

Let's Postgres !

困った時はここを見よう！
役立つPostgreSQLの情報あれこれ

Let's Postgres 運営実行委員会

<http://lets.postgresql.jp/>



はじめに




そもそも、
『Let's Postgres』
ってなに？



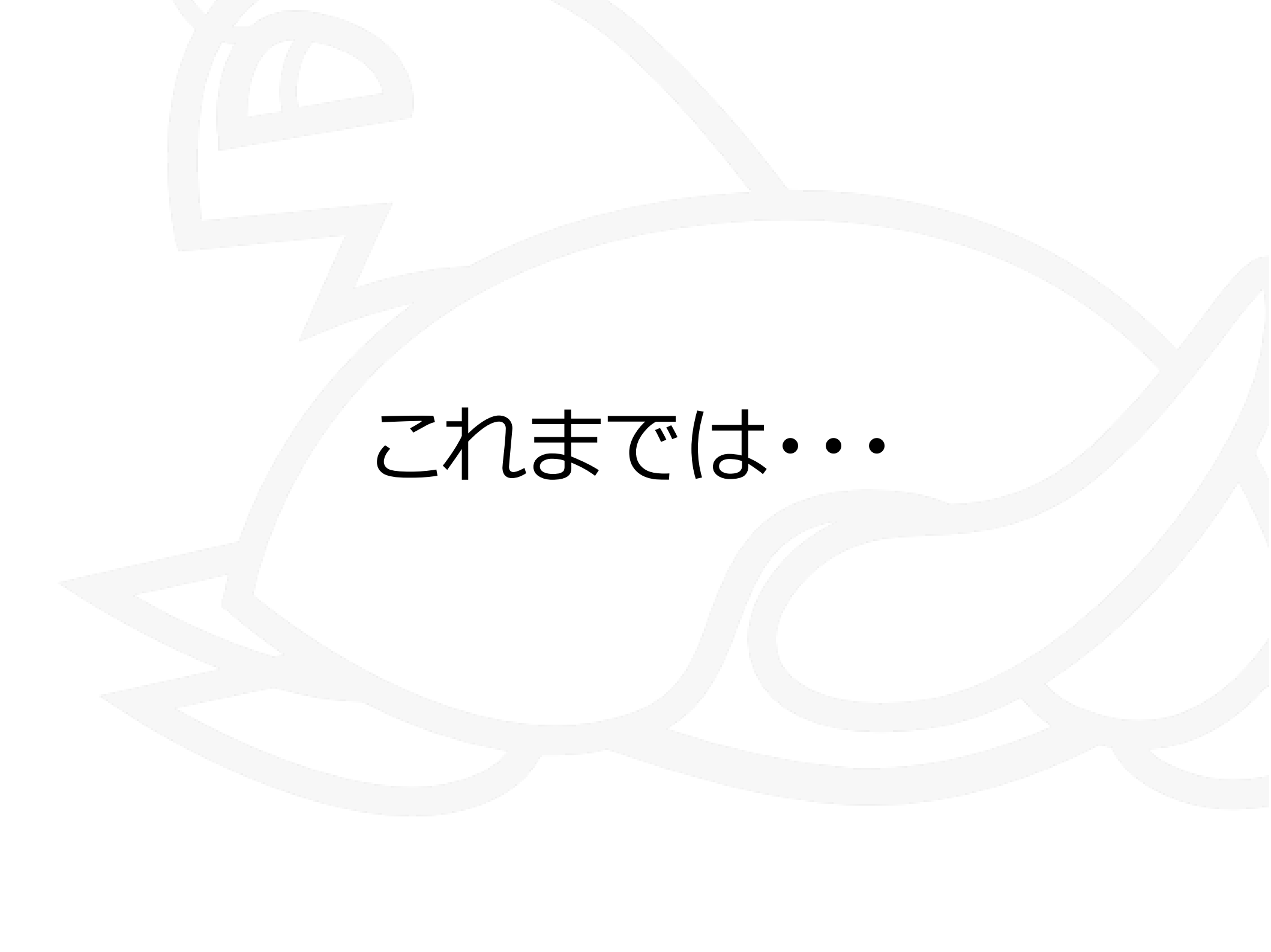
PostgreSQL情報 ポータルサイト by JPUG

です！



なぜ作られたかというと

(※ 2008 年 11 月からスタート)



これまででは...

「これってなんだろう？」



ぐぐる…



ブログ記事がヒット




読む…

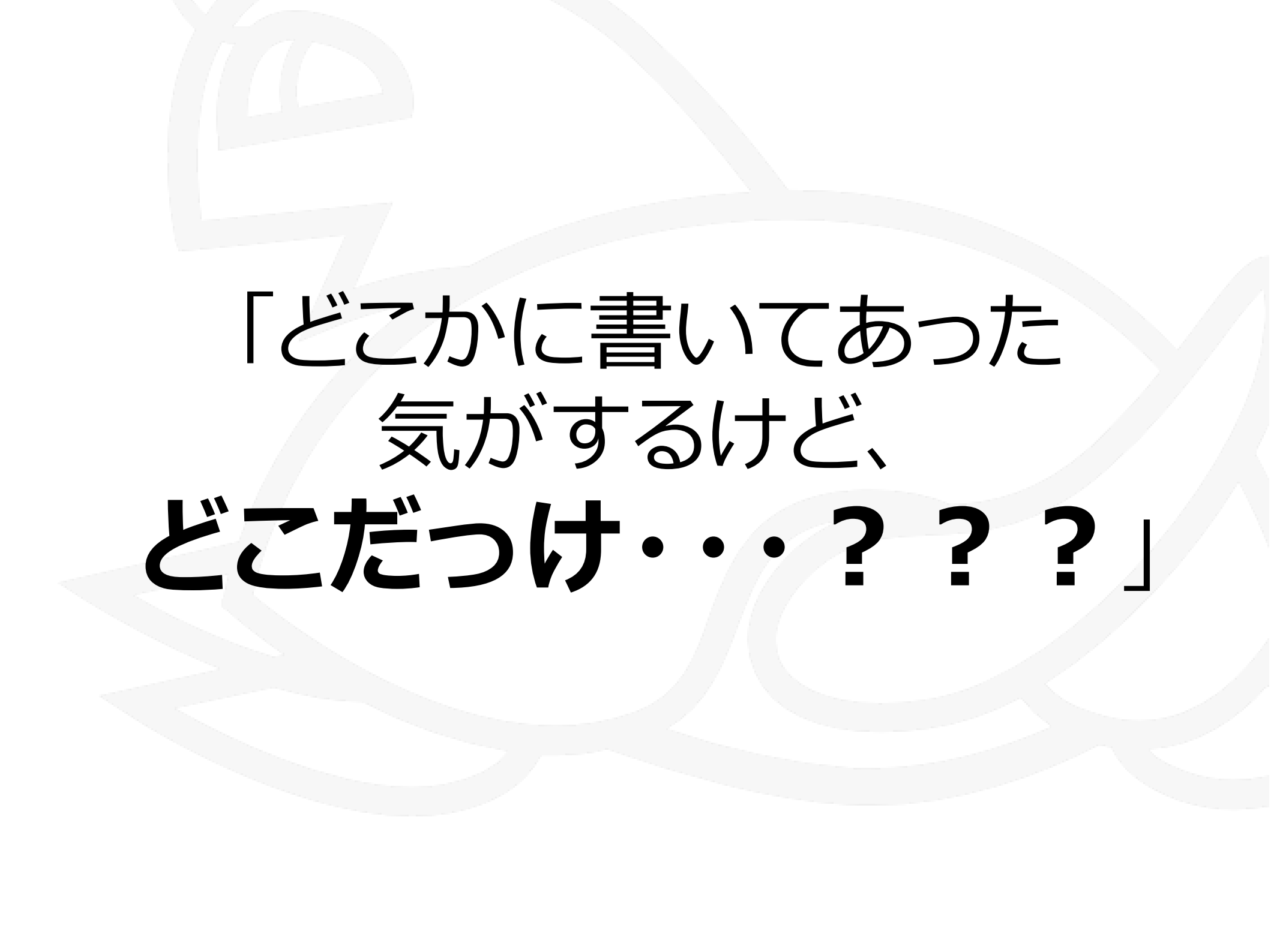


**「なんだ、この記事
失敗例で終わるかよ！」**


(結局よくわからないまま…)



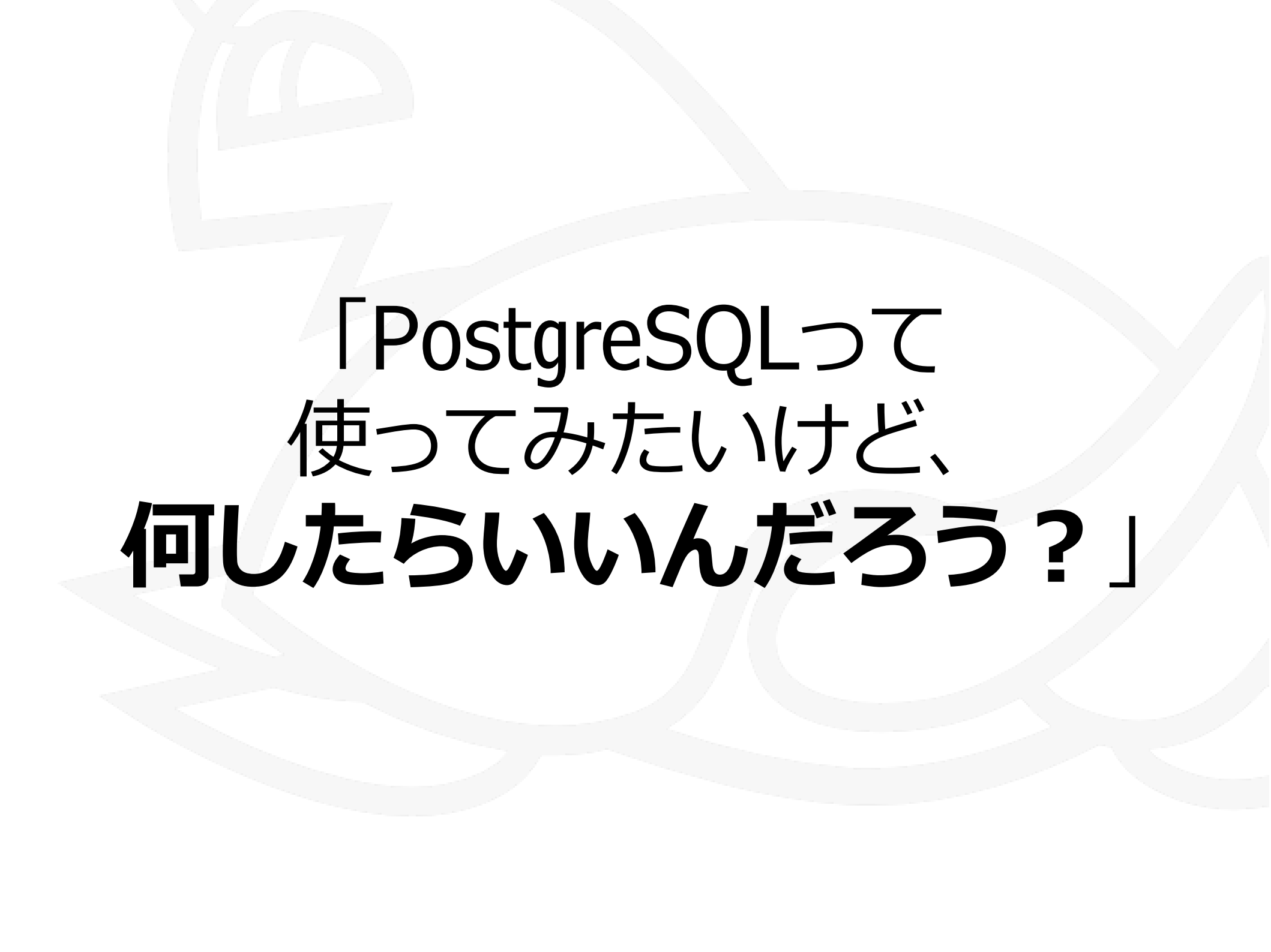
とかい



「どこかに書いてあった
気がするけど、
どこだっけ・・・????」



とかい



「PostgreSQLって
使ってみたいけど、
何したらいいんだろう？」」



などなど

- 貴重な情報が分散
- 全体像が見えない
- 優しくない (とくに入門者に)

という問題がありました・・・

そこで、

専用サイトを作って
情報を集約しよう！

ということに！



では、実際に見てみましょう

PostgreSQL のポータルサイト "Let's Postgres" へようこそ!

- [インストール](#)
- [運用管理](#)
- [チューニング](#)
- [トラブルシュート](#)
- [新機能解説: 9.0 / 8.4](#)
- [PostgreSQL入門](#)
- [サポートとFAQ](#)
- [事例紹介](#)
- [イベントレポート](#)
- [リンク集](#)
- [レプリケーション / クラスタ構成](#)
- [XML / PostGIS / 拡張モジュール](#)
- [パーティショニング / テキスト検索](#)
- [Window関数 / 再帰SQL](#)
- [文書: !\[\]\(38441ceaa711016e0bf2ad46ad394ff4_img.jpg\) 9.0 !\[\]\(6e027340d4263908f264926b1ad81c5e_img.jpg\) 8.4 !\[\]\(781510d64f329bf3c880acf086e884d6_img.jpg\) 8.3 !\[\]\(93cdf5b84f2bfec404f7441e84b6ba5c_img.jpg\) 8.2 !\[\]\(0f0f932ce3b5577a82f34ad23239a6e5_img.jpg\) 8.1](#)

pgpool-II + HS/SRクラスタ構成

2011年01月25日 公開

pgpool-II 3.0 と PostgreSQL 9.0 の Streaming Replication/Hot Standby 機能を組み合わせたクラスタ構成について解説します。

第9回 PostgreSQL Conference 2009 Japan イベント報告

2010年12月27日 公開

2009年11月20日、21日の両日、日本PostgreSQLユーザ会（以下JPUG）主催の初

■ ニュース

2011年 新年のご挨拶

2011年01月01日

PostgreSQL 9.0.0 がついにリリースされました！

2010年09月21日

Let's Postgres 公開編集会議

2010年02月03日

<http://lets.postgresql.jp/>



サイトの歩きかた

PostgreSQL のポータルサイト "Let's Postgres" へようこそ!

- [インストール](#)
- [運用管理](#)
- [チューニング](#)
- [トラブルシュート](#)
- [新機能解説: 9.0 / 8.4](#)
- [PostgreSQL入門](#)
- [サポートとFAQ](#)
- [事例紹介](#)
- [イベントレポート](#)
- [リンク集](#)
- [レプリケーション / クラスタ構成](#)
- [XML / PostGIS / 拡張モジュール](#)
- [パーティショニング / テキスト検索](#)
- [Window関数 / 再帰SQL](#)
- [文書: !\[\]\(0551a83d441798e532995956b603f604_img.jpg\)9.0 !\[\]\(54ee180c0037b66a36ce2219a481afde_img.jpg\)8.4 !\[\]\(73ae654e8897db9b21f1bf9d9efc07ef_img.jpg\)8.3 !\[\]\(278ecf8622de254ce2917d264729f4b0_img.jpg\)8.2 !\[\]\(3b5d74d5eba68301b1a5c22417b6b52c_img.jpg\)8.1](#)

目的別ガイド

pgpool-II + HS/SRクラスタ構成

2011年01月25日 公開

pgpool-II 3.0 と PostgreSQL 9.0 の Streaming Replication/Hot Standby 機能を組み合わせたクラスタ構成について解説します。

第9回 PostgreSQL Conference 2009 Japan イベント報告

2010年12月27日 公開

2009年11月20日、21日の両日、日本PostgreSQLユーザ会（以下JPUG）主催の初

■ ニュース

2011年 新年のご挨拶

2011年01月01日

PostgreSQL 9.0.0 がついにリリースされました！

2010年09月21日

Let's Postgres 公開編集会議

2010年02月03日

<http://lets.postgresql.jp/>

 **読み物**

▲ 一つ上に移動

 **記事一覧** **入門者向けの記事** 作成者 [柴田 淳](#) - 最終変更日時 2009年04月24日 07時28分

概要、導入、機能紹介、トラブルシュートなどの利用ガイドです。

 **技術的な記事** 作成者 [柴田 淳](#) - 最終変更日時 2009年04月24日 07時28分

機能、分析、性能チューニング、運用監視など技術的な話題を扱います。

 **事例紹介** 作成者 [板垣 貴裕](#) - 最終変更日時 2009年05月26日 01時52分

PostgreSQLの採用事例を紹介します。

 **イベントレポート** 作成者 [Nozomi Anzai](#) - 最終変更日時 2009年06月10日 11時36分

イベントやカンファレンスへの参加レポートです。

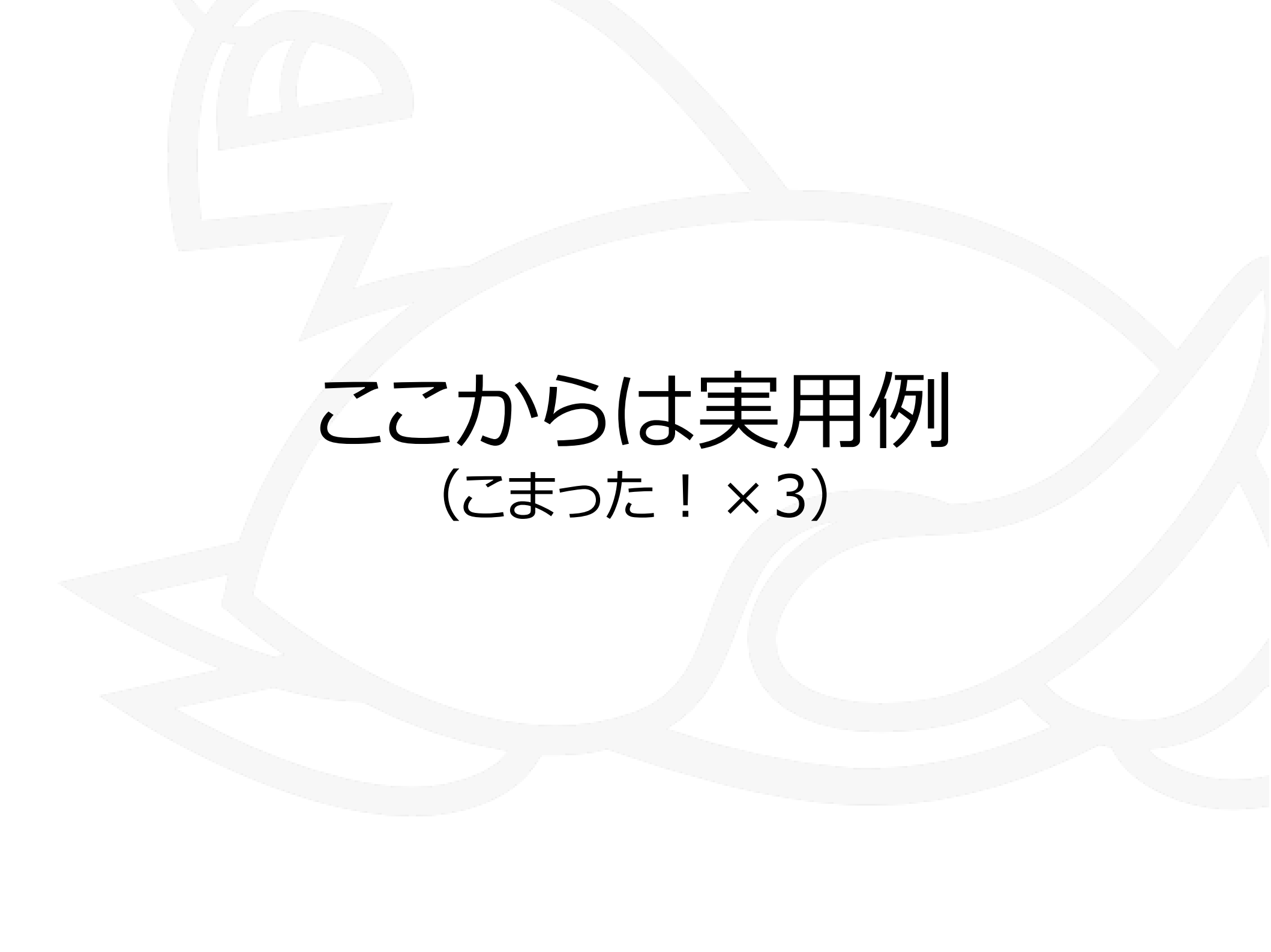
 **リンク集** 作成者 [板垣 貴裕](#) - 最終変更日時 2009年03月19日 20時23分

PostgreSQLに関連するサイトや記事へのリンクです。

 **インタビュー** 作成者 [稲葉 香理](#) - 最終変更日時 2010年11月30日 10時32分

PostgreSQLに関わる人たちのインタビューです。

<http://lets.postgresql.jp/documents/>

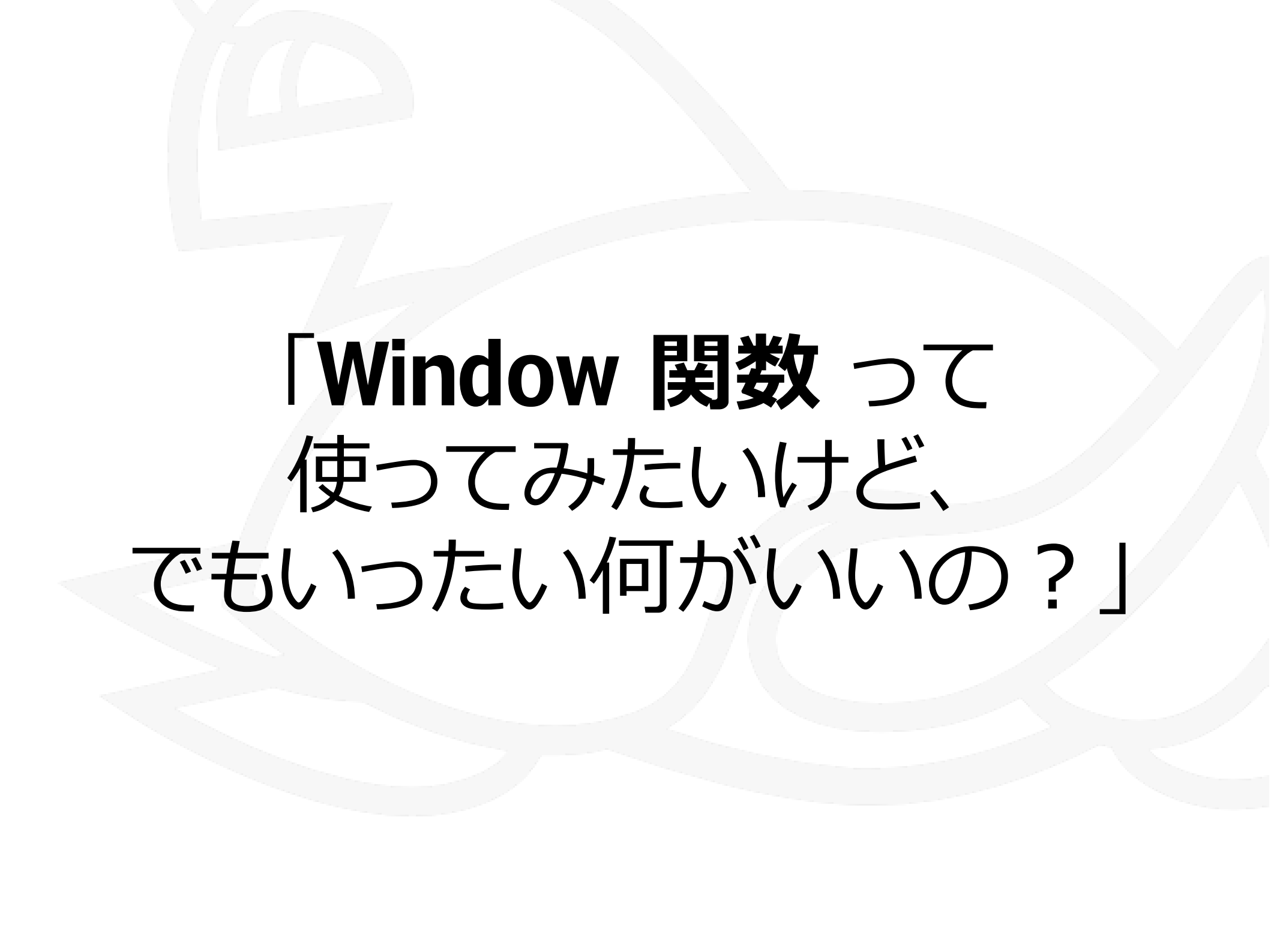


ここからは実用例
(こまった！ × 3)



こまった！

～ 開発編 ～



**「Window 関数 って
使ってみたいけど、
でもいったい何がしいいの？」**



**じゃあ、
Let's Postgresに
きいてみよう！**

Window関数

NTT オープンソースソフトウェアセンタ 板垣 貴裕

Window関数で使える集計関数と利用例を解説します。Window関数はOLAPなどで使われる複雑な集計クエリを効率よく処理するための構文です。Window関数はPostgreSQL 8.4で初めて導入され、9.0でさらに拡張されました。

解説

Window関数はテーブルを区間ごとに集計する機能です。集約関数 (GROUP BY) に似ていますが、Window関数では複数の行がまとめられることはなく、行それぞれが返却されます。また、処理中の行以外の行の値を読み取ることも可能です。

Window関数は以下の構文を使います。

- 関数(...) OVER (PARTITION BY ...) : 区間に分割
- 関数(...) OVER (ORDER BY ...) : 区間ごとに並び替え

関数には以下の表に挙げたものと、通常の集約関数 (countやsumなど) が使用できます。lag() や

http://lets.postgresql.jp/documents/technical/window_functions/

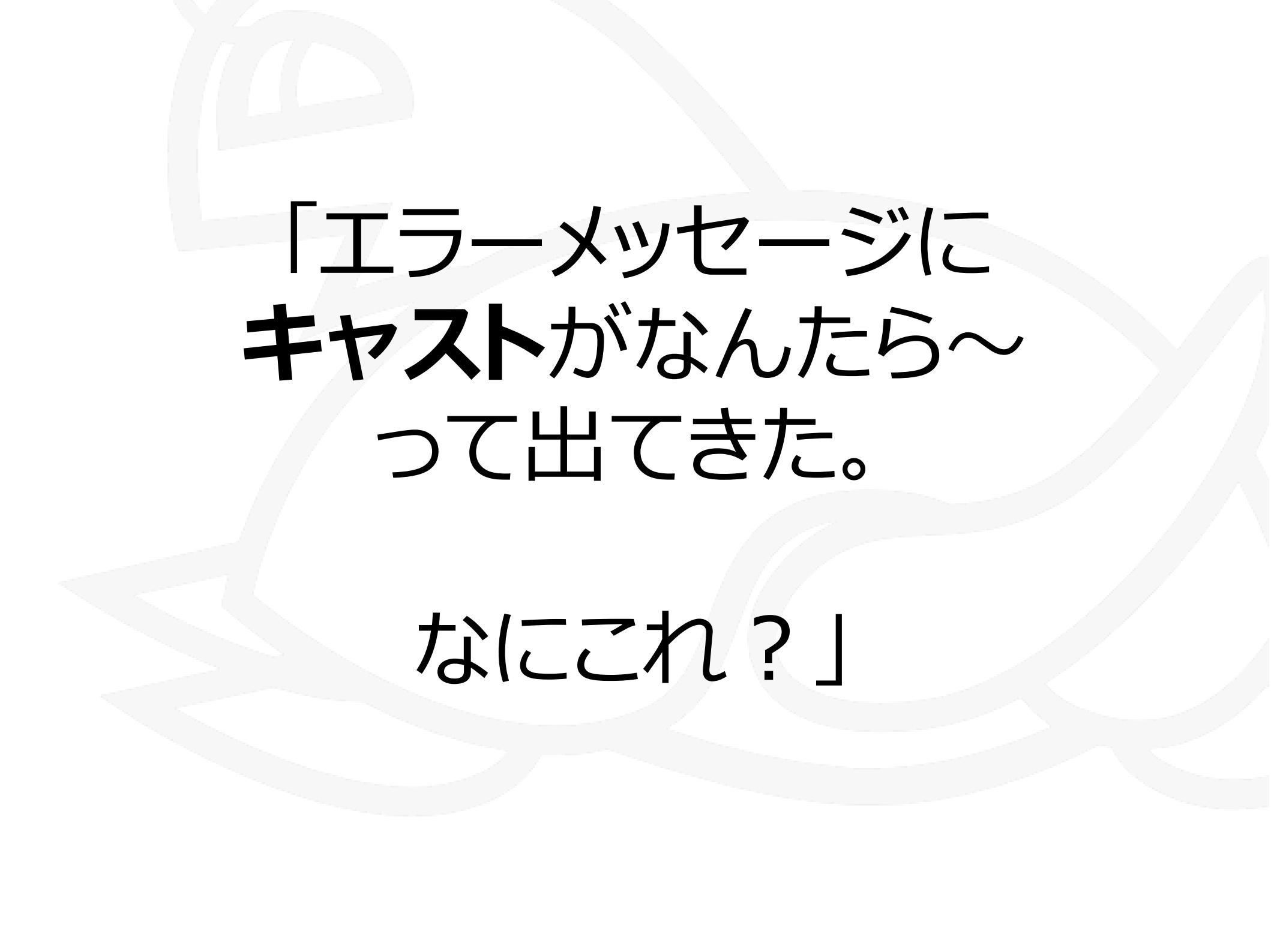
集計を効率よく処理できる構文

→ ソートして行番号つけたり、ランキングしたり

```
SELECT id, 地域, 身長,  
       rank() OVER (PARTITION BY 地域 ORDER BY 身長)  
FROM tbl;
```

→ 歯抜けIDの検索

```
SELECT id, prev_id  
FROM (SELECT id,  
          lag(id, 1, 0) OVER (ORDER BY id) AS prev_id  
      FROM tbl) AS t  
WHERE id <> prev_id + 1;
```



「エラーメッセージに
キャストがなんたら～
って出てきた。

なにこれ？」

ERROR: operator does not exist:
integer = text

HINT: No operator matches the given name
and argument type(s).
**You might need to add explicit
type casts.**

ERROR: function substr(...) does not exist

HINT: No function matches the given name
and argument types.
**You might need to add explicit
type casts.**



**じゃあ、
Let's Postgresに
きいてみよう！**

■ キャストを追加してバージョン互換性を保つには

NTT オープンソースソフトウェアセンタ 板垣 貴裕

PostgreSQL 8.3 では型変換のチェックが厳密になるよう仕様変更されました。特に文字列型への暗黙の型変換が無くなったため、以前のバージョンで動作していたアプリケーションであっても 8.3 上ではエラーが発生するかもしれません。もちろん、一番確実な対応方法は、型を厳密に扱うようアプリケーションやSQLを書き換えることですが、アプリケーションの規模が大きかったり、そもそも書き換えができないケースもあるのではないかと思います。できれば、せつかく 高速化された 8.3 へはバージョンアップしたいですし、互換性を保つ手軽な方法は何か無いでしょうか？

型エラーのメッセージ

型変換でエラーが発生している場合の典型的なエラーメッセージには、以下のようなものがあります。もし 8.3 へのバージョンアップ後にこれらのエラーが出力されるようならば、キャストの互換性が無くなったことが原因だと考えられます。HINT としても、「(SQLを書き換えて) キャストを追加する必要があるかもしれません」と提示されています。

<http://lets.postgresql.jp/documents/tutorial/cast/>

型変換のチェックが厳密に

→ PostgreSQL 8.3 での仕様変更

- 暗黙的な(勝手にやってくれる)型キャストがなくなった
- 特に、文字列型へのキャストをしなくなったことで、影響が出やすい

→ 解決方法

- ちゃんと型を指定する
- 「CREATE CAST」で暗黙キャストを登録してあげる



こまった！

～ 導入決定編 ～

DBMSの選択を任された！

「Oracle、MySQL、PostgreSQL…、
他にもデータベースって
いろんなのがあるけど、

一体どれを選んだらいいの？」



**じゃあ、
Let's Postgresに
きいてみよう！**

■ PostgreSQL 普及の課題と展望

NTT オープンソースソフトウェアセンタ 坂田 哲夫

第2回 PostgreSQL 普及の課題と展望 — 商用サポートの動向とコミュニティの活用

普及する上での課題

この記事の第1回では PostgreSQL の普及の現状について報告しました。まとめると、次のようになります。

1. すでに、30%以上の IT 企業で PostgreSQL を業務として利用しており、近いうちに半数近い企業が利用する見込みです。
2. 近年増えつつある、Oracle からのデータベースの移行先としては、OSS の DBMS では圧倒的に PostgreSQL が選ばれる傾向があり、MySQL が普及していると考えられる Web アプリケーションの分野でも MySQL と同程度には利用されていると見られます。
3. PostgreSQL が選ばれる理由は、その機能や性能が十分実用に耐えるレベルに達しているためである、と考えられます。

<http://lets.postgresql.jp/documents/tutorial/UserSurvey/>

■ PostgreSQLの機能と他のRDBMSの比較

SRA OSS, Inc. 日本支社 稲葉 香理

この記事は、gihyo.jp & Let's Postgres 連動企画「今こそ！PostgreSQL」の第2回記事です。第1回目「[🌍 PostgreSQLの今を知る](#)」は gihyo.jp で公開中です。

第2回 PostgreSQLの機能を他のデータベースと比較してみよう

PostgreSQLをはじめとしたオープンソースデータベースの機能はもはや商用データベースに匹敵するものです。以下に、PostgreSQL、商用データベースの代表格である Oracle Database、PostgreSQLと並んでオープンソースデータベースで著名な MySQL における簡単な機能比較を提示します。

表1 機能比較表

	PostgreSQL (v8.4)	Oracle (11g Enterprise Edition)	MySQL (Enterprise Server 5.1 Advanced / InnoDB)
--	----------------------	---------------------------------------	--

<http://lets.postgresql.jp/documents/tutorial/rdbms-hikaku/>

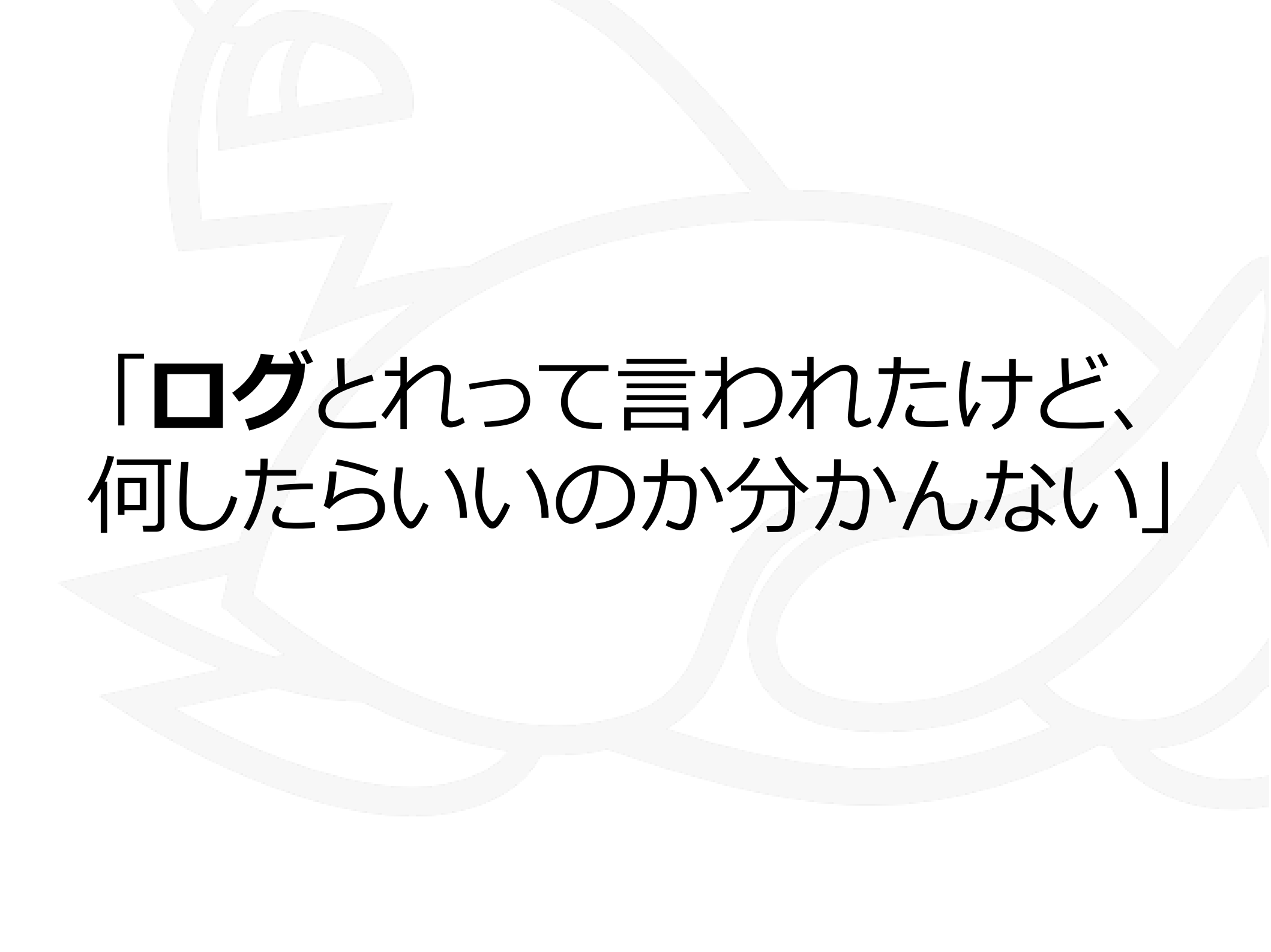
例えばこんなデータを掲載

- PostgreSQL と商用 RDBMS のメリット・デメリット
- 普及の度合い、利用形態のRDBMS間比較グラフ
- PostgreSQL を選んだ理由、選ばなかった理由
- 利用上の大きな課題、サポート



こまった！

～ 運用管理編 ～



「ログとれって言われたけど、
何したらいいのかわかんない」



**じゃあ、
Let's Postgresに
きいてみよう！**

■ ログ関連の設定

NTT オープンソースソフトウェアセンタ 笠原 辰仁

PostgreSQL でのおすすめできるログの設定をお知らせします。ここでいうログとは、PostgreSQL で発生したエラーや警告などの問題を記録するサーバログを指します。ログに関する設定は、PostgreSQLの運用を始める際に最も注意すべきパラメータ設定の一つです。ログは問題発生の有無や問題の原因を切り分ける際に、非常に重要な情報となります。適切なログの運用に向けて、下記を参考にしてみてください。

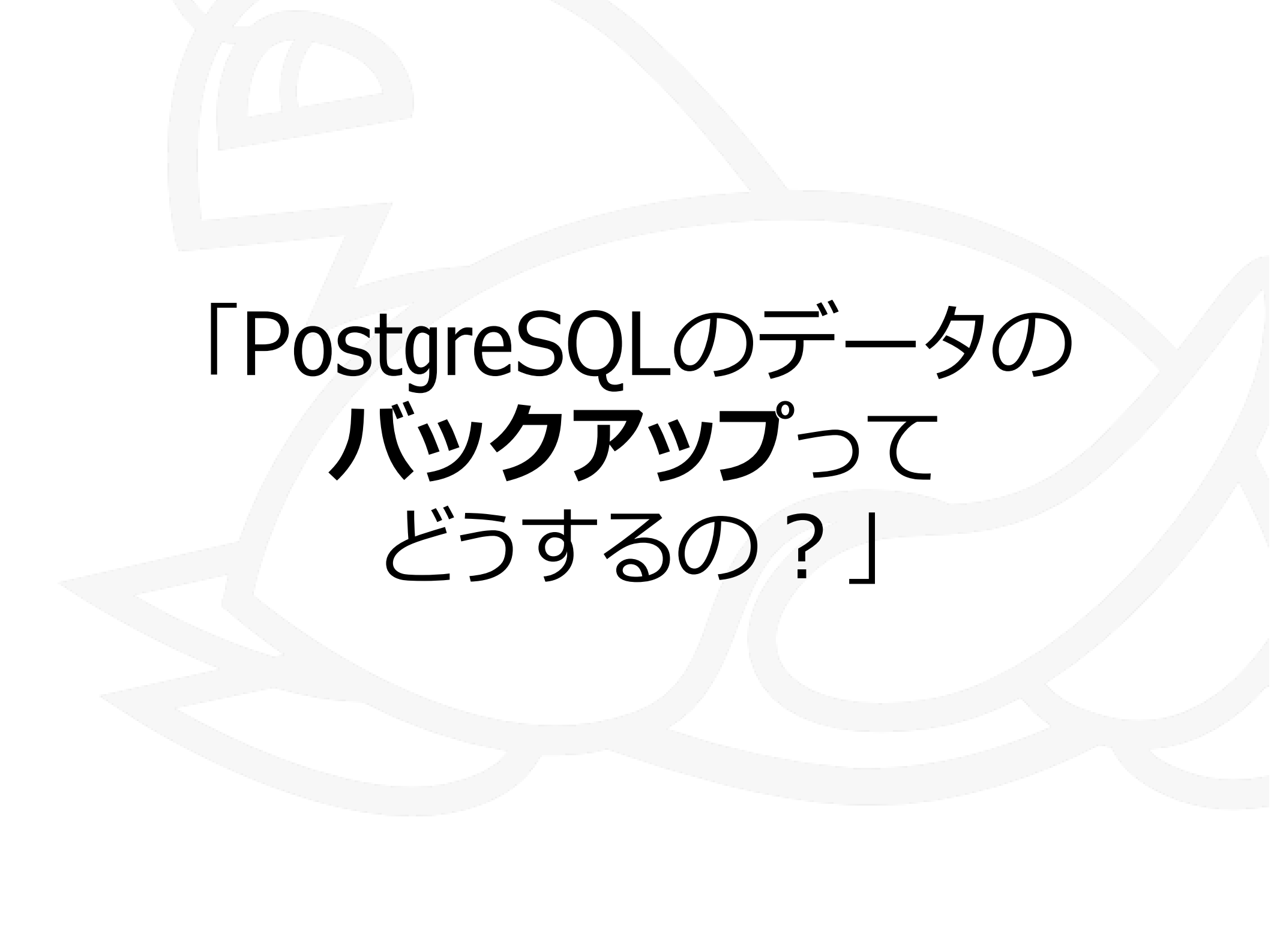
■ ログ運用関連のパラメータ設定

PostgreSQLのデフォルト設定（ソースコードからのインストール）では、ログは残さない設定になっています。インストールに使ったパッケージによっては、初めからログを残すよう設定ファイルがカスタマイズされていることもあります。まずは、確実にログを残すための設定を行いましょ

う。
ログの運用としては大きく3パターンほどあります。ログ運用は、ログ監視を行う運用監視ツールな

ど連携する必要があります。要件を確認していきましょう。
http://lets.postgresql.jp/documents/technical/log_setting/

- 【目的】 問題発生の有無を確認、問題発生の際を察知、問題があったときの原因切り分け
- 【方法】 (1) PostgreSQL独自のログ、(2) syslogを利用、(3) その他ツール
- postgresql.confでのパラメータ設定
 - 出力先指定、ログファイル名、日時・PIDなどの付加情報、SQL実行時間出力 etc...



「PostgreSQLのデータの
バックアップって
どうするの？」



**じゃあ、
Let's Postgresに
きいてみよう！**

■ バックアップの概要と方式一覧

NTT オープンソースソフトウェアセンタ 鈴木 幸市

本記事では、PostgreSQLデータベースのバックアップとリストア方法を解説していきます。今回は、各種バックアップの概要を説明し、次回以降でそれぞれの方法の具体的な手順を示していきます。

用語の整理

「バックアップ」とは、データベースの障害に備えたり、アップグレードの際にデータを移行するために、データベースの内容を別のファイルに取り出すことをいいます。バックアップには、データをSQLやテキストデータとして取り出す「論理バックアップ」と、データファイルをそのまま保存する「物理バックアップ」があります。

論理バックアップではSQLの結果としてデータを保存するので、もちろんPostgreSQLサーバは動作したままです。一方、物理バックアップでは、サーバを停止した状態でバックアップを行う「オフライン・バックアップ」と、サーバを起動したまま行う「オンライン・バックアップ」があります。

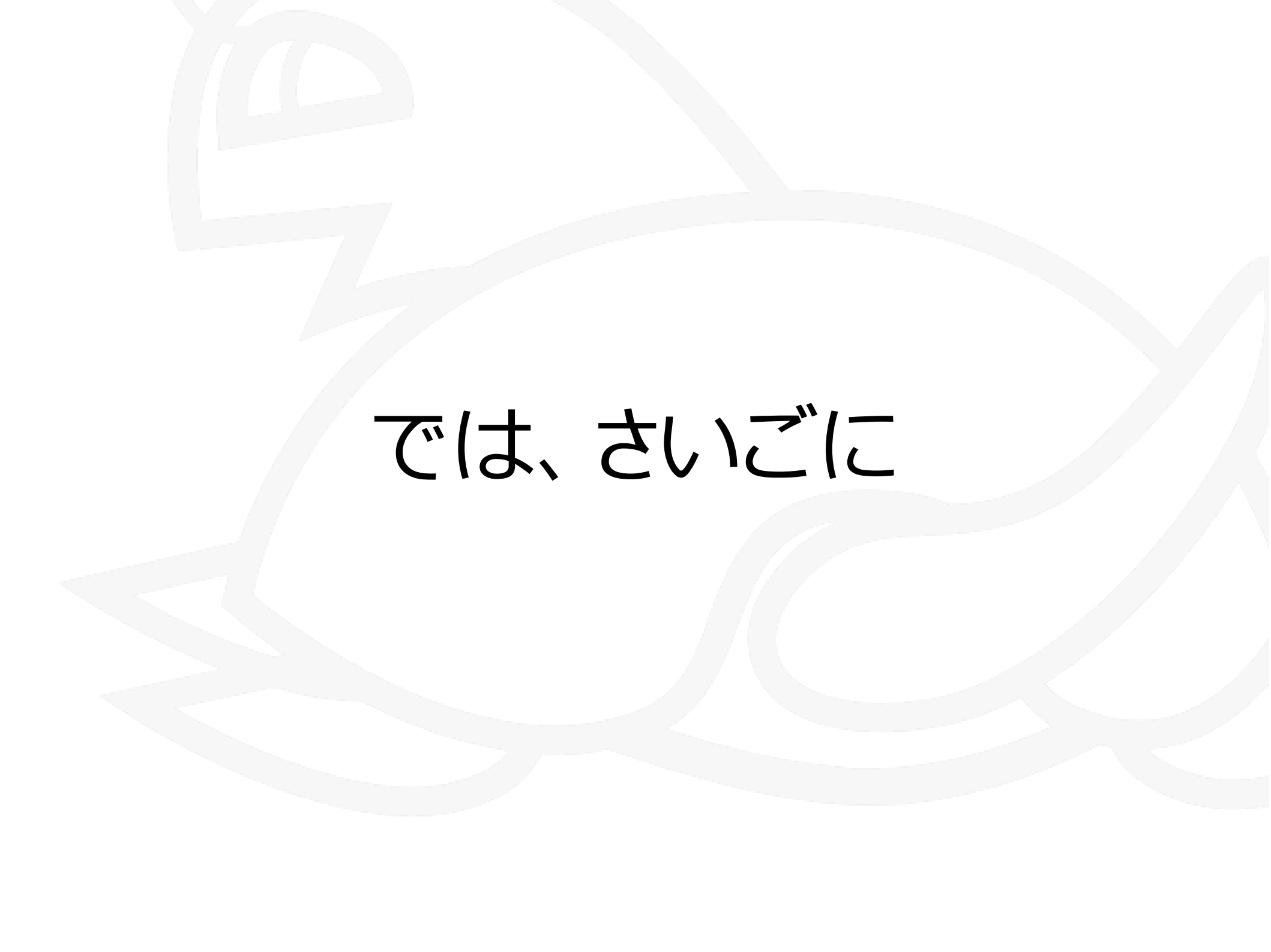
<http://lets.postgresql.jp/documents/technical/backup/>

いろいろな種類のバックアップ

名称	バックアップ方法	リストア方法	復旧時点	バックアップ所要時間
クラッシュ・リカバリ		再起動時に自動的に 行なわれる	サーバ停止状態で ファイルをコピー後、 最新状態または任意 時刻までリカバリを 行なう	
論理バックアップ	サーバを起動した まま、SQLとして保 存 (pg_dump, pg_dumpall)	SQLを再投入 (pg_restore, psql)	バックアップ開始時 点	長い
オフライン・バックアップ	サーバを停止し、 ファイルをコピーす る	サーバ停止状態で ファイルをコピーす る	バックアップ開始時 点	短い
オンライン・バックアップ	サーバを起動した まま、ファイルをコ ピーする。WALを アーカイブする	サーバ停止状態で ファイルをコピー後、 最新状態または任 意時刻までリカバリ を行なう	バックアップ終了直 後から最新状態の間 の任意の時点	短い



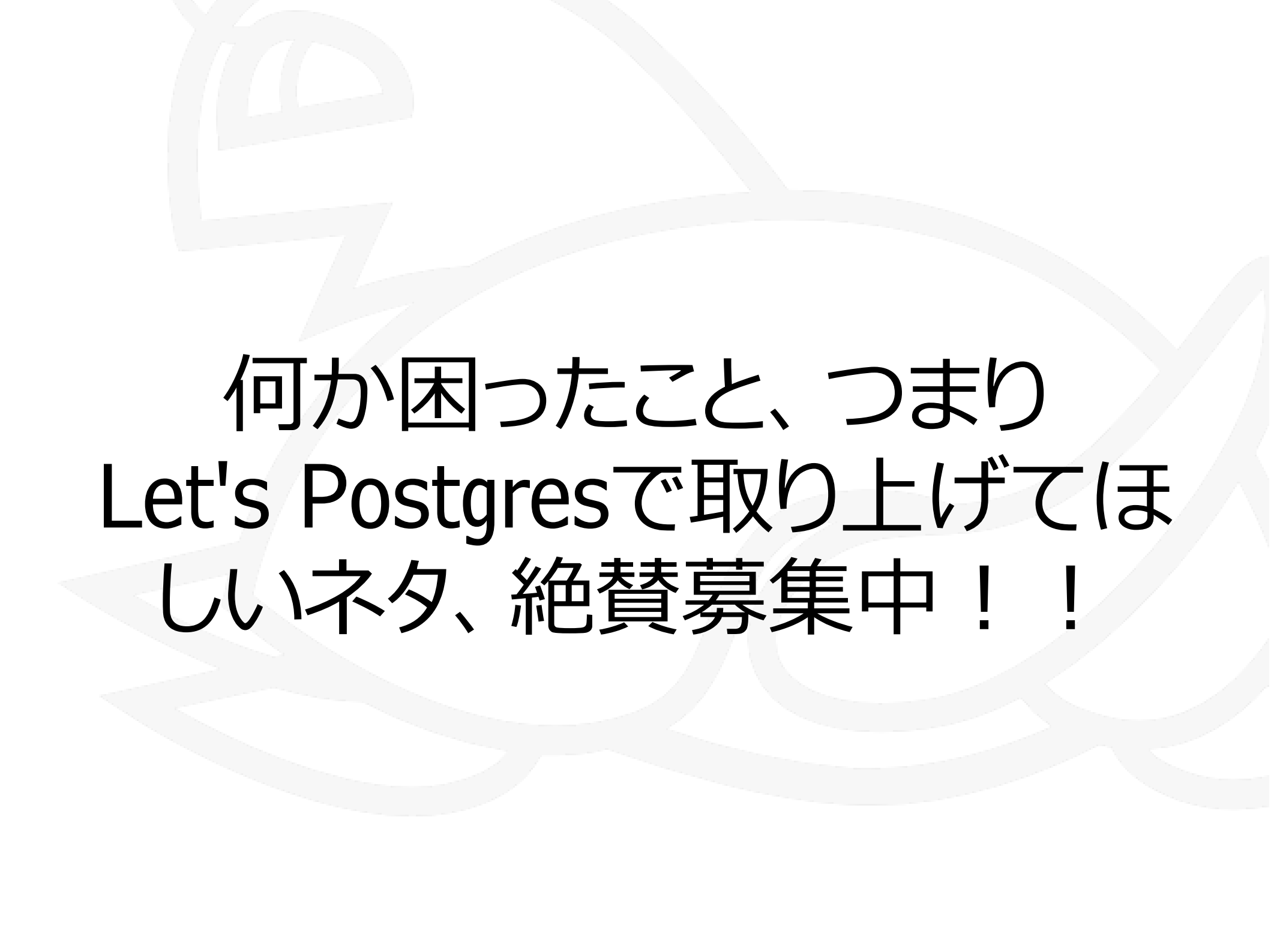
(ここまで)



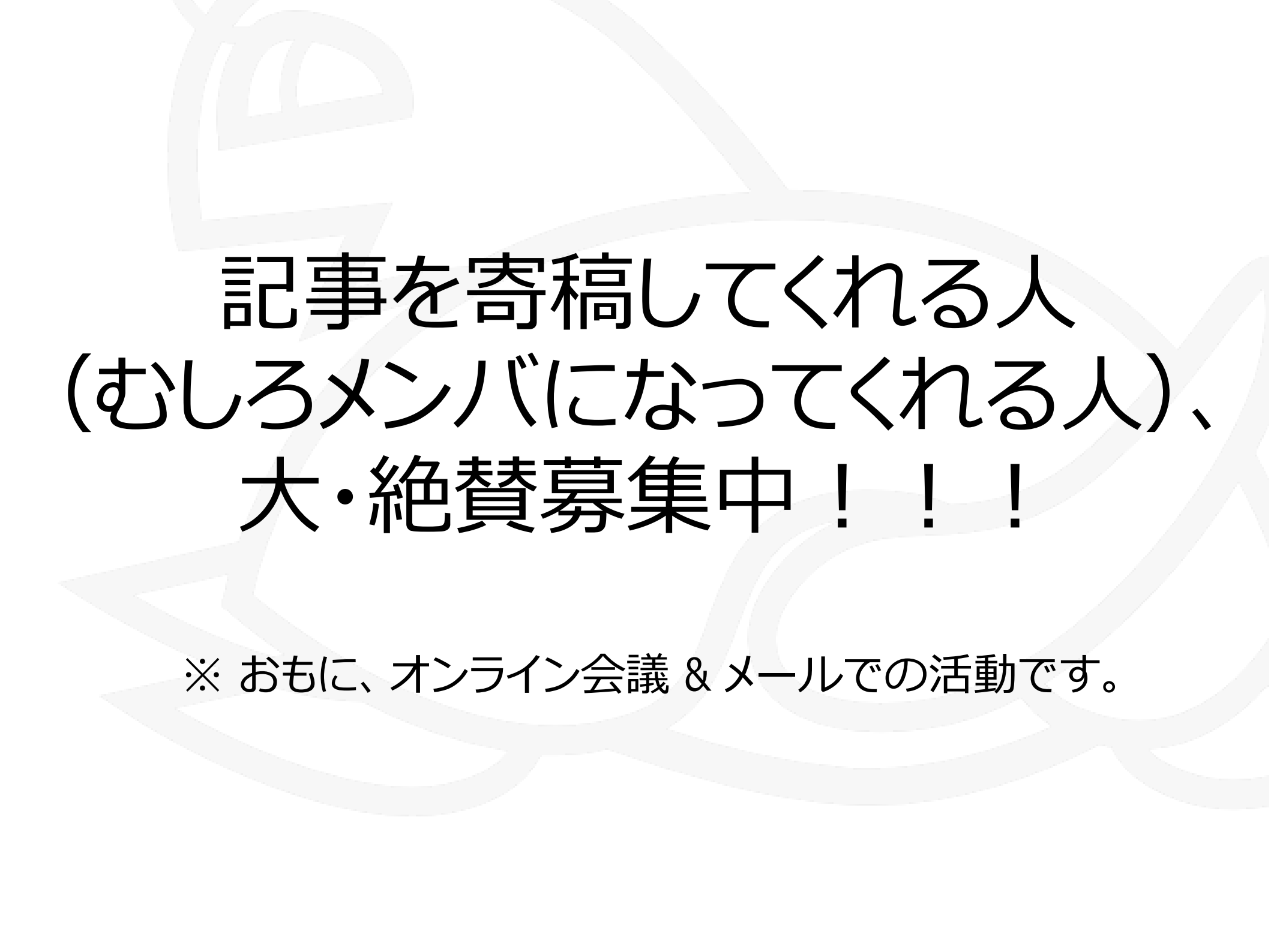
では、さいごに



**Let's Postgresでは
募集中です！**



何か困ったこと、つまり
Let's Postgresで取り上げてほ
しいネタ、絶賛募集中！！



記事を寄稿してくれる人
(むしろメンバになってくれる人)、
大・絶賛募集中！！！！

※ おもに、オンライン会議 & メールでの活動です。

おわり

連絡先: 【メール】 portal@ml.postgresql.jp
【Twitter】 [@lets_pg](https://twitter.com/lets_pg)

※ ときどき公開編集会議をやっています。