842 rows=1001 loops=1) + Sort Key: ((bebe.c1 * 2)) + Sort Method: quicksort Memory: 71kB + Buffers: shared hit=5 + -> Seq Scan on bebe (actual time=0.019..0.224 + Buffers: shared hit=5 + (actual time=133.107..134.741 rows=1001 loops= + Cond: (toto1.c1x2 = toto2.c1x2) + (actual time=133.723..136.958 rows=1001 loops=1) + Sort Key: toto2.c1x2 + Sort Method: quicksort Memory: 71kB + Buffers: shared hit=11 + -> Subquery Scan on toto1 (actual time=2.310..95 + Buffers: shared hit=11 + -> CTE Scan on toto (actual time=2.308..94 + Buffers: shared hit=11 + -> Sort (actual time=36.723..36.958 rows=1001 loops=1) + Sort Key: toto2.c1x2 + Sort Method: quicksort Memory: 71kB + -> Subquery Scan on toto2 (actual time=0.032..35 + -> CTE Scan on toto toto 1 (actual time=0. + Planning Time: 1.722 ms + Actual Total Time: 39.321 Execution Time: 173.614 ms + Actual Rows": 1001, + "Actual Loops": 1 + } + } + } }</pre>

VIEW OPTIONS

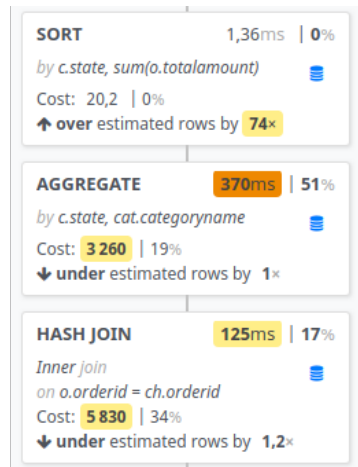
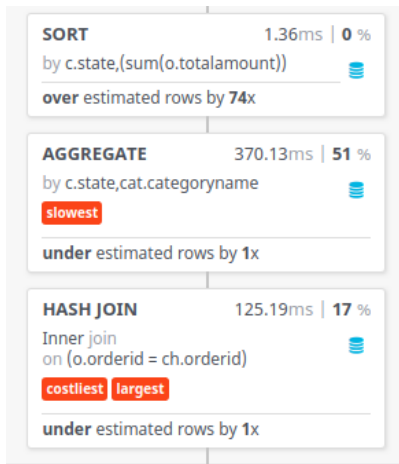


2D VS. Classic

PARALLELISM



MISE EN ÉVIDENCE



(PEV était limité aux superlatifs)

PEV2

UTILISATION COMME COMPOSANT

DISPONIBLE SUR NPM

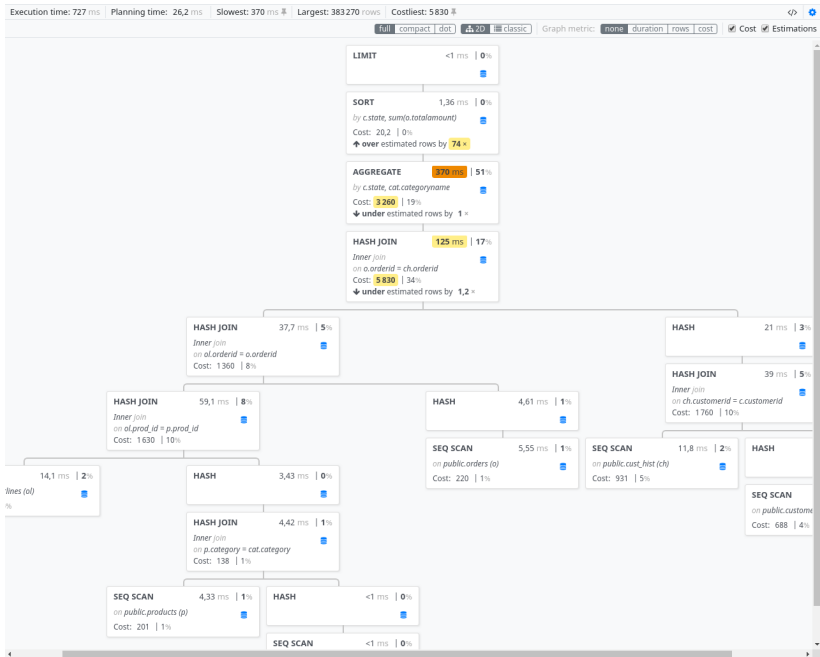
The screenshot shows the NPM package page for 'pev2'. At the top, there is a search bar with the text 'Search packages' and a red 'Search' button. Below the search bar, a banner reads: 'Share private packages across your team with npm Orgs, now with simplified billing via the aws marketplace! [Learn more >](#)'. The package name 'pev2' is displayed in large blue text, with '0.1.15' and 'Public' in smaller text, and 'Published 5 days ago' below. A navigation bar contains 'Readme' (highlighted in yellow), 'Admin', '12 Dependencies', '0 Dependents', and '14 Versions'. The main content area is divided into two columns. The left column contains: 'pev2' in large blue text, a description 'A VueJS component to show a graphical vizualization of a PostgreSQL execution plan.', a 'Greenkeeper enabled' badge, 'See Demo.', and a 'Disclaimer' section stating 'This project is a rewrite of the excellent [Postgres Explain Visualizer \(pev\)](#). Kudos go to [Alex Tatiyants](#).' The right column contains: an 'install' section with a code input field containing '> npm i pev2', a '205 weekly downloads' section with a blue line graph, a 'version 0.1.15' and 'license none' section, and a 'last publish 5 days ago' section.

INTÉGRATION

```
<div id="app">
  <pev2 :plan-source="plan" :plan-query="query"></pev2>
</div>
```

(résumé)

INTÉGRATION



INTÉGRATION

Application de démo dans le dépôt github.

dalibo.github.io/pev2

fonctionne dans le navigateur, pas de rétention

PEV2

INTÉGRATION




Dimitri Fontaine
@tapoueh




I'm publishing a [#PostgreSQL](#) Query Plan Visualizer at theartofpostgresql.com//explain-plan-... ; that's an installation of PEV2! Give it a try, we have an experimental "save as PNG" feature!

À PARTAGER

explain.dalibo.com

Brought to you with  by Dalibo ;-)

 explain.dalibo.com

Title

Plan (text or JSON)

Paste execution plan
Or drop a file

Query

Paste corresponding SQL query
Or drop a file

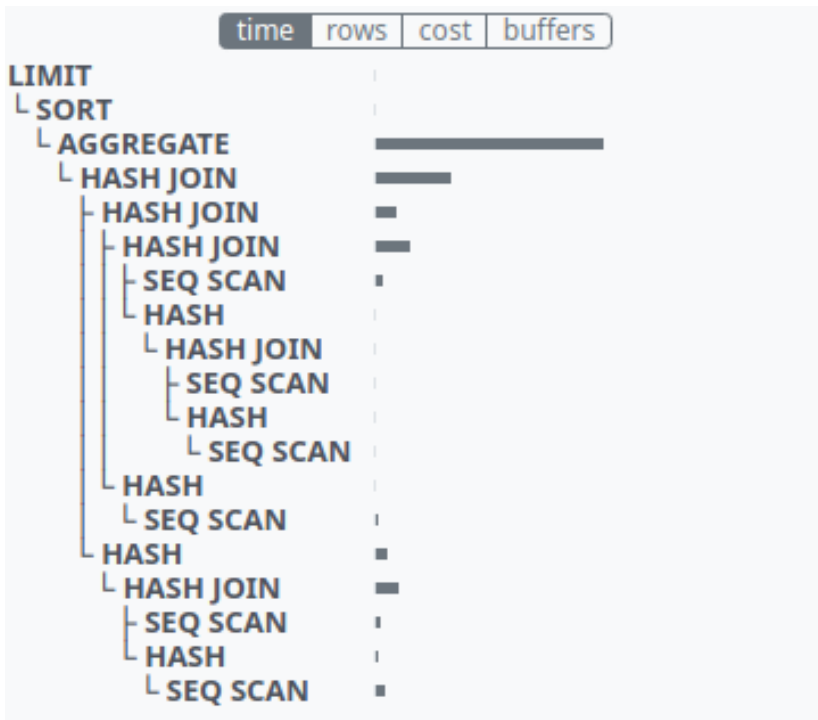
For best results, use `EXPLAIN (ANALYZE, COSTS, VERBOSE, BUFFERS, FORMAT JSON)`
`psql` users can export the plan to a file using `psql -qat -f explain.sql > analyze.json`

explain.dalibo.com

Vous pouvez envoyer vos plans en POST.

À VENIR

Diagramme (vue synthétique)



Feedbacks and contributions are welcome!

Pierre Giraud

DALIBO

PEV2 on github

www.dalibo.com

PEV2

pgira