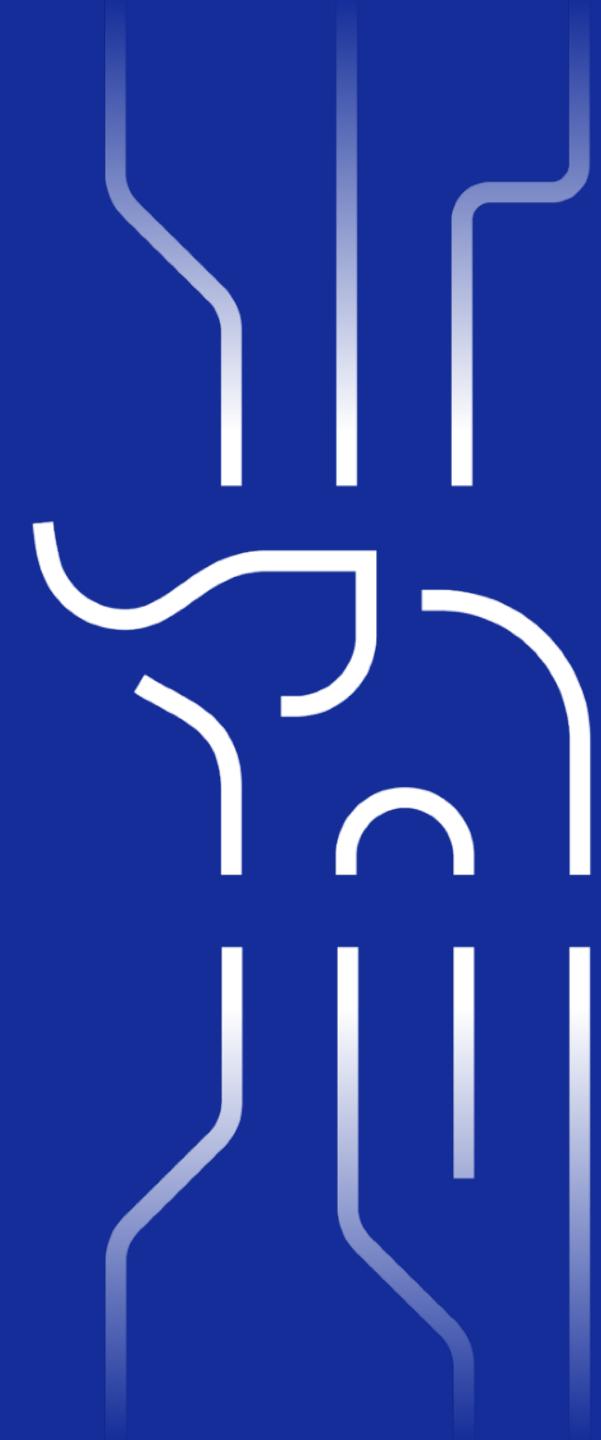




PostgreSQL как искусство

Соколова Екатерина

PGMeetup.VLD 2025



Михаил Жилин



Postgres Professional

Директор департамента
системной
производительности

Сергей Зимин



Postgres Professional

Старший консультант по
предпродажной подготовке

Евгений Безручкин



АО Флант

DevOps инженер



Андрей Билле

Postgres Professional

Главный QA инженер

ЭЗОТЕРИКА

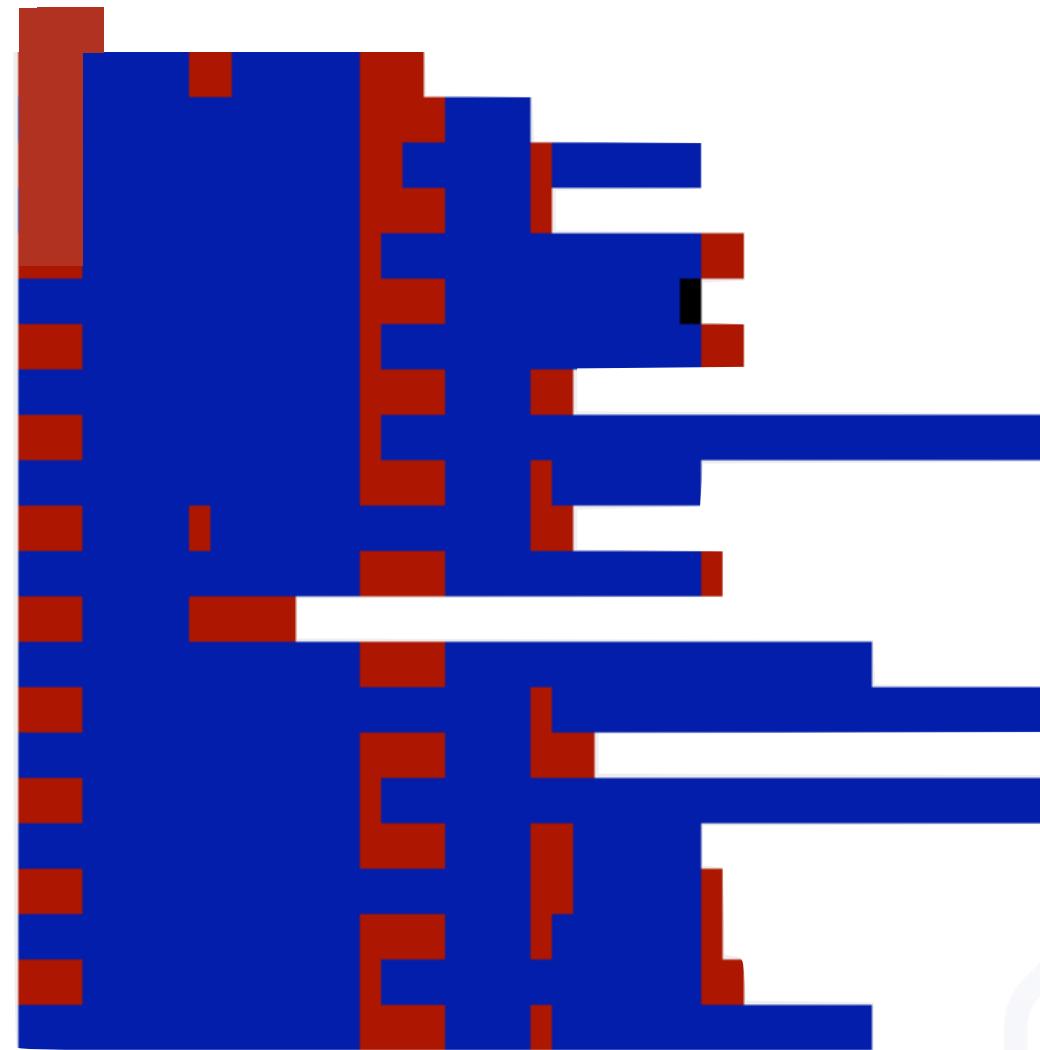


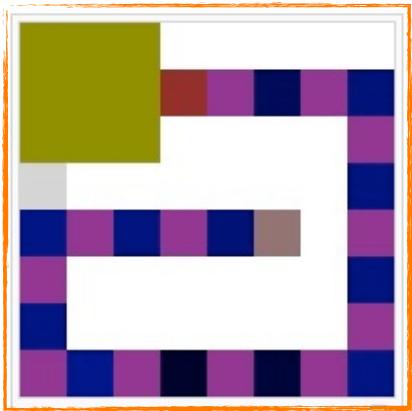
ЭЗОТЕРИКА

* совокупность знаний и сведений, доступных только посвященным, которые стремятся понять скрытые стороны жизни и реальности



Whitespace





```
1 IT'S SHOWTIME
2 TALK TO THE HAND "hello world"
3 YOU HAVE BEEN TERMINATED
4
```

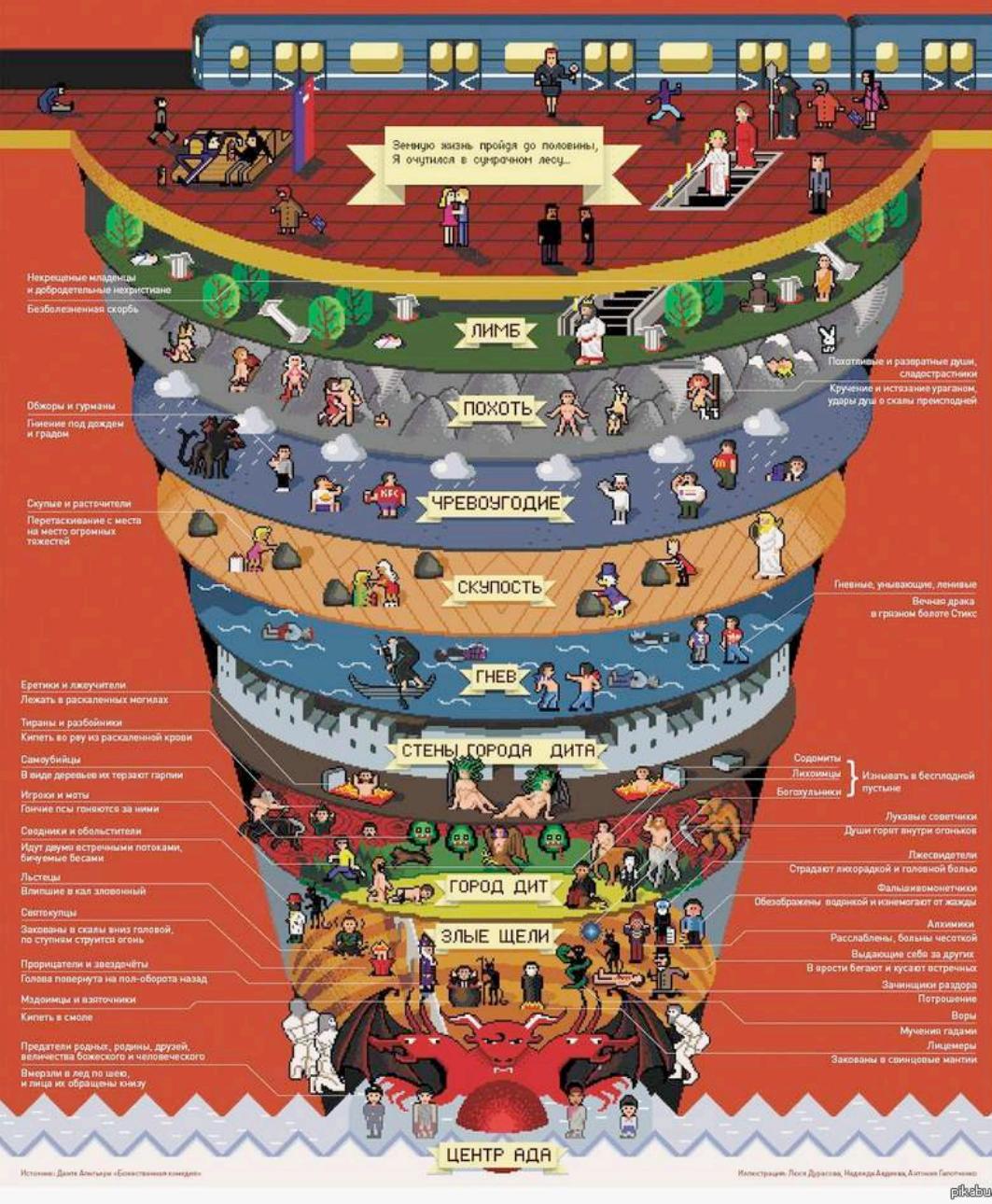
```
1 Ook. Ook? Ook. Ook.
2 Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
3 Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook.
4 Ook! Ook. Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
5 Ook. Ook. Ook! Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook?
6 Ook! Ook! Ook? Ook? Ook. Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook. Ook.
7 Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook. Ook! Ook. Ook. Ook. Ook. Ook.
8 Ook. Ook. Ook! Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook.
9 Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook. Ook.
10 Ook. Ook. Ook. Ook. Ook. Ook. Ook? Ook! Ook! Ook? Ook! Ook? Ook.
11 Ook. Ook? Ook. Ook? Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook.
12 Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook. Ook! Ook? Ook? Ook. Ook.
13 Ook. Ook? Ook! Ook? Ook? Ook. Ook! Ook! Ook! Ook! Ook! Ook! Ook!
14 Ook? Ook. Ook? Ook. Ook? Ook. Ook. Ook! Ook. Ook. Ook. Ook. Ook. Ook.
15 Ook! Ook. Ook! Ook!
16 Ook! Ook!
17 Ook! Ook!
18 Ook! Ook. Ook? Ook. Ook? Ook. Ook! Ook.
```



Malbolge

PostgresPro

КРУГИ АДА ПО ДАНТЕ

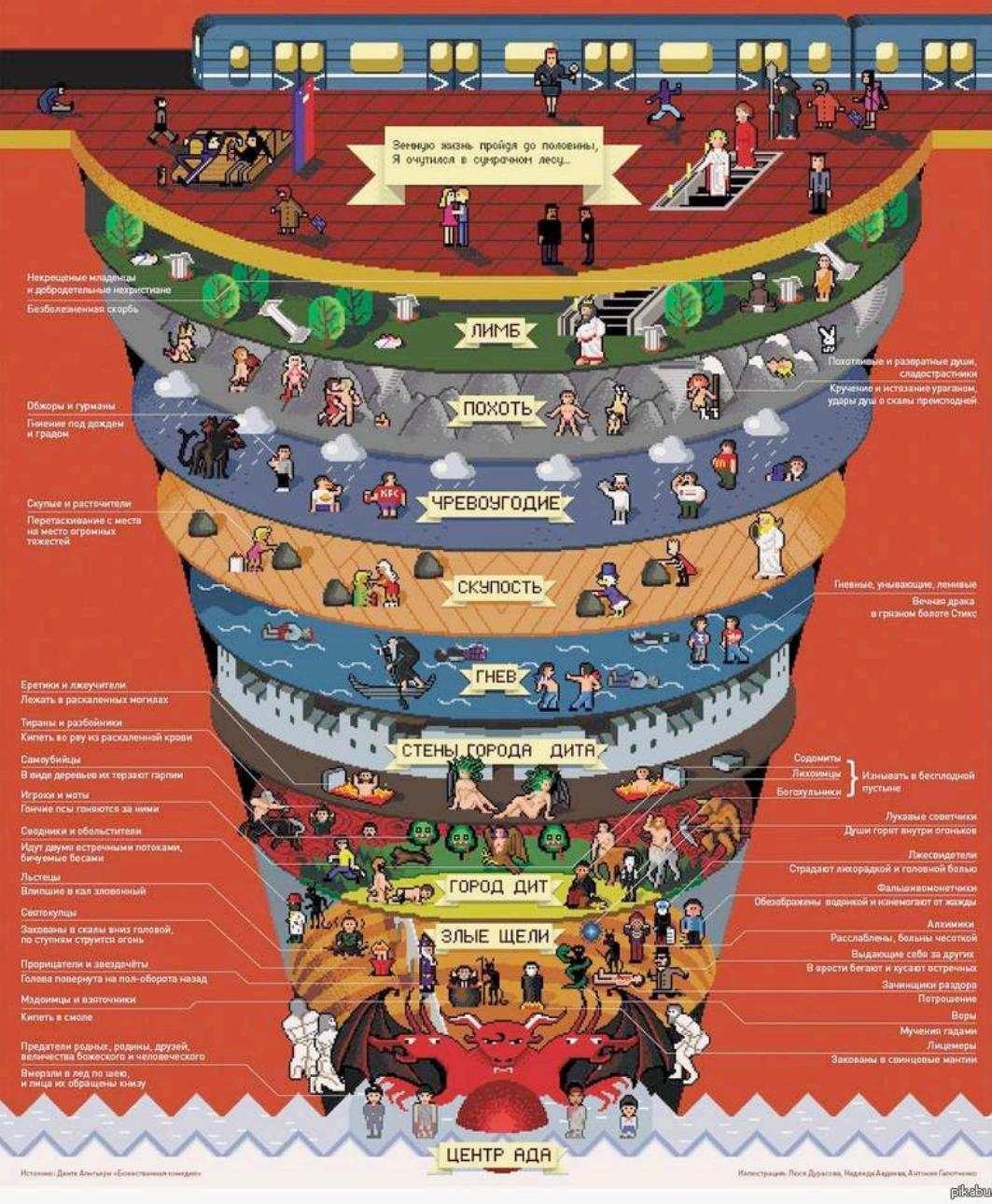


Malbolge

PostgresPro

- назван в честь 8-го круга ада
- создатель не смог написать на нём программу
- код первой программы «Hello World» сгенерирован спустя 2 года

КРУГИ АДА ПО ДАНТЕ



Malbolge

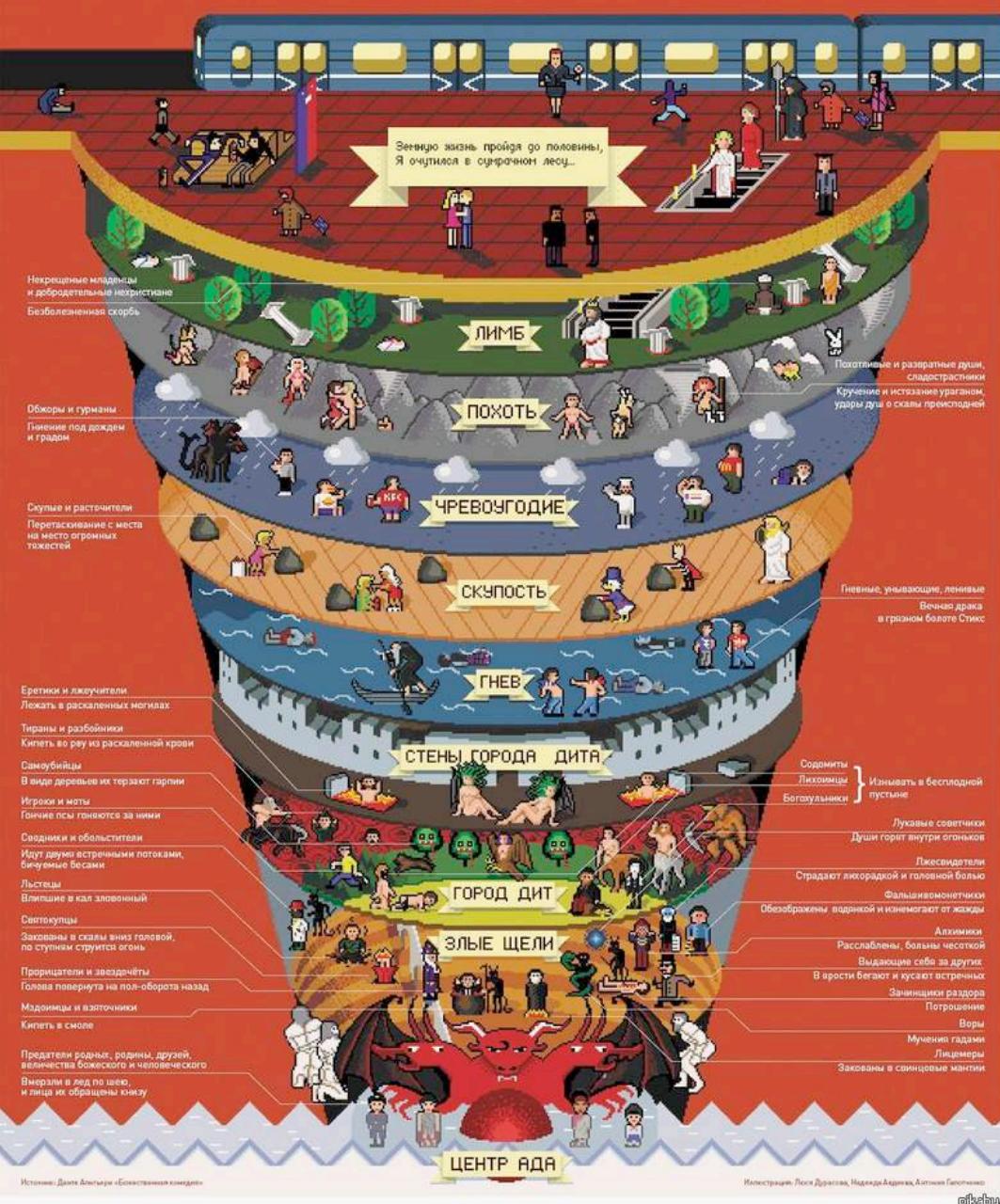
- назван в честь 8-го круга ада
- создатель не смог написать на нём программу
- код первой программы «Hello World»
сгенерирован спустя 2 года

```
(=<`9876Z4321UT.-Q+*)M'&%$H"!~}lBzy?=l{z]KwZY44Eq0
/{mlk**hKs_dG5[m_BA{?-Y;;Vb'rR5431M}/.zHGwEDCBA@
98\6543W10/.R,+O<
```

```
('&%:9]!~}lz2Vxwv-,POqponl$Hjig%eB@{@>}=<M:9wv6WsU
2T|nm-,jcL(I&%$#"CB]V?Tx<uVtT`Rp03NIF.Jh++FdbCBA
@?]!~|4XzyTT43Qsqq(Lnmkj"Fhg${z@>
```

```
(=<`#9]~6ZY32Vx/4Rs+0No-&Jk)"Fh}|Bcy?`=*z]Kw%oG4U
US0/@-ejc(:'8dc
```

КРУГИ АДА ПО ДАНТЕ



Malbolge

- назван в честь 8-го круга ада
- создатель не смог написать на нём программу
- код первой программы «Hello World»
сгенерирован спустя 2 года

```
(=<`9876Z4321UT.-Q+*)M'&%$H"!~}lBzy?=l{z]KwZY44Eq0
/{mlk**hKs_dG5[m_BA{?-Y;;Vb'rR5431M}/.zHGwEDCBA@
98\6543W10/.R,+O<
```

```
('&%:9]!~}lz2Vxwv-,POqponl$Hjig%eB@{@>}=<M:9wv6WsU
2T|nm-,jcL(I&%$#"CB]V?Tx<uVtT`Rp03NIF.Jh++FdbCBA
@?]!~|4XzyTT43Qsqq(Lnmkj"Fhg${z@>
```

```
(=<`#9]~6ZY32Vx/4Rs+0No-&Jk)"Fh}|Bcy?`=*z]Kw%oG4U
US0/@-ejc(:'8dc
```

- цель: самый сложный ЯП

github.com/postgrespro/

[github.com/postgrespro/ imgsmlr](https://github.com/postgrespro/imgsmlr)

ImgSmlr – is a PostgreSQL extension which implements similar images searching functionality.

ImgSmlr method is based on Haar wavelet transform. The goal of ImgSmlr is not to provide most advanced state of art similar images searching methods. ImgSmlr was written as sample extension which illustrate how PostgreSQL extendability could cover such untypical tasks for RDBMS as similar images search.

[github.com/postgrespro/ imgsmlr](https://github.com/postgrespro/imgsmlr)

ImgSmlr – is a PostgreSQL extension which implements similar images searching functionality.

ImgSmlr method is based on Haar wavelet transform. The goal of ImgSmlr is not to provide most advanced state of art similar images searching methods. ImgSmlr was written as sample extension which illustrate how PostgreSQL extendability could cover such untypical tasks for RDBMS as similar images search.



[github.com/postgrespro/!\[\]\(3313ab456208781028d87c207f762ca9_img.jpg\) imgsmlr](https://github.com/postgrespro/imgsmlr)



1.jpg



2.png



3.gif



4.jpg



5.png



6.gif

github.com/postgrespro/imgsmr



1.jpg

2.png

3.gif

4.jpg

5.png

6.gif

	1	2	3	4	5	6
1	0.00	1.54	1.62	2.85	3.18	2.72
2		0.00	1.26	2.75	2.90	2.68
3			0.00	2.79	3.10	2.81
4				0.00	2.43	2.10
5					0.00	2.95
6						0.00

github.com/postgrespro/imgsmlr



	1	2	3	4	5	6
1	0.00	1.54	1.62	2.85	3.18	2.72
2		0.00	1.26	2.75	2.90	2.68
3			0.00	2.79	3.10	2.81
4				0.00	2.43	2.10
5					0.00	2.95
6						0.00

[github.com/postgrespro/ imgsmr](https://github.com/postgrespro/imgsmr)

[github.com/postgrespro/!\[\]\(6701d2b106a848d9d8071847353cdc08_img.jpg\) imgsmlr](https://github.com/postgrespro/imgsmlr)



1.png



2.png

[github.com/postgrespro/ imgsmplr](https://github.com/postgrespro/imgsmplr)



github.com/postgrespro/imgsmlr



1.png



2.png



3.png



4.png

pattern

	1	2	3	4
1	0.00	3.25	4.34	4.66
2		0.00	3.22	3.56
3			0.00	1.65
4				0.00

signature

	1	2	3	4
1	0.00	2.10	2.47	2.67
2		0.00	2.42	2.15
3			0.00	0.95
4				0.00

Своё расширение!

Своё расширение!



contrib/**NAME**



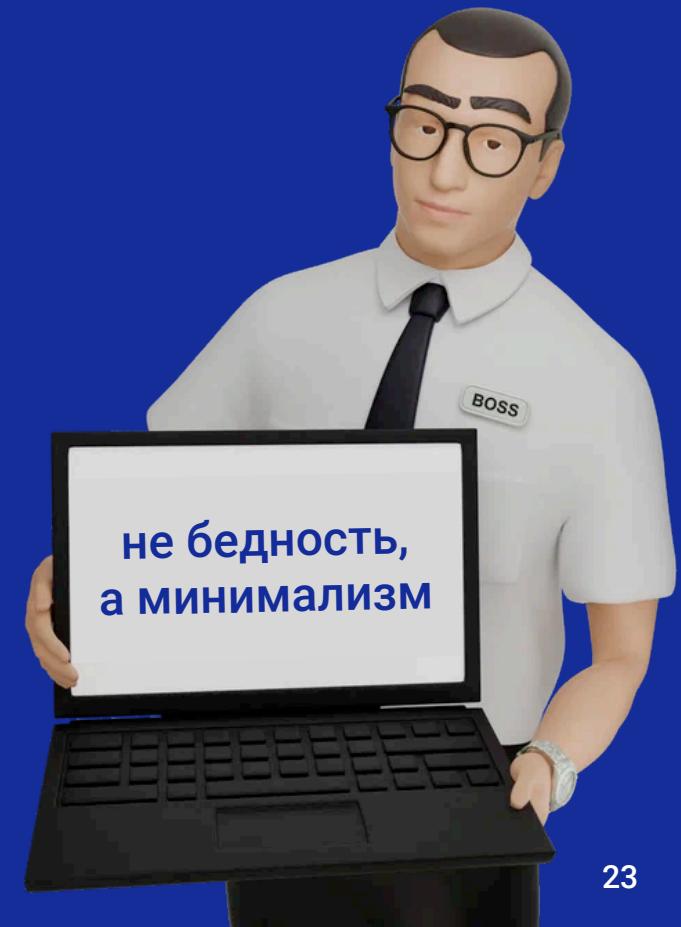
NAME.control



Makefile



NAME--1.0.sql



Своё расширение!

```
default_version = '1.0'  
module.pathname = '$libdir/my_mini_ext'  
relocatable = true
```

my_mini_ext.control

CREATE ...

```
MODULE_big = my_mini_ext  
DATA = my_mini_ext--1.0.sql  
EXTENSION = my_mini_ext  
subdir = contrib/my_mini_ext  
top_builddir = ../../..  
include $(top_builddir)/src/Makefile.global  
include $(top_srcdir)/contrib/contrib-global.mk
```

Makefile

my_mini_ext--1.0.sql

Своё расширение!

```
default_version = '1.0'  
module.pathname = '$libdir/my_mini_ext'  
relocatable = true
```

my_mini_ext.control

```
CREATE FUNCTION hello_world()  
RETURNS TEXT AS $$  
SELECT 'Hello, world!';  
$$ LANGUAGE SQL;
```

```
MODULE_big = my_mini_ext  
DATA = my_mini_ext--1.0.sql  
EXTENSION = my_mini_ext  
subdir = contrib/my_mini_ext  
top_builddir = ../../..  
include $(top_builddir)/src/Makefile.global  
include $(top_srcdir)/contrib/contrib-global.mk
```

Makefile

my_mini_ext--1.0.sql

Своё расширение!

```
$ make install  
$ pg_ctl restart  
$ psql postgres
```

```
postgres=# CREATE EXTENSION my_mini_ext;
```

```
CREATE EXTENSION
```

```
CREATE FUNCTION hello_world()  
RETURNS TEXT AS $$  
    SELECT 'Hello, world!';  
$$ LANGUAGE SQL;
```

my_mini_ext--1.0.sql

```
postgres=# SELECT hello_world();
```

```
hello_world
```

```
-----
```

```
Hello, world!
```

```
(1 row)
```

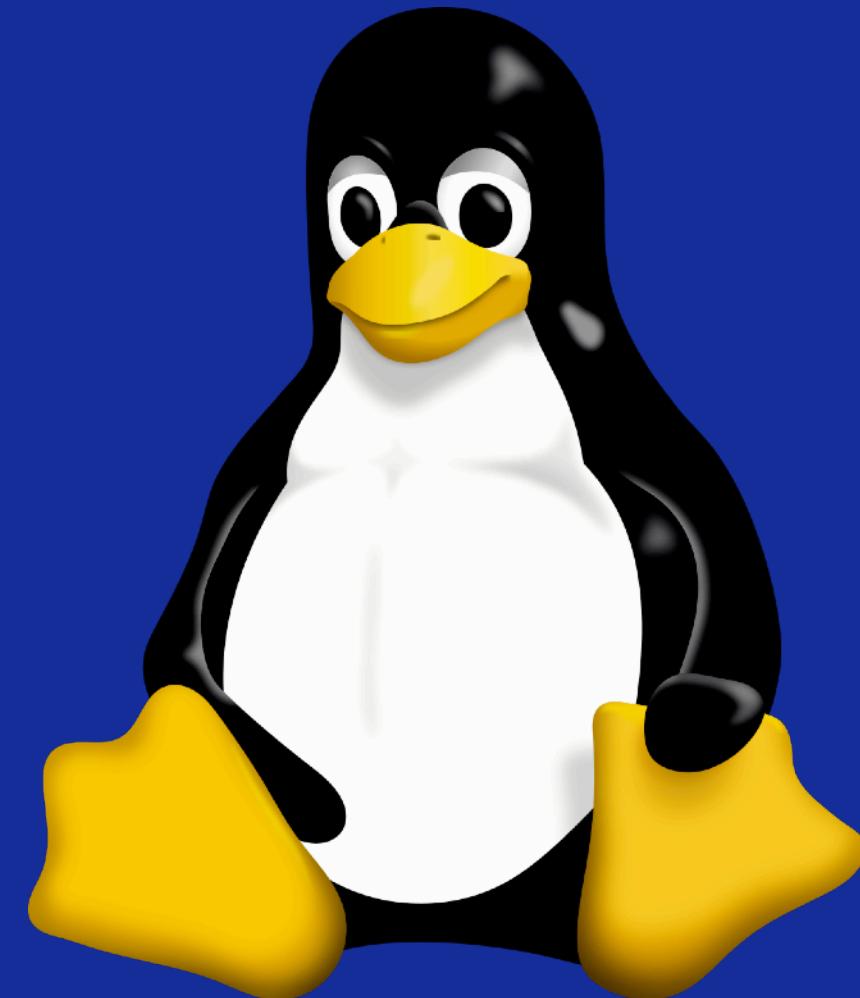
```
SELECT aa_out(data, 40) FROM Logo ORDER BY name;  
aa_out
```

```
SELECT aa_out(data, 40) FROM Logo ORDER BY name;  
aa_out
```

```
-----  
ygaawaag ,n +  
qQQQQWWQQQmc +  
jQQQQQQQQQQQQc +  
]QWWQQQB$QQQQ +  
]Faq4QB_wq ]QQQr +  
]L$T3?? )$EjQQQ[ +  
)@` _/_WQQk +  
jWo( __~q/?QQQQp +  
jmD' "+Y" " 4WQQQa +  
.wWD' ]QQQQg, +  
jmQW' )$QWWQQa +  
yQEm( /4Qw9QWL. +  
jEd( WWQk$WQC +  
j@jf ]QBjQQQ, +  
jQkd' ]QfmQQQf +  
yWT$w= - . ]$wggZ$' +  
-$J' "$wp, :2-3WWQQ@"c +  
)` )$QQa, f "T?" ), +  
)r ?QW[ qf 7^g, +  
j' ]g, _amQf 7^-)' +  
]( ]QmgysmQQQQf y_7^' +  
W~~~?-- , .UT?????TTVc .p^t +  
?~--" ~uw``
```

```
SELECT aa_out(data, 40) FROM Logo ORDER BY name;  
aa_out
```

```
-----  
ygaawaag ,n +  
qQQQQWWQQQmc +  
jQQQQQQQQQQQQc +  
]QWWQQQB$QQQQ +  
]Faq4QB_wq ]QQQr +  
]L$T3?? )$EjQQQ[ +  
)@` _/ )WQQk +  
jWo( __~q/?QQQQp +  
jmD' "+Y" " 4WQQQa +  
.wWD' ]QQQQg, +  
jmQW' )$QWWQQa +  
yQEm( /4Qw9QWL. +  
jEd( WWQk$WQC +  
j@jf ]QBjQQQ, +  
jQkd' ]QfmQQQf +  
yWT$w= - . ]$wggZ$' +  
-$J' $"wp, :2-3WWQQ@"c +  
)` )$QQa, f "T?" ), +  
)r ?QW[ qf 7^S, +  
j' ]g, _amQf 7^-) +  
]( ]QmgysmQQQQf y_7^' +  
W~~~?-- , .UT??????TTVc p^t +  
?~--" ~uw``
```

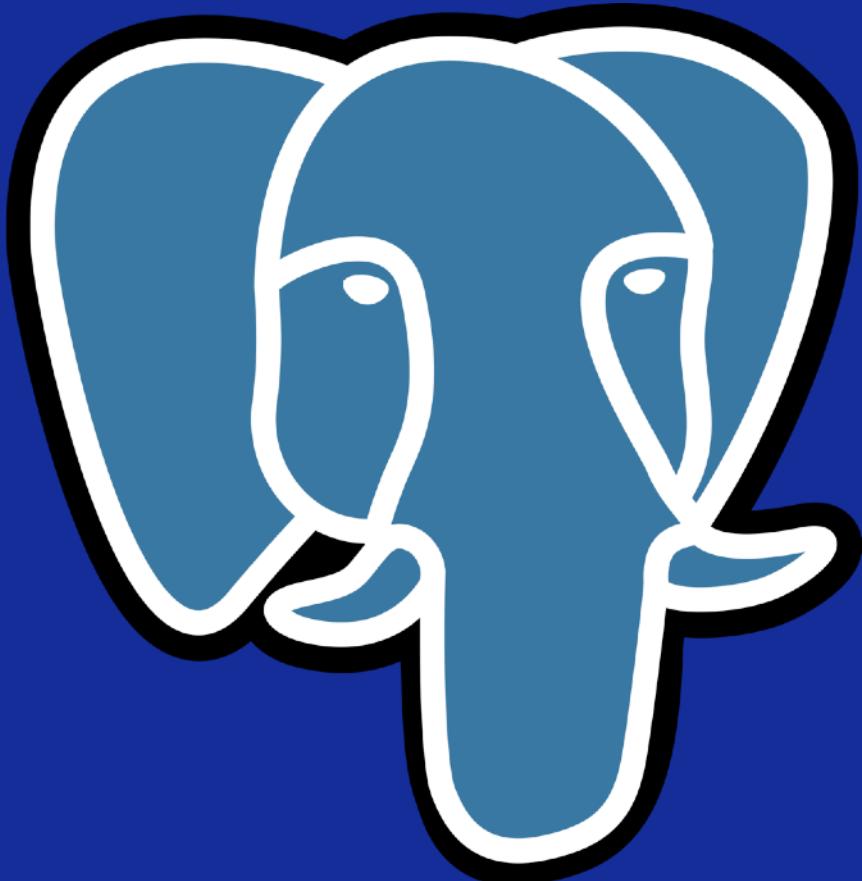


```
SELECT aa_out(data, 40) FROM Logo ORDER BY name;  
aa_out
```

```
-----  
ygaawaag , n + . . ----, - .-g----gaaaagg, - +  
qQQQQWWQQQmc + yawTTYYYYTVSwwVTYYYYTVWwc | o*YYAwg, +  
jQQQQQQQQQQQQc + jm!mWWWWWWQmYKwmW##mWwmITmaWmWmWYmp +  
]QWWQQQB$QQQQ + .m1WQWWWWWWQ1WWWQWWQWWWWWW9m%WWWW )Q +  
]Faq4QB_wq ]QQQr + ]WWWWWWQWWWWmk#WWWWWWWWQWWWWWW?m%WWWWQQt +  
]L$T3?? )$EjQQQ[ + ]QWWWWWWQWWQaQWWWWWWQayXXQkWWQjE +  
)@` _/ )WQQk + :WcWWQWWWWiQYowXS$p#WQWWQxWIT( dkmWQm( +  
jWo( __~q/?QQQQp + ]kQWWWWQWWQWWQWm )Q#WWWWWW )QpQQWcQWjE +  
jmD' "+Y" " 4WQQQa + ]Q%WWWWWWQ { QWWWWWiQmQWWQWQ3mWQdkmx@' +  
.wWD' ]QQQQg, + 3kmWWQWW { QmQWWQjDWWWWWWW9m>dkxW' +  
jmQW' )$QWWQqa + W$cQWWWWQ$p#WwXW( QWWQWWWW3mQbZ' +  
yQEm( /4Qw9QWL . + 4QmQQWWWW$mwaQC>QWQWWWWQWW3QD' =y . , +  
jEd( WWQk$WQc + )m%WWwsyY' )qDY$zQWWWWQWWQx@!YTvvQD' +  
j@jf ]QBjQQQ, + "$wawPsyZT*swwkWWWWWWWWdDY2Y??~ +  
jQkd' ]QfmQQQf + ?`` W"??~ ]kWWWWQWWQmf +  
yWT$w= - . ]$wggZ$' + jkmWWWWQWWQ' +  
-s-J' $"wp, :2-3WWQQ@"c + ]kQWQWWQWWjE +  
]` )$Qqa, f "T?"` ), + jQWWWWWWQm[ +  
)r ?QW[ qf 7^S,+ ?maiiawd! +  
j ]g, _amQf + /"??~  
](` ]QmgysmQQQQf y_7^' +  
W~~~?-- , .UT?????TTVc ~uw` .p^t +
```

(2 rows)

```
SELECT aa_out(data, 40) FROM Logo ORDER BY name;  
aa_out
```



(2 rows)

```
. .----, -.-g----gaaaagg, -  
yawTTYYYYTVSwwVTYYYYTVWwc|o*YYAwg, +  
jm!mWWWWWWQmYKwmW##mWwmITmaWmWmWYmp +  
.m1WQWWWWWWQQ1WWWQWWQWWWWWW9m%WWWW )Q +  
]WWWWWWQWWWWmk#WWWWWWWWQWWWWWW?m%WWWWQt +  
]QWWWWWWQWWQawwwqaQQWWWWQayXXQkWWQjE +  
:WcWWQWWWWiQYoWX$#p#WQWWQxWIT(dkmWQm( +  
]kQWWWWQWQQQWQWm)Q#WWWWWW)QpQQWcQWjE +  
]Q%WWWWWWQ{QWWWWiQmQWWQWQ3mWQdkmx@' +  
3kmWWQWW{QmQWWQjDWWWWWWW9m>dkxW' +  
W$cQWWWWQQ$p#WwXW(QWWQWWWW3mQbZ' +  
4QmQQWWWW$mwaQC>QWQWWWWQWW3QD'=y., +  
)m%WWwsyY')qDY$zQWWWWQWWQx@!YTVVQD' +  
"$wawPsyZT*swwkWWWWWWWWdDY2Y??~ +  
?"` W"??"??~]kWWWWQWWQmf +  
 jkmWWWWWWQQQ' +  
]kQWQWQWWjE +  
jQWWWWWWQm[ +  
?maiiawd! +  
/"?"~ +
```

```
SELECT aa_out(data, 40) FROM Logo ORDER BY name;
-----
```

```
ygaawaag, n          +
qQQQQWWQQQmc         +
jQQQQQQQQQQQQC       +
]QWWQQQB$QQQQ        +
]Faq4QB_wq ]QQQr     +
]L$T3?? )$EjQQQ[    +
)@`      _/)WQQk     +
jWo( __~q/?QQQQp    +
jmD' "+Y" " 4WQQQa +
.wWD'                +
jmQW'                +
yQEm(               +
jEd(                +
j@jf                +
jQkd'                +
yWT$w=              +
-$J'   "$wp,         +
)`      )$QQa,        +
)r      ?QW[          +
j'      ]g,           +
](`      ]QmgysmQQQQf +
W~~~?-- ,`~--"^     +

```

```
....., -.-g----gaaaagg, - +
yawTTYYYYTVSwvTVYYYYTVWwc|o*YYAwg, +
jm!mWWWWWWQmYKwmW##mWwmITmaWmWmWYmp +
.m1WQWWWWWWQ1WWWQWWQWWWWWW9m%WWWW )Q +
]WWWWWWQWWWWmk#WWWWWWWWQWWWWWW?m%WWWWQQt +
]QWWWWWWQWWQaQWWWWWWQayXXQkWWQjE +
:WcWWWWWWiQYowXS$p#WQWWQxWIT(dkmWQm( +
]kQWWWWQWWQWWQWm)Q#WWWWWW)QpQQWcQWjE +
]Q%WWWWWWQ{QWWWWWWiQmQWWQWQ3mWQdkmx@` +
3kmWWQWW{QmQWWQjDWWWWWWW9m>dkxW' +
W$cQWWWWQ$p#WwXW(QWWQWWWW3mQbZ' +
4QmQQWWWW$mwaQC>QWWQWWQWW3QD'=y., +
)m%WWwsyY')qDY$zQWWWWQWWQx@!YTvvQD' +
"$wawPsyZT*swwkWWWWWWWWdDY2Y??~ +
?"` W"??"`~]kWWWWQWWQmf +
jkmWWWWWWQQQ' +
]kQWQWQWWjE +
jQWWWWWWQm[ +
?maiiawd! +
"/?"`~
```

(2 rows)

Аккуратно, сейчас включится свет

Делаем анимацию: \watch 1

Делаем анимацию: \watch 1

1. Создаём функцию, итерирующуюся по строкам таблицы и возвращающую текущую строку. При достижении конца таблицы начать с начала.
2. Заполняем таблицу отдельными «кадрами» будущей анимации.

Делаем анимацию: \watch 1

```
data
-----
+
+
+
+
+
DVD
(1 row)
```

Делаем анимацию: \watch 1

```
data  
-----  
+  
+  
+  
DVD +  
  
( 1 row )
```

Делаем анимацию: \watch 1

```
data
-----
+  
+  
+  
+  
DVD
```

(1 row)

Делаем анимацию: \watch 1

```
data
-----
+-----+
| +   |
| +   |
| +   |
| +   |
+-----+
DVD
( 1 row )
```

Делаем анимацию: \watch 1

```
data
-----
DVD      +
+
+
+
+
( 1 row )
```

Делаем анимацию: \watch 1

```
data
-----
+-----+
| DVD |
+-----+
( 1 row )
```

Делаем анимацию: \watch 1

```
data  
-----  
+  
+  
+  
+  
DVD  
  
( 1 row )
```

Делаем анимацию: \watch 1

```
data
-----
+  
+  
+  
DVD +  
  
( 1 row )
```

Делаем анимацию: \watch 1

```
data
-----
+
+
+
+
+
DVD
(1 row)
```

Делаем анимацию: \watch 1

```
data  
-----  
+  
+  
+  
DVD +  
  
( 1 row )
```

Делаем анимацию: \watch 1

```
data
-----
+-----+
| +   |
| +   |
| DVD + |
|       +|
+-----+
(1 row)
```

Делаем анимацию: \watch 1

```
data
-----
+-----+
| DVD |
+-----+
( 1 row )
```

Делаем анимацию: \watch 1

```
data
-----
DVD+
+
+
+
( 1 row )
```

Делаем анимацию: \watch 1

```
data  
-----  
DVD+  
+  
+  
+( 1 row)
```





Что общего у...



DOOM

pg_doom

pg_doom

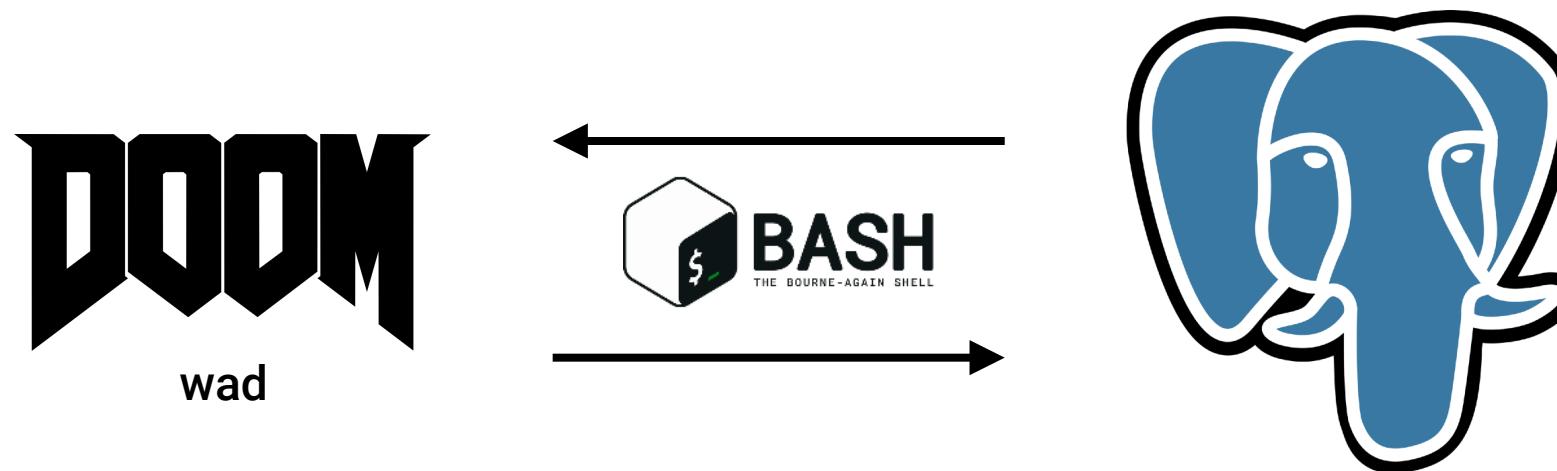
Исходный код расширения для PostgreSQL будет состоять из:

- файла с метаданными расширения — `pg_doom.control`;
- файлов с SQL кодом инициализации расширения в базе — `pg_doom--1.0.sql`;
- файла сборки расширения — `Makefile`;
- файлов с исходным кодом — `pg_doom.c` и другие.

pg_doom

Исходный код расширения для PostgreSQL будет состоять из:

- файла с метаданными расширения — `pg_doom.control`;
- файлов с SQL кодом инициализации расширения в базе — `pg_doom--1.0.sql`;
- файла сборки расширения — `Makefile`;
- файлов с исходным кодом — `pg_doom.c` и другие.

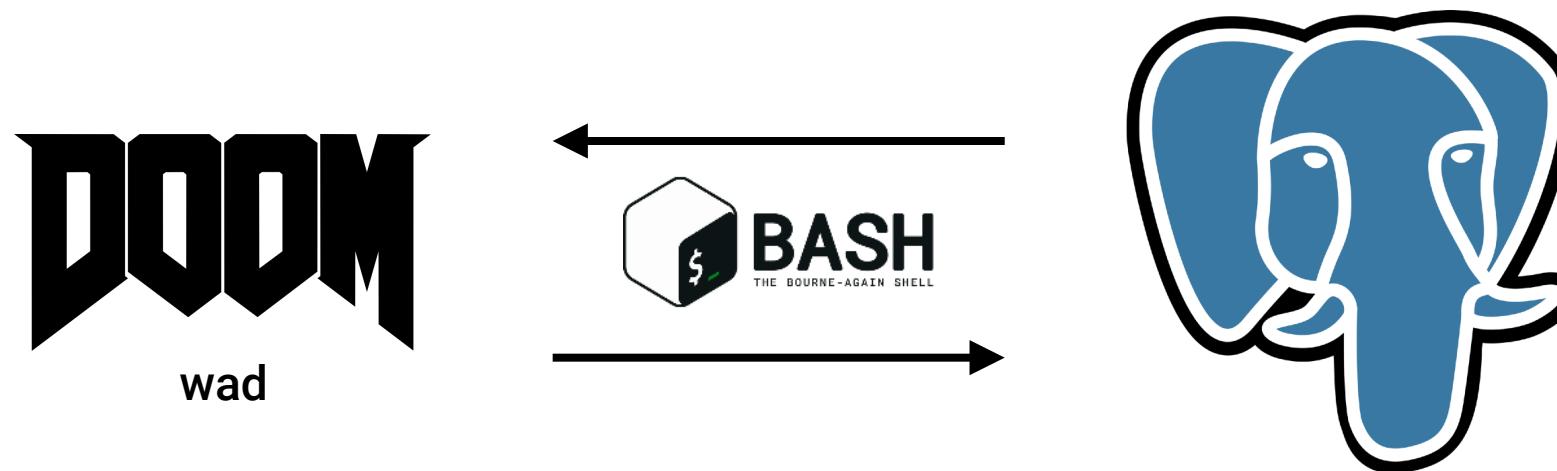


1. передаёт нажатые клавиши
2. получает "картинку" для отображения

pg_doom

Исходный код расширения для PostgreSQL будет состоять из:

- файла с метаданными расширения — `pg_doom.control`;
- файлов с SQL кодом инициализации расширения в базе — `pg_doom--1.0.sql`;
- файла сборки расширения — `Makefile`;
- файлов с исходным кодом — `pg_doom.c` и другие.



1. передаёт нажатые клавиши
2. получает "картинку" для отображения

ВЫ НЕ ПОНЯЛИ МАЕТЕ

ЭТО

ДРУГОЕ

Чудеса расширяемости



TIME~~S~~CALE

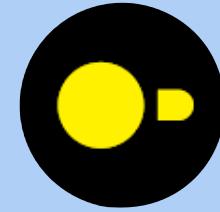


хранилища-расширения
PostgreSQL



Cedar

СУБД с
синтаксисом
PostgreSQL



DuckDB

duckdb-doom

duckdb-doom

- The Database is the World

```
-- MAP: Creating a 16x16 world with walls around the edges
CREATE TABLE map(x INT, y INT, tile CHAR);
INSERT INTO map (x,y,tile)
  SELECT x,y, '#' FROM range(0,15) AS t(x) CROSS JOIN (VALUES(0),(15)) AS y(y)
UNION ALL
  SELECT x,y, '#' FROM (VALUES(0),(15)) AS x(x) CROSS JOIN range(1,14) AS t(y);
INSERT INTO map (x,y,tile)
  SELECT x,y,'.' FROM range(1,14) AS t1(x) CROSS JOIN range(1,14) AS t2(y);
-- Add some interior walls
UPDATE map SET tile = '#' WHERE (x,y) IN ((5,5),(6,5),(7,5),(8,5),(5,10),(6,10),(7,10),(8,10));

-- PLAYER: Starting position in the middle of the map
CREATE TABLE player(x DOUBLE, y DOUBLE, dir DOUBLE, icon CHAR DEFAULT '@');
INSERT INTO player(x,y,dir) VALUES (8.5,8.5,0);
```

duckdb-doom

- The Database is the World
- SQL Dictates Reality

```
-- Bullet physics and collision in pure SQL
-- Move all bullets forward based on their direction vectors
UPDATE bullets SET x = x+dx, y = y+dy;

-- Delete bullets that hit walls
DELETE FROM bullets b
WHERE EXISTS (
    SELECT 1 FROM map m
    WHERE m.x = CAST(b.x AS INT)
    AND m.y = CAST(b.y AS INT) AND m.tile = '#'
);

-- Create a temporary table for bullet-enemy collisions
CREATE TEMP TABLE collisions AS
SELECT b.id AS bullet_id, e.id AS enemy_id
FROM bullets b JOIN enemies e
ON CAST(b.x AS INT) = CAST(e.x AS INT) AND CAST(b.y AS INT) = CAST(e.y AS INT);

-- Remove hit enemies and their bullets
DELETE FROM enemies WHERE id IN (SELECT enemy_id FROM collisions);
DELETE FROM bullets WHERE id IN (SELECT bullet_id FROM collisions);
```

duckdb-doom

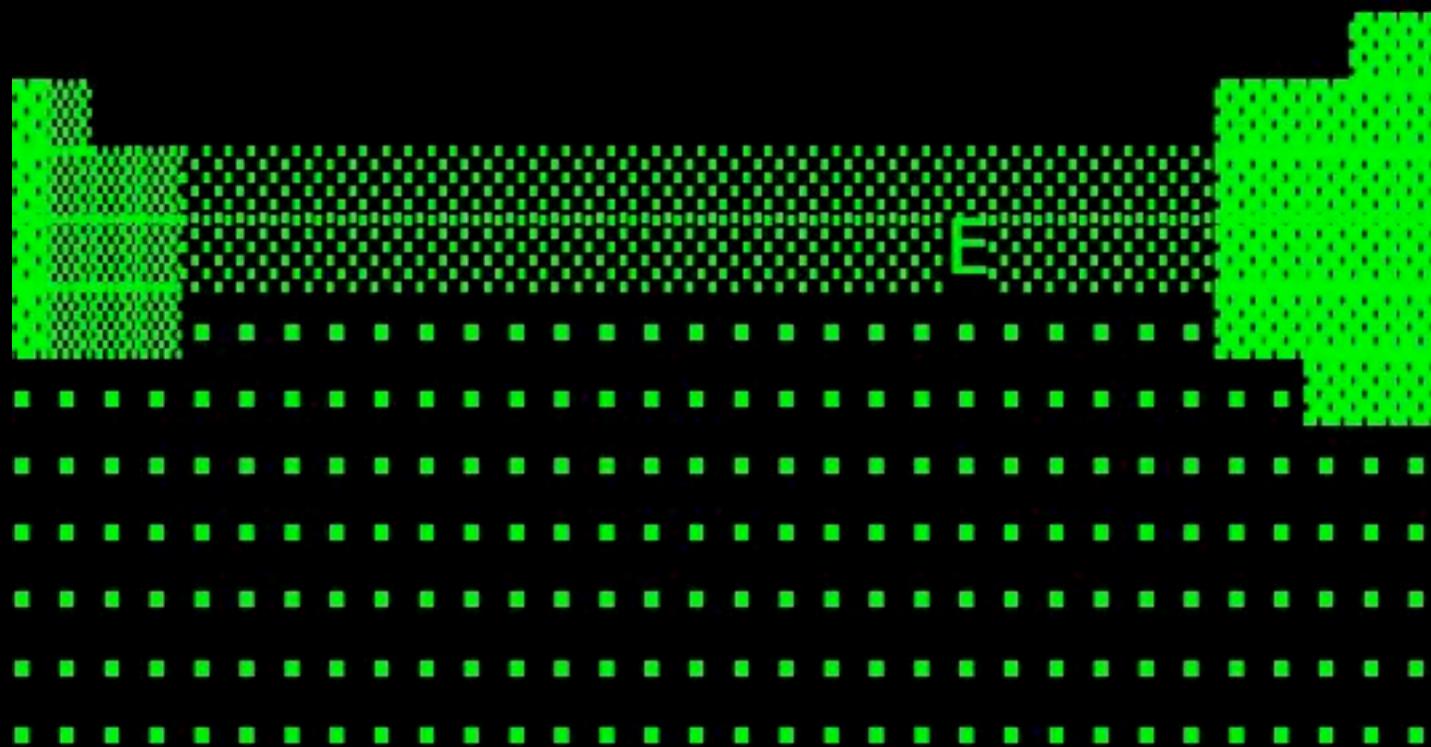
- The Database is the World
- SQL Dictates Reality
- The Renderer is a SQL

SQL-представление render_3d_frame фактически выполняетрейкастинг и рендерит 3D-сцену

- использует рекурсивные СТЕ-выражения для испускания лучей для каждого столбца экрана
- вычисляет расстояние до стен (с коррекцией эффекта «рыбьего глаза»!)
- определяет высоту среза стены для этого столбца
- затем использует string_agg для сшивания символов (' ', ., █, ████, ███, █)
- для каждой строки итогового текстового фрейма

duckdb-doom

- The Database is the World
- SQL Dictates Reality
- The Renderer is a SQL
- JavaScript Glues It Together (and Handles Sprites)
 - обрабатывает ввод с клавиатуры
 - запускает игровой цикл setInterval
 - вызывает SQL-представление для получения фонового кадра
 - извлекает позиции сущностей (пули/врага) и предварительно рассчитанные расстояния до стен (из другого SQL-представления!)
 - выполняет быструю проверку Z-буфера в JS, чтобы определить, находится ли спрайт ближе стены в проецируемом столбце экрана и отображает его на фоновом кадре, если это так
 - выводит полученный текст в тег <pre>



duckdb-doom

- The Database is the World
- SQL Dictates Reality
- The Renderer is a SQL
- JavaScript Glues It Together (and Handles Sprites)
 - обрабатывает ввод с клавиатуры
 - запускает игровой цикл setInterval
 - вызывает SQL-представление для получения фонового кадра
 - извлекает позиции сущностей (пули/врага) и предварительно рассчитанные расстояния до стен (из другого SQL-представления!)
 - выполняет быструю проверку Z-буфера в JS, чтобы определить, находится ли спрайт ближе стены в проецируемом столбце экрана и отображает его на фоновом кадре, если это так
 - выводит полученный текст в тег <pre>

ВЫ НЕ ПОНИМАЕТЕ

ЭТО

ДРУГОЕ

DOOMQL

- ВСЁ на SQL
- DuckDB-Doom: частота 8 кадров/сек, маленькая область просмотра

DOOMQL

- ВСЁ на SQL
- DuckDB-Doom: частота 8 кадров/сек, маленькая область просмотра
- Настоящие спрайты с элементом прозрачности
- Многопользовательская игра

Меня как настоящего нерда очаровало эмпирическое сходство между сервером БД и традиционным игровым сервером. Основное назначение БД – синхронизировать состояние, разделяемое между множеством клиентов. Благодаря изоляции транзакций, каждый из игроков имеет непротиворечивое представление об игровом мире, независимо от того, что делают другие клиенты. Почему бы на это не положиться?

DOOMQL

- ВСЁ на SQL
- DuckDB-Doom: частота 8 кадров/сек, маленькая область просмотра
- Настоящие спрайты с элементом прозрачности
- Многопользовательская игра
- Получится ли ускорить, переключившись на CedarDB?

Хотелось бы мне вам соврать, что мне всё это удалось — и всё благодаря тому, как хороша из себя база данных CedarDB. Но, честно говоря, мой внутренний технарь просто хотел выкрутить все регуляторы до отказа и проверить, что в итоге откажет.

DOOMQL

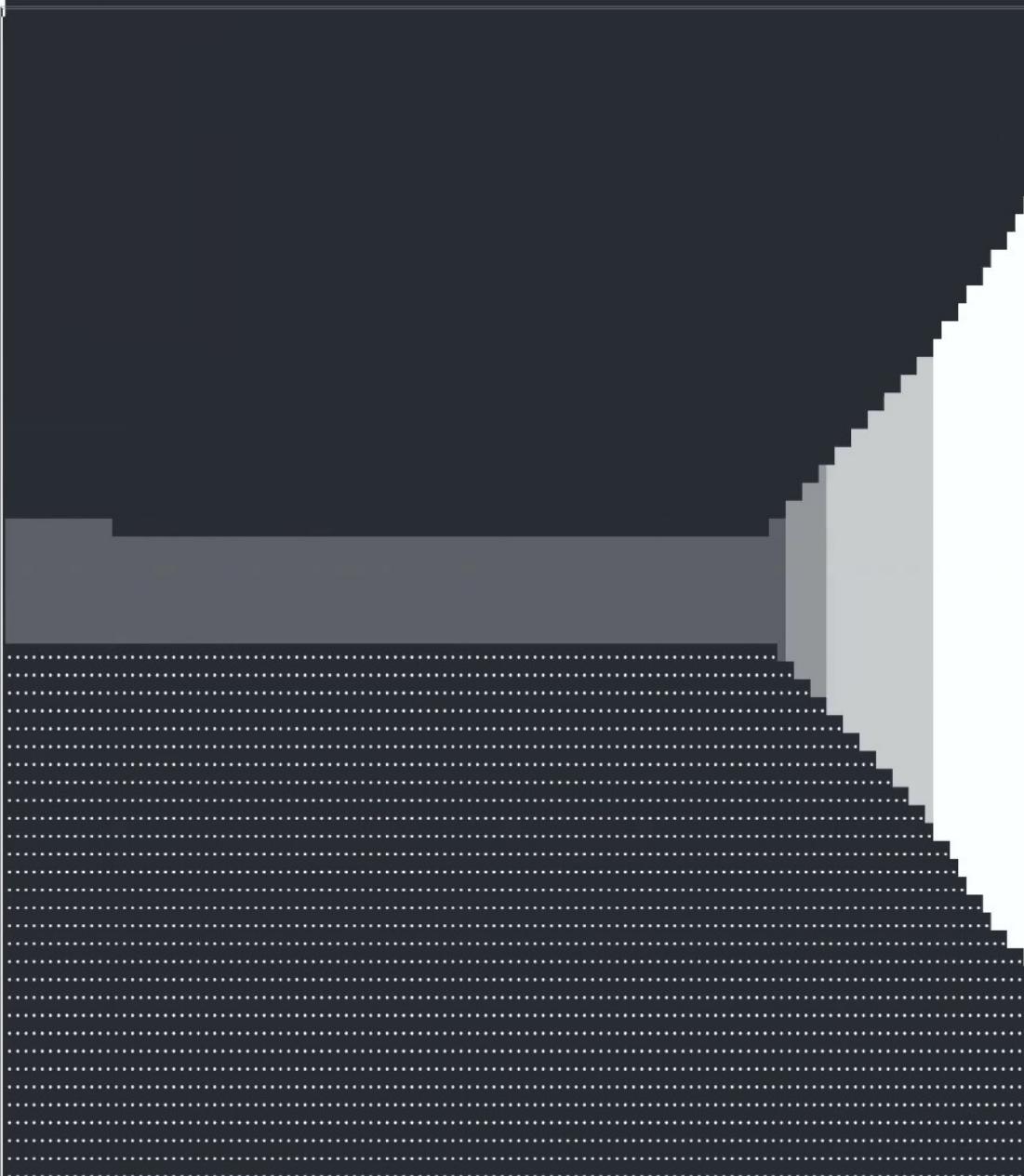
- Состояние хранится в таблицах (карта, игроки, мобы, входные данные, конфиги, спрайты, ...)
- Рендеринг осуществляется через стек представлений SQL, реализующих рейкастинг и проецирование спрайтов.
- Игровой цикл заключён в миниатюрном шелл-скрипте, выполняющем SQL-файл ~ 30 раз в секунду.
- Клиент написан примерно в 150 строках Python: он обращается за вводом и запрашивает из базы данных 3D-представление для вашей игры.
- Можно играть, видеть других игроков и даже жульничать (отправляя необработанный SQL).

WASD to move, X to shoot, Q to quit,
1-9 to switch view to player with that ID, 0 to watch yourself

Viewing player 1: Lukas

x=14.43, y=2.63, dir=-9.599999999999998

--- Game View ---



##| 1: Lukas (L) score: 0 HP: [██████████] 100 AMMO: *****
##| 4: Nico (K) score: 0 HP: [██████████] 100 AMMO: *****

Не doom-ом единственным

Люди, которые
делают игры про
программирование



Люди, которые
играют в игры про
программирование



Score: 0 / Lines: 0 / Level: 1

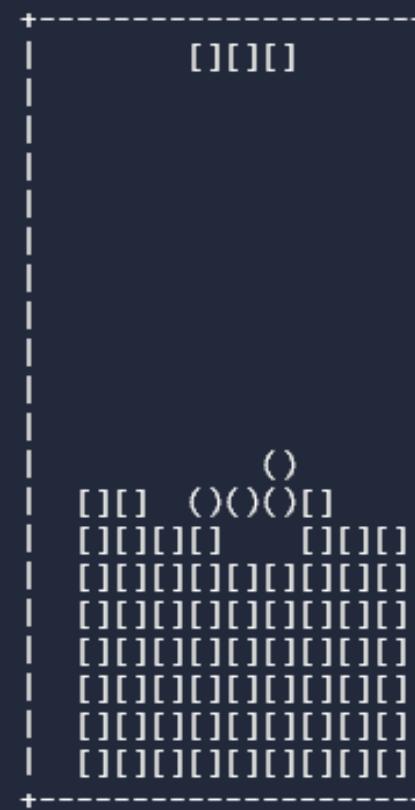
Next: [][][][]



```
> docker exec -it pg ./tetris-sql/input.py
Connecting to localhost:5432/postgres ... connected.
Controls:
  Arrow keys/WASD - move
  Space - hard drop
  P - pause (move/hard drop to unpause)
  Q - stop the input script
```

Score: 0 / Lines: 0 / Level: 1

Next: [][][][]



```
> docker exec -it pg ./tetris-sql/input.py
Connecting to localhost:5432/postgres ... connected.
Controls:
  Arrow keys/WASD - move
  Space - hard drop
  P - pause (move/hard drop to unpause)
  Q - stop the input script
```

PostGIS

**Расширение PostgreSQL, поддержку работы с геоданными:
координатами, линиями, полигонами и другими объектами на карте**

- Где ближайший объект, удовлетворяющий условию?
- Входит ли объект в зону?
- Где пересекаются границы областей?



PostGIS

QGIS, gvSIG, uDig





atlasofthrones.com

The North

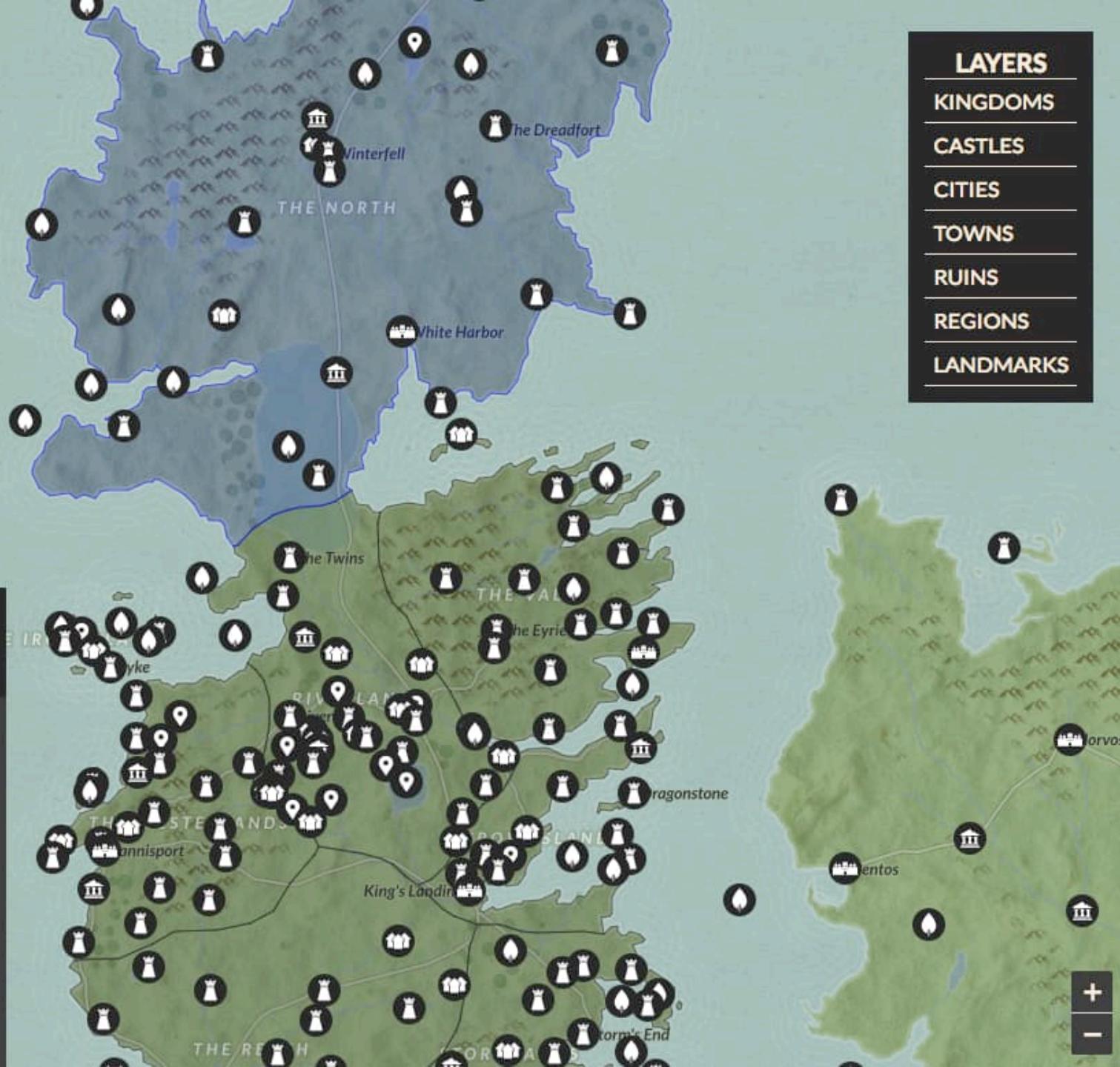
KINGDOM

Size Estimate - 2.621.185 km²

Number of Castles - 13

Summary

The north or the North is one of the constituent regions of Westeros and was a sovereign nation ruled by Kings in the North before Aegon's Conquest. The largest region of the Seven Kingdoms, the dominion of House Stark extends from the border of the New Gift, which is controlled by the Night's Watch, to the southern edge of



SQL

speak() SQL FROM my_heart

DISCLAIMER: автор презентации не несёт ответственности за попытку признаться в чувствах ЭТИМ

speak() SQL FROM my_heart

DISCLAIMER: автор презентации не несёт ответственности за попытку признаться в чувствах ЭТИМ

-- Love real

```
DECLARE @love AS INT;
```

-- Set love equals one

```
SET @love = 1;
```

-- While love more than zero

```
WHILE @love > 0
```

```
BEGIN
```

-- Set love: love plus one

```
SET @love = @love + 1;
```

```
END
```

speak() SQL FROM my_heart

DISCLAIMER: автор презентации не несёт ответственности за попытку признаться в чувствах ЭТИМ

```
CREATE OR REPLACE FUNCTION my_love()
RETURNS TEXT AS $$  
DECLARE
    love INTEGER := 1;
BEGIN
    -- Бесконечный цикл
    WHILE love > 0 LOOP
        love := love + 1;
    END LOOP;

    EXCEPTION
        WHEN NUMERIC_VALUE_OUT_OF_RANGE THEN
            RETURN 'Love reached maximum';
END;
$$ LANGUAGE plpgsql;
```

speak() SQL FROM my_heart

DISCLAIMER: автор презентации не несёт ответственности за попытку признаться в чувствах ЭТИМ

```
CREATE OR REPLACE FUNCTION my_love()
RETURNS TEXT AS $$  
DECLARE
    love INTEGER := 1;
BEGIN
    -- Бесконечный цикл
    WHILE love > 0 LOOP
        love := love + 1;
    END LOOP;

    EXCEPTION
        WHEN NUMERIC_VALUE_OUT_OF_RANGE THEN
            RETURN 'Love reached maximum';
END;
$$ LANGUAGE plpgsql;
```

speak() SQL FROM my_heart

DISCLAIMER: автор презентации не несёт ответственности за попытку признаться в чувствах ЭТИМ

```
CREATE OR REPLACE FUNCTION my_love()
RETURNS TEXT AS $$  
DECLARE
    love INTEGER := 1;
BEGIN
    -- Бесконечный цикл
    WHILE love > 0 LOOP
        love := love + 1;
    END LOOP;

    EXCEPTION
        WHEN NUMERIC_VALUE_OUT_OF_RANGE THEN
            RETURN 'Love reached maximum'; ≈ 2 min
END;
$$ LANGUAGE plpgsql;
```

speak() SQL FROM my_heart

DISCLAIMER: автор презентации не несёт ответственности за попытку признаться в чувствах ЭТИМ

```
CREATE OR REPLACE FUNCTION infinite_love()
```

```
RETURNS TEXT AS $$
```

```
DECLARE
```

```
    love BIGINT := 1;
```

```
BEGIN
```

```
    -- Бесконечный цикл
```

```
    WHILE love > 0 LOOP
```

```
        love := love + 1;
```

```
    END LOOP;
```

```
EXCEPTION
```

```
    WHEN NUMERIC_VALUE_OUT_OF_RANGE THEN
```

```
        RETURN 'Love reached maximum';
```

```
END;
```

```
$$ LANGUAGE plpgsql;
```

Для детей 5-6 лет



Мой детский сад

Учимся играя



Учимся играя

○ SQL Noir

Почувствовать себя детективом из нуарного фильма

Учимся играя

○ SQL Noir

Почувствовать себя детективом из нуарного фильма

○ SQL Island

Выжить на острове после крушения самолёта

Учимся играя

○ SQL Noir

Почувствовать себя детективом из нуарного фильма

○ SQL Island

Выжить на острове после крушения самолёта

○ SQL Murder Mystery

Вы потеряли отчёт с места преступления, который вам передал детектив. Придётся найти его в БД.

○ SQL Squid Game

Да, игра в кальмара. Твой коллега-аналитик уже проиграл

○ Lost at SQL

Спасаем подводную лодку и команду

○ SQLPD

Вы – аналитик SQL-полиции

Учимся играя

○ SQL Noir

Почувствовать себя детективом из нуарного фильма

○ SQL Island

Выжить на острове после крушения самолёта

○ SQL Murder Mystery

Вы потеряли отчёт с места преступления, который вам передал детектив. Придётся найти его в БД.

○ SQL Squid Game

Да, игра в кальмара. Твой коллега-аналитик уже проиграл

○ Lost at SQL

Спасаем подводную лодку и команду

○ SQLPD

Вы – аналитик SQL-полиции

○ Текстовые квесты

DBQuacks или... можете сделать самостоятельно!



Спасибо за внимание!

With @love,
sokolcati@gmail.com
e.sokolova@postgrespro.ru
tg: [@para_para_pam](https://t.me/para_para_pam)

PGMeetup.VLD 2025

