Monitoring Postgres

Let's build something for Casual DBAs!

pganalyze.com

Who am I?

- **Michael Renner**
- Sysadmin & DBA Background
- Lurker in int'l community, more active in .de/.at
- Working with Lukas Fittl on pganalyze.com

Monitoring is Hard

- Everybody is looking for something different

- For some things there isn't a nice/well-defined interface

- and for other things no interface exists at all

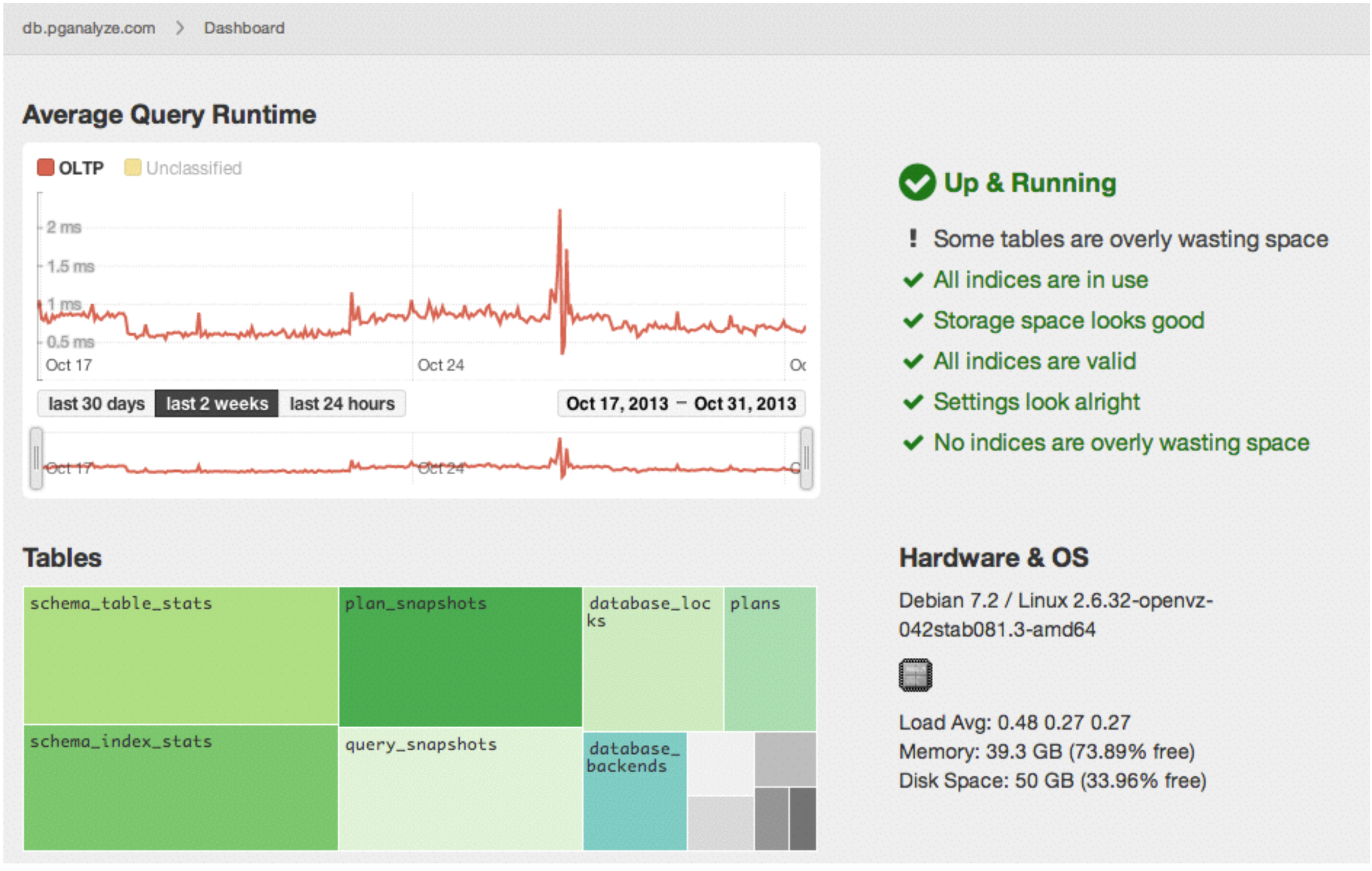
Used today

- Home-baked in-house solutions
- or Enterprise products targeted at large setups
- or nothing at all!

```
SELECT pgn.nspname, relname, pg_size_pretty(relpages::bigint * 8 * 1024) AS size, CASE WHEN relkind =
't' THEN (SELECT pgd.relname FROM pg_class pgd WHERE pgd.reltoastrelid = pg.oid) WHEN nspname =
'pg_toast' AND relkind = 'i' THEN (SELECT pgt.relname FROM pg_class pgt WHERE SUBSTRING(pgt.relname
FROM 10) = REPLACE(SUBSTRING(pg.relname FROM 10), '_index', '')) ELSE (SELECT pgc.relname FROM
pg_class pgc WHERE pg.reltoastrelid = pgc.oid) END::varchar AS refrelname, CASE WHEN nspname =
'pg_toast' AND relkind = 'i' THEN (SELECT pgts.relname FROM pg_class pgts WHERE pgts.reltoastrelid =
(SELECT pgt.oid FROM pg_class pgt WHERE SUBSTRING(pgt.relname FROM 10) = REPLACE(SUBSTRING(pg.relname
FROM 10), '_index', ''))) END AS relidxrefrelname, relfilenode, relkind, reltuples::bigint, relpages
FROM pg_class pg, pg_namespace pgn WHERE pg.relnamespace = pgn.oid AND pgn.nspname NOT IN
('information_schema', 'pg_catalog') ORDER BY relpages DESC;
```

nspname	relname	size	refrelname
public	schema_table_stats	1409 MB	
public	schema_index_stats	1284 MB	
public	plan_snapshots	1111 MB	
public	query_snapshots	974 MB	
public	database_locks	659 MB	pg_toast_25107
public	database_backends	379 MB	pg_toast_25096
public	index_schema_index_stats_on_schema_index_id	360 MB	
public	index_schema_index_stats_on_snapshot_id	335 MB	
public	schema_index_stats_pkey	326 MB	
public	plans	322 MB	pg_toast_18874
public	index_plan_snapshots_on_plan_id	198 MB	
public	index_query_snapshots_on_query_id	191 MB	
public	index_plan_snapshots_on_snapshot_id	184 MB	
public	plan_snapshots_pkey	182 MB	
public	index_query_snapshots_on_snapshot_id	172 MB	
public	query_snapshots_pkey	170 MB	
public	schema_table_stats_schema_table_id_snapshot_id	160 MB	
public	index_database_locks_on_snapshot_id	155 MB	
public	index_schema_table_stats_on_schema_table_id	150 MB	
public	schema_table_stats_pkey	138 MB	

pganalyze


db.pganalyze.com
Dashboard
Average Query Runtime
OLTP
Unclassified
2 ms
1.5 ms
1 ms
0.5 ms
Oct 17
Oct 24
last 30 days
last 2 weeks
last 24 hours
Oct 17, 2013 – Oct 31, 2013
Up & Running
Some tables are overly wasting space
All indices are in use
Storage space looks good
All indices are valid
Settings look alright
No indices are overly wasting space
Tables
schema_table_stats
plan_snapshots
database_locks
plans
schema_index_stats
query_snapshots
database_backends
Hardware & OS
Debian 7.2 / Linux 2.6.32-openvz-042stab081.3-amd64
Load Avg: 0.48 0.27 0.27
Memory: 39.3 GB (73.89% free)
Disk Space: 50 GB (33.96% free)

pganalyze

- Python agent collecting aggregated data every 10 minutes
- Converts it to JSON, posts it to our web service
- Visualisation, recommendations & alerts done on server side

What are we collecting

- pg_stat_plans
- pg_stat(io)_user_(tables|indexes)
- Postgres configuration (GUCs)
- locks, backends, database statistics
- OS performance data (IO, CPU, Memory)
- …

For the Community

- Free for postgresql.org and non-profit projects
- Python collector designed to be reusable, BSD license
  - Will be converted to a library in the future
- JS graphing library also open-sourced

Status quo

- Monitoring 18 production databases
- First trials with paying customers
- If this works out, we'll do & sponsor more work in the monitoring area

Thanks for listening

- Give it a try and tell us what you think:
  - https://**pganalyze.com**
  - https://**github.com/pganalyze**
- Thanks to Peter & Abhijit for their work on **pg_stat_plans**
- Questions?