

PL/Parrot

Embedding the Parrot Virtual Machine in PostgreSQL

Jonathan "Duke" Leto and David Fetter

Parrot Virtual Machine

- Process (Application) Virtual Machine
- Register-based
- Continuation Passing Style
- Design Goals
 - Pluggable
 - Interoperable
 - Dynamic

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

Why Embed?

- PL's are (very) hard to write and maintain
- Framework for DSL's
- Platform independent, fast, stored procedures
- Allow various PL's to communicate
- Freeze/thaw subtransaction-level states

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

History

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

Features

- PL/PIR(U)
- Pass and return basic datatypes
- Basic security model (Don't do that)

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

Bugs

- Documentation
- SPI
- Triggers
- Parrot Bugs
 - IMCC Syntax Errors
 - Loading libraries from Embed API

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

[What is Parrot
Virtual Machine?](#)[Why Embed
Parrot VM in
PostgreSQL?](#)[History of
PL/Parrot](#)[Current Features](#)[Current
Bugs/Unimplemented
Features](#)[Example Code](#)[Future Goals](#)[How to Get](#)

Code

```
CREATE FUNCTION test_float_add(float) RETURNS float AS $$  
    .param num x  
    x += 5  
    .return(x)  
$$ LANGUAGE plparrot;
```

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

Future Goals

- PL/Rakudo - Perl 6 in your database!
- PL/Pynie - Python in your Parrot in your database!
- Tools to help create a new DSL a.k.a. PL on PL/Parrot

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

Get involved

- GitHub Issues:
- Try PL/Parrot on your system and submit detailed bug reports
- Fork on github and hack on stuff!

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

Thanks

- PL/Parrot team: Joshua Tolley, David E. Wheeler, Daniel Arbelo Arrocha + others
- Everyone working on Parrot VM and PostgreSQL

Embedding the Parrot Virtual Machine in Post- greSQL

Leto+Fetter

What is Parrot
Virtual Machine?

Why Embed
Parrot VM in
PostgreSQL?

History of
PL/Parrot

Current Features

Current
Bugs/Unimplemented
Features

Example Code

Future Goals

How to Get

Resources

- <http://parrot.org>
- @parrotvm / !parrot on twitter/identi.ca
- #plparrot on freenode