

# **THE FUTURE OF THE POSTGRES EXTENSION ECOSYSTEM**

**DAVID E. WHEELER  
PGXN, TEMBO**



# BEGINNINGS

# **LEGACY**

**Long history of extensibility**

**Two approaches**

**shared\_preload\_libraries**

**Pure SQL (including PLs)**

**A few intrepid adopters**

**PostGIS**

**BioPostgres**

**PL/R**

**PL/Proxy**

**pgTAP**

# **POSTGRES 9.1**

**Dimitri Fontaine adds extensions**

**Key features:**

**Compile and install**

**CREATE/UPDATE/DROP EXTENSION**

**pg\_dump and pg\_restore**

***“PostgreSQL today is not merely  
a database, it’s an application  
development platform.”***

ME, 2010



# PGXN

FEATURES

**Distribution**

**Source Code**

**Registration**

**Namespacing**

**Discovery**

**Search**

**Documentation**

**Tags**

**Installation**

**CLI**

**Compile & install**

**PGXS or Configure**



# PGXN

OUT OF SCOPE

**Binary packaging**

**Binary distribution**

**Defer to apt/yum**

**Developer tooling**

**Build tooling**

**Defer to core (PGXS)**



# PGXN

## HISTORY

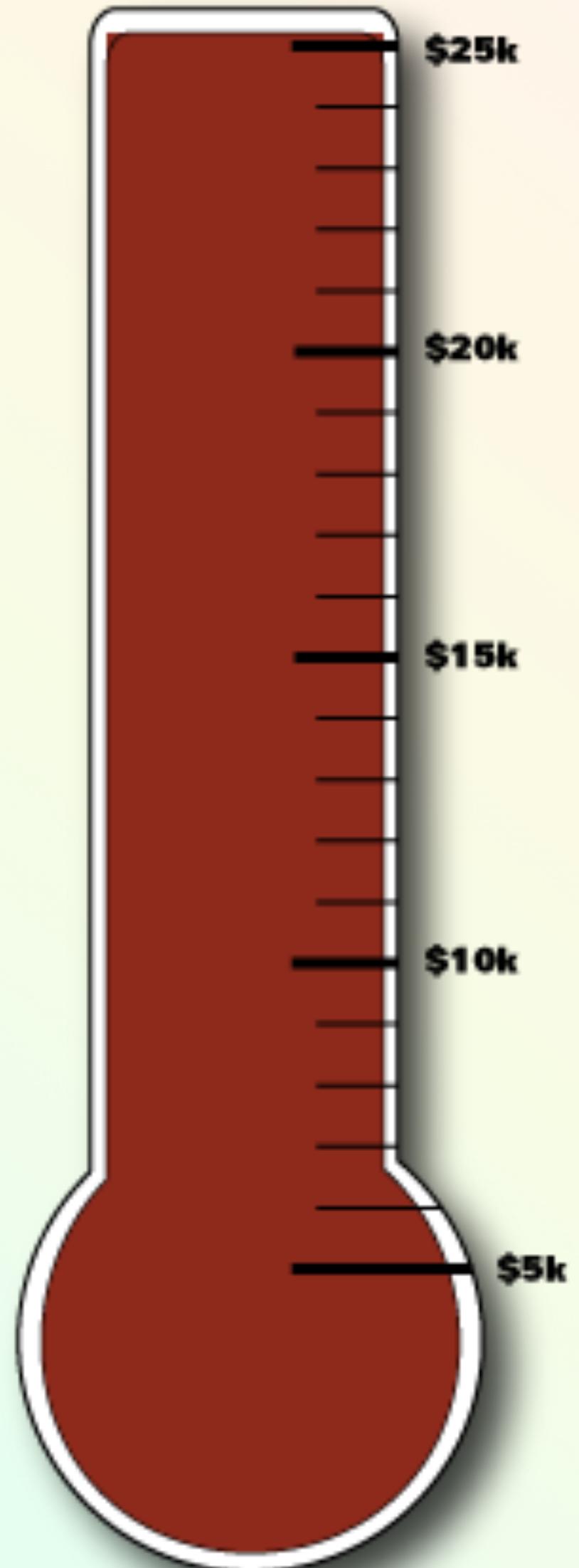
**Started 2010**

**Funded December 2010**

**pgxn.org launched April 2011**

**Client by Daniele Varrazzo April 2011**

**Dev utils by Dickson S. Guedes June 2011**



***“In classic SDLC fashion, the PGXN POC shipped as an MVP and was neglected.”***

ME, JUST NOW



**TIME PASSED...**

# UPTAKE

**PGXN: 396 Extensions/369 Distributions**

**joelonsql/PostgreSQL-EXTENSIONs.md: 1,186**

**CPAN: 217,860 Perl modules/45,137 distributions**

**PyPI: 543,205 Python projects**

**RubyGems: 180,839 Ruby Gems**

**crates.io: 146,401 Rust Crates**



WHY?

# **LOST OPPORTUNITIES**

**POSTGRES EXTENSIONS:**

**Difficult to find and discover**

**Under-documented and difficult to understand**

**Maturity difficult to gauge**

**Hard to configure and install**

**No comprehensive binary packaging**

**Centralized source distribution insufficient**

**Insufficient developer tooling**

The background features a series of overlapping, wavy shapes in shades of teal and dark blue, creating a sense of depth and motion.

# FILLING THE GAPS

# The Database Package Manager

For Postgres Trusted Language Extensions

[Getting started](#)[Documentation](#)

## Download and install

Install extensions via a SQL command

```
select dbdev.install('olirice-index_advisor');
```

PUBLISHER HANDLE

EXTENSION NAME

A Postgres

Get Trunk

**Cargo**

cargo install pg-trunk

**Copy**

Trunk is an open-source package installer and registry for  
PostgreSQL extensions.

**Featured**

7

**Analytics**

13

**Auditing / Logging**

7

**All Extensions**

Search 205 extensions

**Search**

# > extensions, made easy

npm for PostgreSQL

[Brew](#)[Curl](#)

```
brew install pgxman/tap/pgxman 
```



# EMPHASES

Ease of use

Platform neutrality

Stats

Curation

Architecture

x86-64 ⓘ

Operating system

Debian/Ubuntu ⓘ

## Downloads

20 all time downloads

0 downloads in the last 30 days

1 download in the last 90 day

9 downloads in the last 180 days

```
> curl -sfL https://install.pgx.sh | sh
```

```
🎉 pgxman successfully installed
```

```
> pgxman install pgvector
```

The following Debian packages will be installed:  
postgresql-14-pgxman-pgvector=0.5.1

```
> Do you want to continue? [Y/n] y
```

```
pgvector has been successfully installed
```

Featured

7

Analytics

13

Auditing / Logging

7

Change Data Capture

6

Connectors

27

Data / Transformations

49

Debugging

2

Index / Table Optimizations

14

Machine Learning

4

Metrics

18

Orchestration

9

Procedural Languages

16

Query Optimizations

5

Search

11

Security

11

Tooling / Admin

14

# MVPS

**Trunk**

**Pull request integration**

**Postgres 14–16**

**Debian/Ubuntu-only**

**AMD64-only**

**210 Extensions**

**PGXMan**

**Form-based submission**

**Postgres 12–16**

**Debian/Ubuntu-only**

**AMD64 & ARM64**

**Apt-only**

**81 Extensions**

**dbdev**

**TLEs**

**CLI publishing**

**Postgres 11-16**

**No C/pgrx**

**pg\_tle && ptsql-http**

**Count Unknown**

# **STATE OF THE ECOSYSTEM**

# **STATE OF AFFAIRS**

**Despite early promise...**

**No single source**

**Many incomplete attempts**

**Poor discovery**

**Difficult installation**

**Insufficient docs**

**Low adoption**

**But...**

***“There sure has been a lot of  
excitement around  
extensions lately.”***

**ME, AGAIN**

# OPPORTUNITIES (NEW)

# **PGXN V2**

**The New World Order**

**Started this year**

**Made possible by Tembo**

**Consider emerging patterns**

**Meditate on deficiencies**

**Engage deeply with the community**

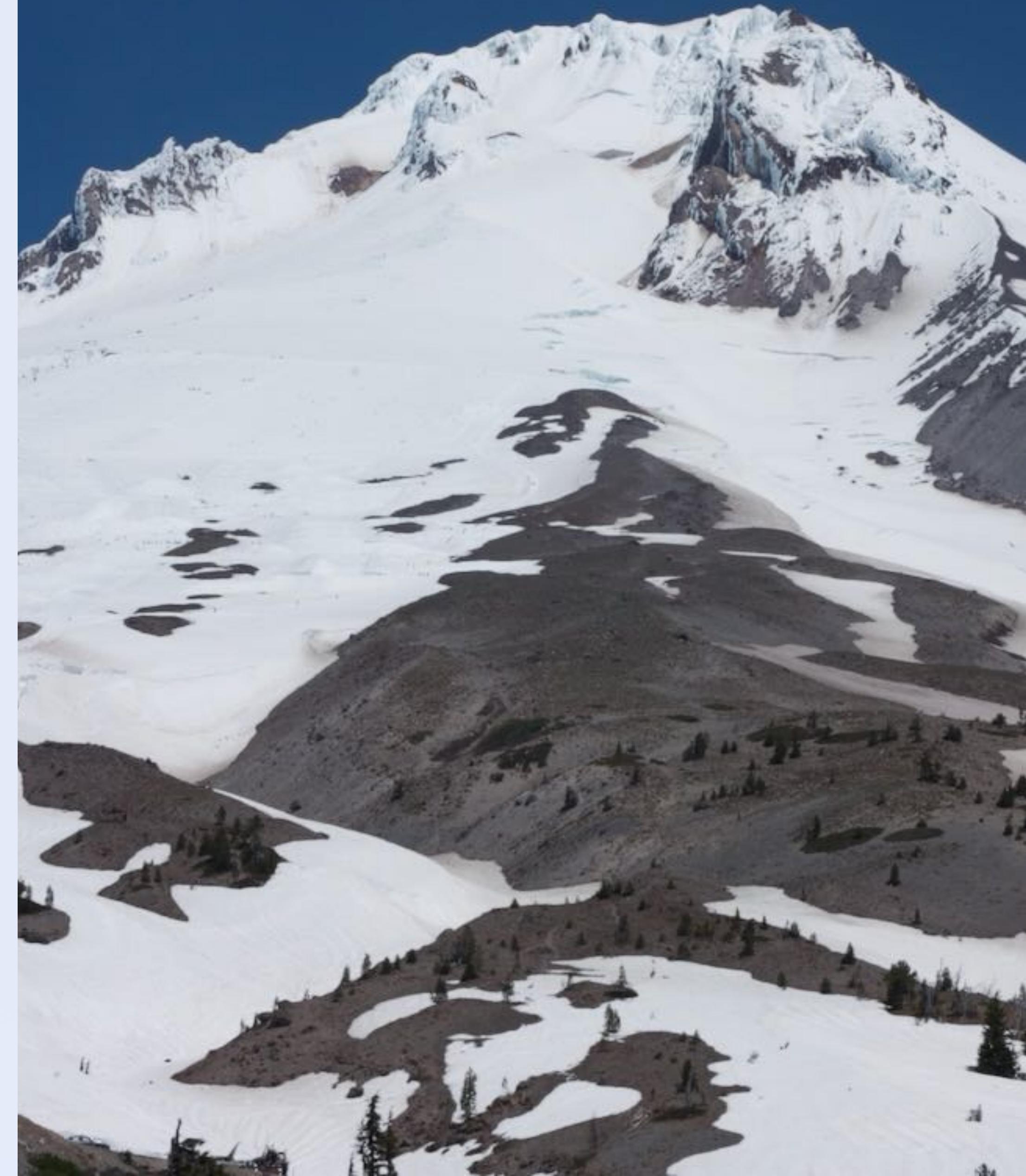
**Design and implement new architecture**

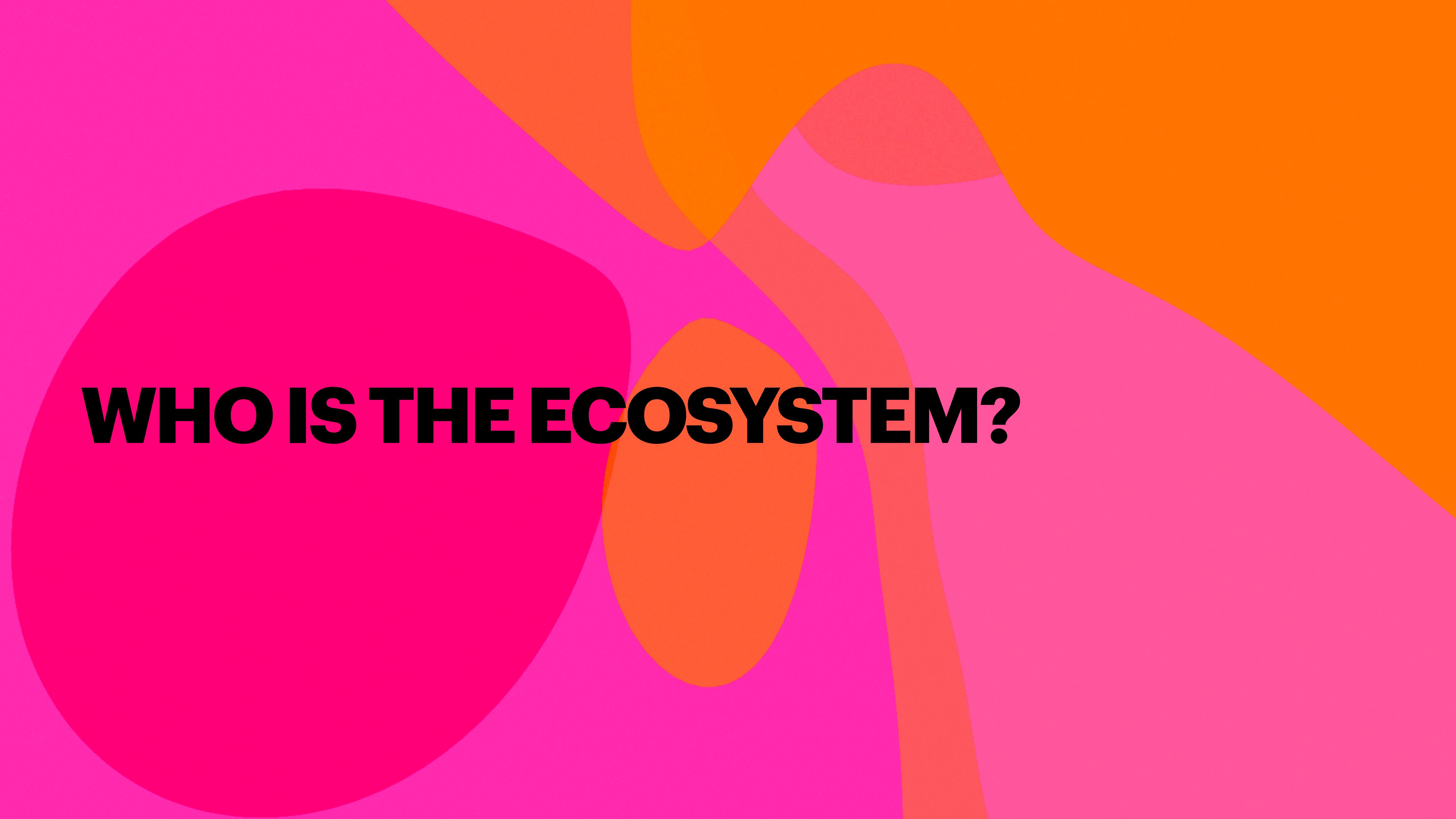
**Serve the ecosystem for next decade**



# **EXTENSION ECOSYSTEM SUMMIT**

**Collaborate to examine the ongoing work on PostgreSQL extension distribution, examine its challenges, identify questions, propose solutions, and agree on directions for execution.**



The background features a complex arrangement of overlapping organic shapes in shades of pink, magenta, and orange. These shapes resemble petals or leaves, creating a sense of depth and movement. The colors transition from bright magenta on the left to a deep burnt orange on the right.

# **WHO IS THE ECOSYSTEM?**



# EXTENSION AUTHORS



# EXTENSION AUTHORS

Needs

Tools to create extensions

Learning materials

Standard patterns

Testing tools

Metadata management

Distribution & discovery

Packaging



# EXTENSION AUTHORS

## Delivery

Documentation

How-tos

Examples

Tutorials

References

Book?

CLI for tasks

Metadata

Testing

Scaffolding

Release

Documentation

CI/CD Automation

Canonical Registry

Discovery

Binary Packaging

Category Curation

Incentives

Quality Metrics

Documentation standards

Test coverage & matrices

# Install Management

```
> pgxn installs ls
```

install	path
homebrew/14.12	/opt/homebrew/Cellar/postgresql@14/14.12
pgenv/16.3	/Users/david/.pgenv/pgsql-16.3
pgenv/17-devel	/Users/david/.pgenv/pgsql-17-devel
postgres.app/16.3	/Applications/Postgres.app/Contents/Versions/16/

```
> pgxn manage pgenv/16.3
```

```
Now managing extensions for pgenv/16.3
```

```
>
```

# Develop CLI

```
› pgxn develop new --extension --language c --vcs github bike
```

Scaffolding C extension “bike”...

⚙️ Create bike.control

- META.json
- bike.control
- Makefile
- README.md

- src/bike.c
- sql/bike--0.0.1.sql
- doc/bike.md

- test/sql/base.sql
- test/expect/base.out

- .github/workflows/ci.yml
- .github/workflows/release.yml

✨ Init repository  
git init bike

# Develop CLI

```
› pgxn develop add-upgrade 0.0.2
```

Adding upgrade from 0.0.1 to 0.0.2...

- 🔥 Rename sql/bike--0.0.1.sql to sql/bike--0.0.2.sql
- ⚙️ Create sql/bike--0.0.1--0.0.2.sql
- 🆙 Update META.json  
bike.control

# Develop CLI

› pgxn develop require 14

Require PostgreSQL 14...

 Update META.json

› pgxn develop require semver 0.32.0

Requiring semver 0.32.0...

 Update META.json  
semver.control

›

# Develop CLI

```
> pgxn develop test --username postgres
```

```
Testing bike 0.0.2 in pgenv/16.3
```

```
make
```

```
  gcc -c -o src/pair.o src/pair.c  
  gcc src/pair.o -o src/pair.so
```

```
make install
```

```
  install -c -m 644 bike.control '/pgsql/share/extension/'  
  install -c -m 644 sql/bike--0.0.2.sql '/pgsql/share/bike/'  
  install -c -m 644 sql/bike--0.0.1--0.0.2.sql '/pgsql/share/bike/'  
  install -c -m 644 src/bike.so '/pgsql/lib/'
```

```
make installcheck PGUSER=postgres
```

```
ok 1          - base           15 ms  
1..1  
# All 1 tests passed.
```

# Develop CLI

› `pgxn develop release`

Releasing bike 0.0.2 on PGXN...

- ✓ Validate META.json... Okay
- 📦 Bundle bike-0.0.2.zip with git archive...
- 🚀 Publish bike-0.0.2.zip on PGXN...

🎉 Done: <https://pgxn.org/theory/bike/0.0.2>

← CI

## ✓ Bump version in README to 0.32.1 #146

### Summary

Jobs

✓ PostgreSQL 16

✓ PostgreSQL 15

✓ PostgreSQL 14

✓ PostgreSQL 13

✓ PostgreSQL 12

✓ PostgreSQL 11

✓ PostgreSQL 10

✓ PostgreSQL 9.6

✓ PostgreSQL 9.5

✓ PostgreSQL 9.4

✓ PostgreSQL 9.3

✓ PostgreSQL 9.2

Triggered via push 2 months ago

 theory pushed · 804f954 main

ci.yml

on: push

Matrix: build

✓ 12 jobs completed

Show all jobs

Annotations

12 warnings



Release on PGXN and GitHub

succeeded last month in 35s

> ✓ Set up job

> ✓ Initialize containers

> ✓ Check out the repo

> ✓ Bundle the Release

> ✓ Release on PGXN

> ✓ Generate Release Changes

> ✓ Create GitHub Release

> ✓ Upload Release Asset

> ✓ Post Check out the repo

> ✓ Stop containers

> ✓ Complete job



# PACKAGERS



# EXTENSION PACKAGERS

**Needs**

**Release notification**

**Build automation**

**Predictable download URLs**

**Flexible sourcing**

**Extensive metadata**

**Feedback loops**



# EXTENSION PACKAGERS

**Delivery**

**Release feeds & notification Webhooks**

**Package on every release**

**Build from source or repackage community binaries**

**Metadata to automate building from source**

**Report build and test failures**

**Submit download stats to root registry**

```
Name:          bike
Version:       0.0.2
Source0:        https://pgxn.org/dist/theory/bike/bike-%{version}.zip
BuildRoot:      %{_tmppath}/%{name}-%{version}-%{release}-root-%(%{_id_u} -n)

%prep
%setup -q -n bike-%{version}

%build
pgxn package build

%install
pgxn package install --destdir $RPM_BUILD_ROOT
%{_fixperms} $RPM_BUILD_ROOT/*
%{_fixperms} $RPM_BUILD_ROOT/*/_data

%check
pgxn package installcheck --username postgres

%clean
rm -rf $RPM_BUILD_ROOT
```



# **ADMINISTRATORS**



## Needs

**Reliable sourcing**

**Extension management**

**Inventory & audit reporting**

**Packaging choice & standardization**

**Trusted package sources**

**Internal packaging options**

**Mirroring & curation**

**Upgrade paths**



## Delivery

### Extension Management

**Fast binary installation**

**No compilation required**

**Use community binary packages**

**Or community OS packages**

**Or build own packages from binaries or source**

**Extension reporting**

**Registry access controls**

**Simplify upgrades (dupe extensions)?**

### Binary Registry Federation

**Run own binary registry**

**Package your own extensions**

**Use the same tooling**

**Gate it by authentication or network**

**Subscribe to commercial registries with additional features**

### Binary Registry Mirroring

**Mirror upstream registry**

**Fast local performance**

# Package Management

```
> pgxn package ls
```

Packages installed in pgdb/16.3:

package	installed	latest	description
core/citext	1.6.0	✓	data type for case-insensitive ch
core/jsonb_plperl	1.0.0	✓	transform between jsonb and plperl
core/unaccent	1.1.0	✓	text search dictionary that remov
theory/semver	0.32.0	0.32.1	Semantic version data type

```
>
```

# Package Management

```
> pgxn upgrade -v theory/semver
```

```
Upgrading theory/semver 0.32.1 in pgdb-vum/16.3...
Found theory-semver-0.32.1+1.pg16+2.linux-arm64.zip
Unpacking theory-semver-0.32.1+1.pg16+1.linux-arm64.zip
Installing lib/semver.so in /usr/lib/postgresql/16/lib
Installing share/semver-0.32.1.sql in /usr/share/postgresql/16/extensions/
Installing share/semver.control in /usr/share/postgresql/16/extensions/
Installing doc/semver.md in /usr/share/doc/postgresql-doc-16
```

```
Done!
>
```

Also macOS,  
\*BSD, Windows,

# Package Management

```
> pgxn package ls
```

Packages installed in pgdb-yum/16.3:

package	installed	latest	description
core/citext	1.6.0	✓	data type for case-insensitive ch
core/jsonb_plperl	1.0.0	✓	transform between jsonb and plper
core/unaccent	1.1.0	✓	text search dictionary that remov
theory/semver	0.32.0	✓	Semantic version data type

```
>
```

theory > **semver**

[users](#) [tags](#) [recent](#)

# semver



**This Release:** semver 0.32.1

**Date:** 2023-08-01

**Status:** Stable

**Other Releases:** 0.32.1 – 2023-08-01 ▾

**Abstract:** A semantic version data type

**Description:** A Postgres data type for the Semantic Version format with support for btree and hash indexing.

**Released By:** theory

**License:** PostgreSQL

**Resources:** [git](#) ♦ [repo](#) ♦ [bugs](#)

**Special Files:** [Changes](#) ♦ [LICENSE](#) ♦ [META.json](#) ♦ [Makefile](#) ♦ [semver.control](#)

**Packagers:** PGDB Yum ♦ PGDB Deb ♦ Trunk ♦ Homebrew ♦ Chocolatey ♦ PGXMan

**Tags:** semantic version ♦ semver ♦ version ♦ version number



# USERS



# USERS

## Needs

**Reliable index of extensions**

**Effective search**

**Curation & categorization**

**Quality signals**

**Version support info**

**Quick & easy installation**

**Informative documentation**



# USERS

## Delivery

### Canonical Registry

Discovery

Quality Signals

Category curation

Links & easy install instructions

### Documentation

Getting started with extensions

How to use community tools

Beautiful extension documentation

### Installation

One-line command

Fast binary installation

Many platforms

Many packages (Postgres installs)

OS packaging support

### Usage

Documentation

Tested examples

Tutorials & references

# Package Management

› pgxn search json

rating	package	version	date	
★★	hadi/json_fdw	1.0.0	2013-11-18	Foreign Data
★★★★★	theory/jsonschema	0.1.0	2024-04-30	JSON Schema
★★★	jma/yamltodb	0.10.0	2022-11-08	Database ch
★★	shanekilkelly/bedquilt	2.1.0	2016-09-12	A json docu
★★★	furstenheim/is_jsonb_valid	0.0.1	2017-08-17	JSONB valid
★★★	postgrespro/jsonb_schema	1.0.0	2020-08-24	Decouple js
★★	kungfury/jsoncdc	0.1.0	2017-11-23	Translates

›

# Package Management

```
> pgxn install -v theory/jsonschema
```

```
Installing theory/jsonschema 0.1.0 into pgenv/16.3.  
Found theory-jsonschema-0.1.1+2.pg16+1.darwin-arm64.zip  
Unpacking theory-jsonschema-0.1.1+2.pg16+1.darwin-arm64.zip  
Installing lib/jsonschema.dylib in ~/.pgenv/pgsql-16.3/lib/  
Installing share/jsonschema-0.1.0sql in ~/.pgenv/pgsql-16.3/share/extension  
Installing share/jsonschema.control in ~/.pgenv/pgsql-16.3/share/extension  
Installing doc/jsonschema.md in ~/.pgenv/pgsql-16.3/share/doc/
```

```
Done!  
>
```

# Package Management

```
> pgxn package ls
```

Packages installed in pgenv/16.3:

package	installed	latest	description
core/citext	1.6.0	✓	data type for case-insensitive ch
core/jsonb_plperl	1.0.0	✓	transform between jsonb and plperl
core/unaccent	1.1.0	✓	text search dictionary that remov
theory/jsonschema	0.1.0	✓	JSON Schema validation functions

```
>
```



# INDUSTRY



# INDUSTRY

**Needs**

**Embrace and extend**

**Extension curation products**

**Extension rating products**

**Linked from root registry metadata**

**Offer extensions for customers**

**Use APIs to provide UX for DBaaS products**

**Immutability**



# INDUSTRY

**Delivery**

**Specialized Registries**

**Additional metadata, stats, reports, signal**

**Binary Registry Federation**

**Curate extensions for customers by copying only approved extensions to own registry**

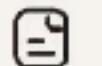
**Use APIs to provide UX for DBaaS products**

**Report back stats**

**Separate mutable extension directory**



chm16



## 0+ Extensions

[Manage](#)    [Explore](#)

Search PGXN extensions

[Featured](#)

7

**adminpack**

v2.1.0

adminpack serves as an auxilliary support to administrative management...

Tooling / Admin

[Metrics](#)

16

**age**

v1.5.0

Leverage a graph database on top of the existing relational database.

Data / Transformations

[Query Optimizations](#)

4

**amcheck**

v1.3.0

The amcheck module provides functions that allow you to verify the lo...

Index / Table Optimizations

[Change Data Capture](#)

2

**argm**

v1.1.2

Argm postgresql extension: argmax/argmin and anyold aggregate functio...

[Orchestration](#)

6

Query Optimizations

[Index / Table Optim...](#)

13

**auto\_explain**

v15.3.0

The auto\_explain module provides a means for logging execution plans ...

[Procedural Languages](#)

14

[Machine Learning](#)

2

**autoinc**

Functions for Autoincrementing

**Relies on****Community APIs**



# THE VISION

# **PGXN V2 FEATURES**

**Canonical source registry**

**Search & discover all extensions**

**Streaming & federation**

**Quality metrics & reports**

**Binary packaging & distribution**

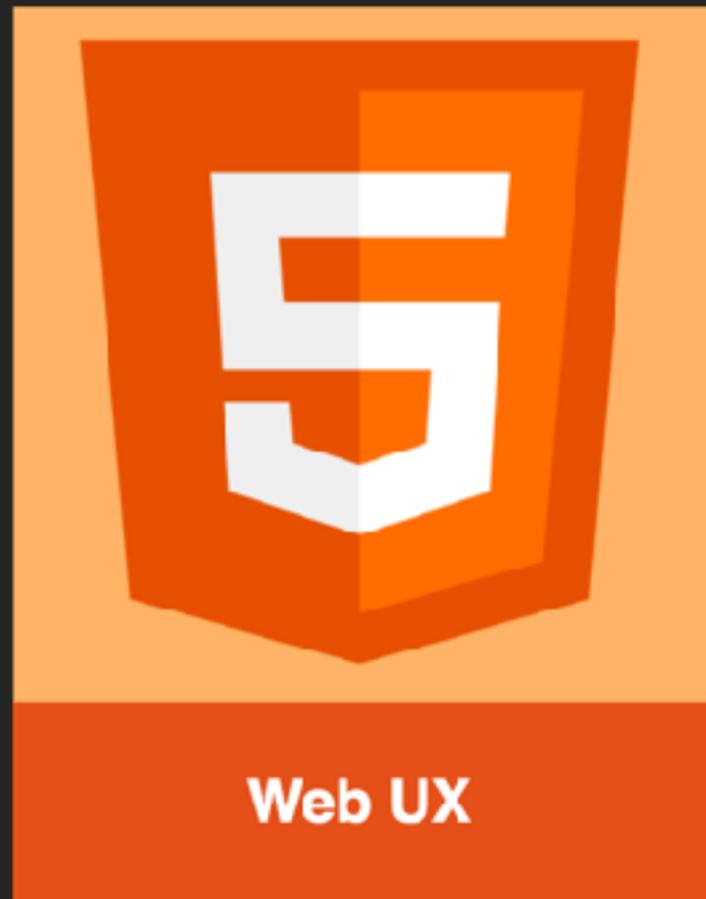
**Multi-platform & multi-version**

**Simple installation & package management**

# ARCHITECTURE SKETCH

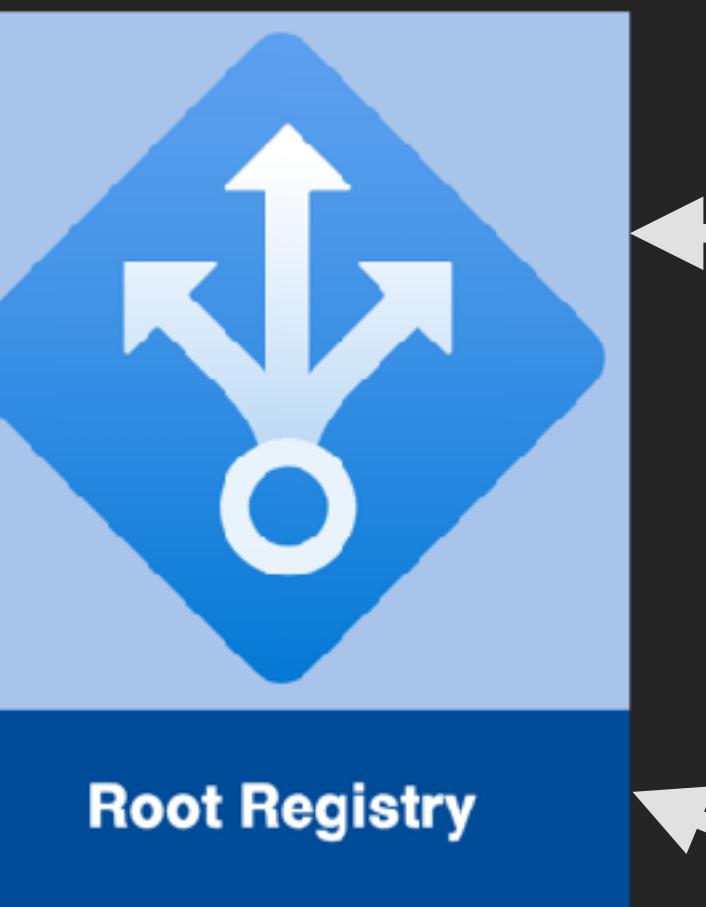
- Search
- Curation
- Docs
- Download
- Badges
- Stats
- Links

- Search
- Download
- Install

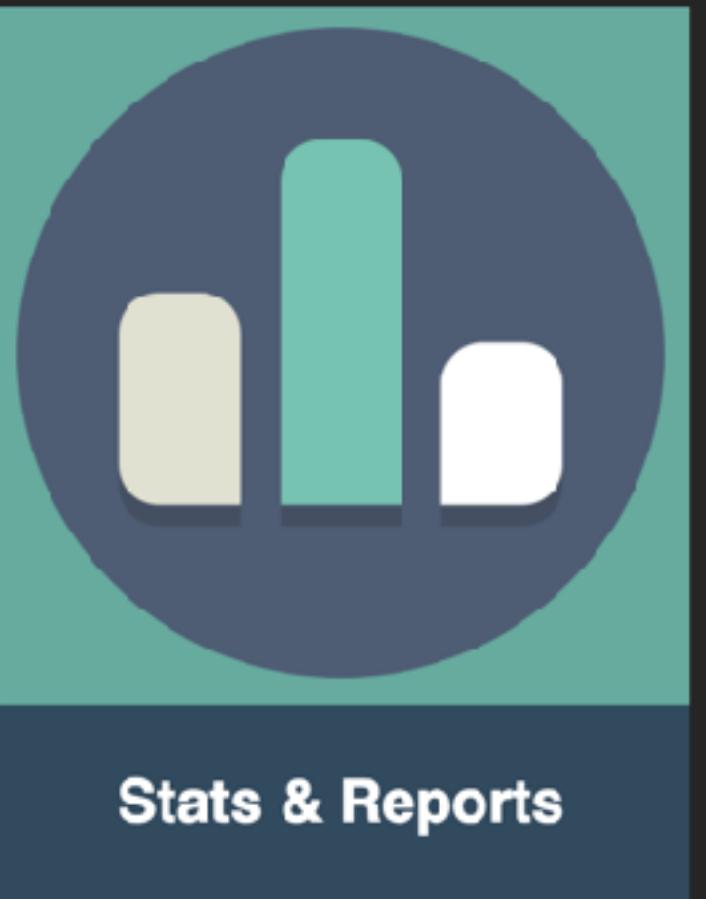


Web UX

- Register
- Release

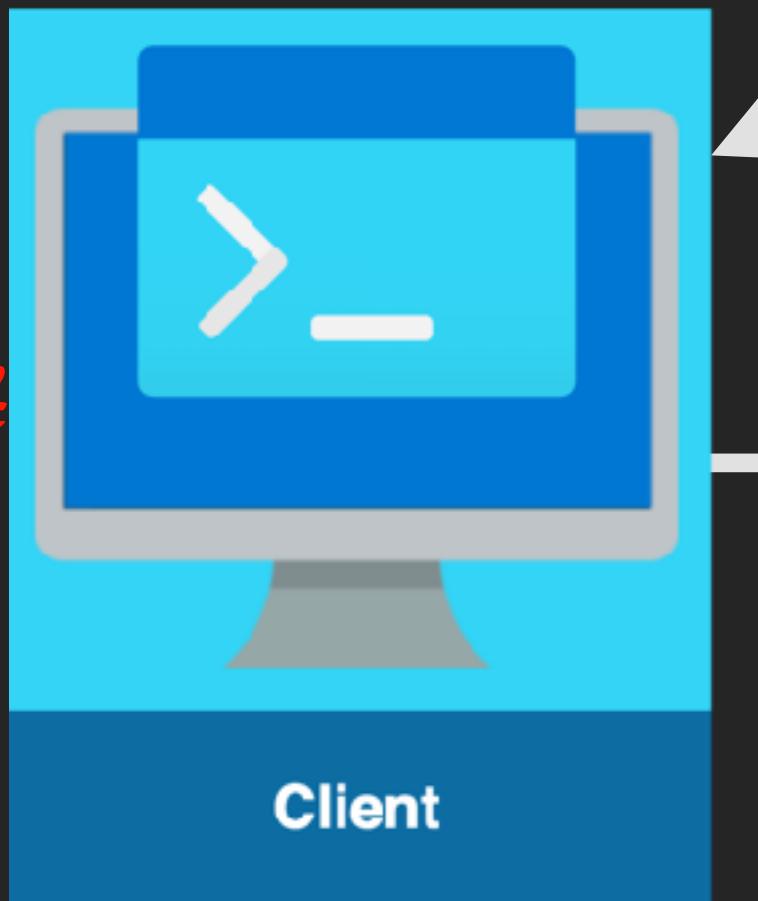


Root Registry



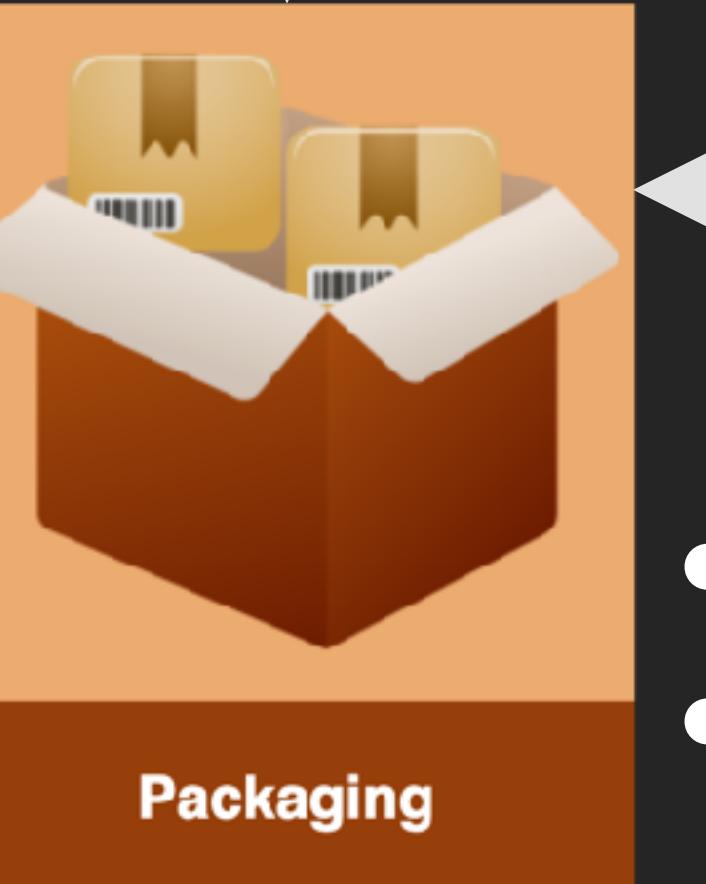
Stats & Reports

- Announce
- Tests
- Stats
- Comments
- Reviews



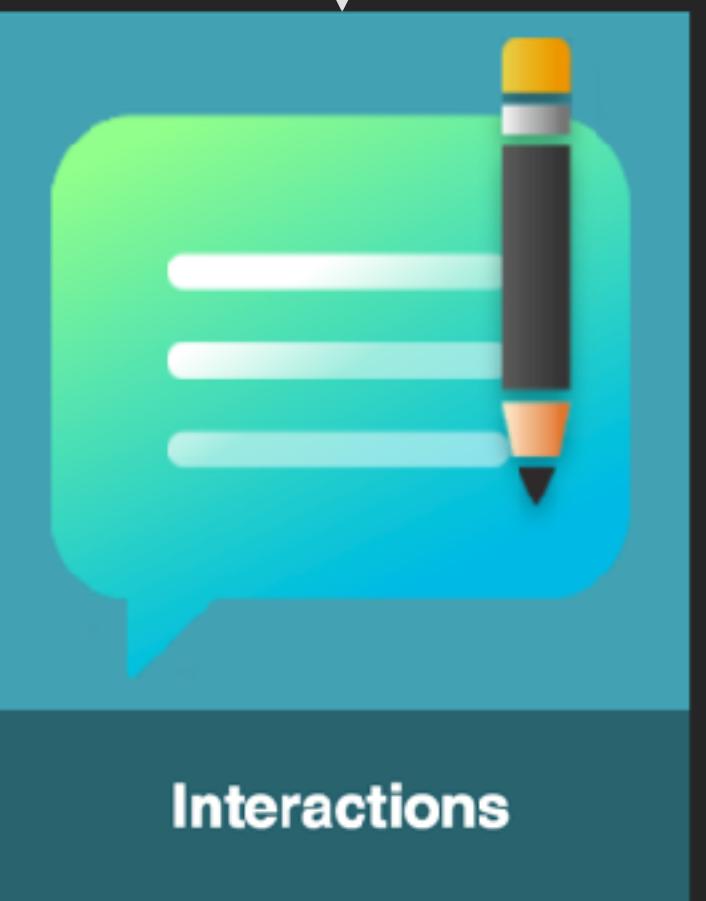
Client

- Compile
- Test



Packaging

- Package
- Release



Interactions

- Stream
- Notify
- Badges
- Reports

# JOIN US

**Background:** [https://wiki.postgresql.org/wiki/PGXN\\_v2](https://wiki.postgresql.org/wiki/PGXN_v2)

**Architecture:** [https://wiki.postgresql.org/wiki/PGXN\\_v2/Architecture](https://wiki.postgresql.org/wiki/PGXN_v2/Architecture)

**Project:** <https://github.com/orgs/pgxn/projects/1>

**Tasks:** <https://github.com/pgxn/planning/issues>

**Writing:** <https://justatheory.com/tags/pgxn/>

**Slack:** <https://pgtreats.info/slack-invite>

# **THANK YOU**

## **THE FUTURE OF THE POSTGRES EXTENSION ECOSYSTEM**

**DAVID E. WHEELER  
PGXN, TEMBO**