



PGECons
PostgreSQL Enterprise Consortium

2021年度CR部会活動報告

PostgreSQL エンタープライズ・コンソーシアム
CR部会
2022年5月20日

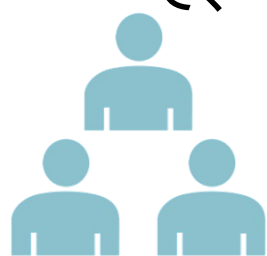
アジェンダ

- CR部会のご紹介
 - CR部会の目的、参加メンバ
- 2021年度の取り組みのご紹介
 - コミュニティへのフィードバックに向けた取り組み
 - PostgreSQL機能改善への協力
- おわりに
 - 2022年度にむけて

CR部会のご紹介

CR(Community Relations)部会の目的

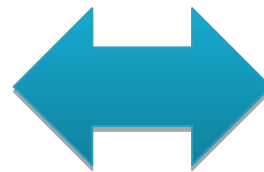
- PostgreSQL開発コミュニティへのフィードバック
 - エンタープライズ領域へのPostgreSQLの適用に向けて、開発コミュニティに技術的課題をフィードバック



企業さま



CR部会



開発コミュニティ

抱えている課題が解決される！

ユーザーの声が届く！

CR部会 2021年度参加メンバ

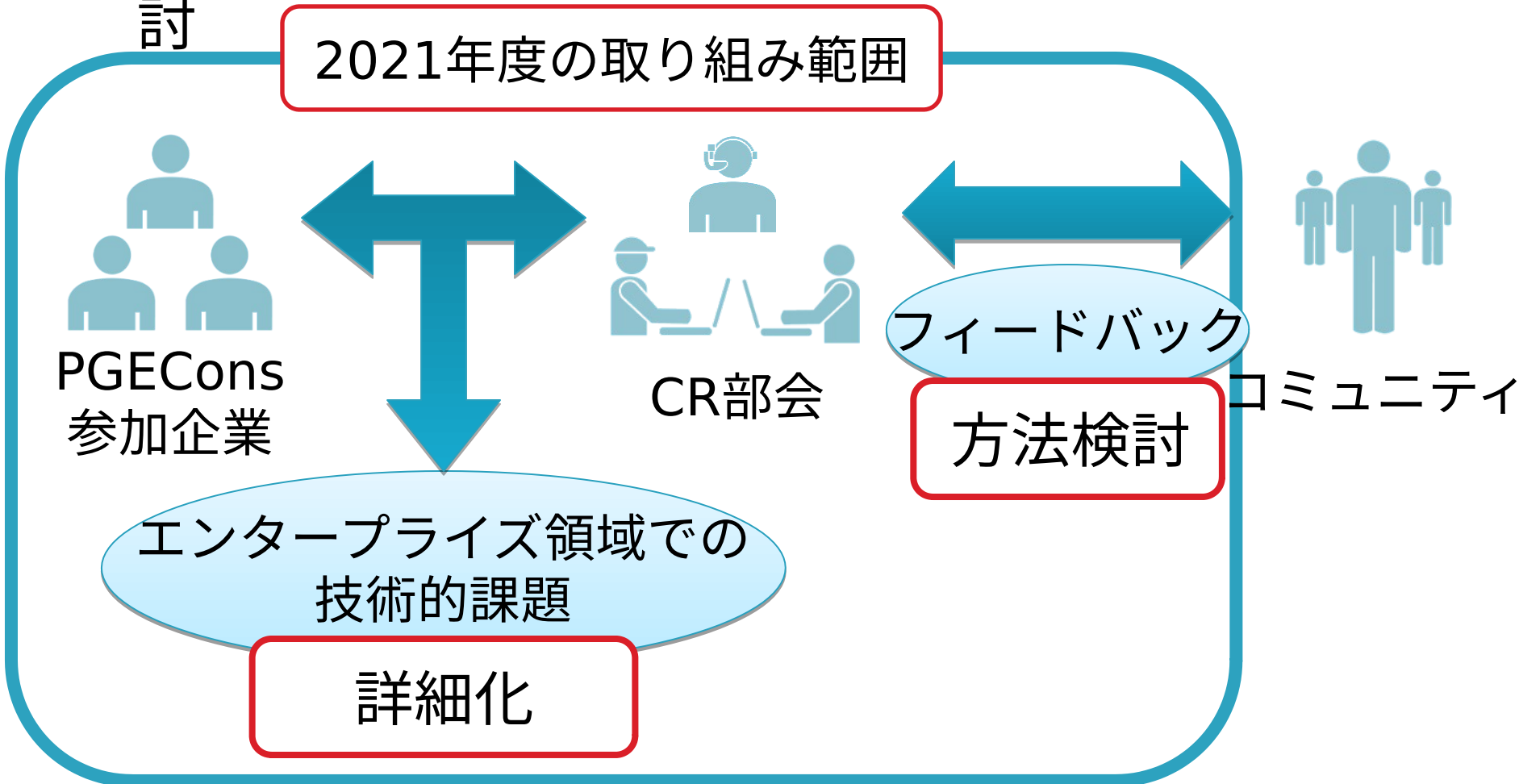
- SRA OSS, Inc.日本支社 【主査】
- NECソリューションイノベータ株式会社
- 日本電信電話株式会社
- 富士通株式会社

(企業名50音順・敬略称)

2021年度の取り組み

フィードバックに向けた2021年度の活動概要

- 技術的課題の詳細化とフィードバック方法の検討



2021年度の取り組み内容

1. チューニングノウハウの不足への対応
2. 負荷分散方式の現状調査とヒアリング実施
3. IVM (Incremental View Maintenance、マテリアライズドビューの増分リフレッシュ)開発の支援活動

2021年度の取り組み内容

1. 開発コミュニティへのフィードバックを目指し、エンタープライズ領域における技術的課題を詳細化 - チューニングノウハウの不足への対応
2. 負荷分散方式の現状調査とヒアリング実施
3. IVM (Incremental View Maintenance、マテリアライズドビューの増分リフレッシュ)開発の支援活動

チューニングノウハウの不足への対応

- 現状、データベースチューニングのノウハウが一箇所にまとまっておらず、共有されていない
 - PostgreSQLの公式マニュアルには「性能に関するヒント」という章があるが、記述が不足
 - この章の記述を充実させることを目指す
- 調査した項目
 - PostgreSQLのWikiなど、インターネット上のリソース
 - カテゴリ: GUCパラメタ、OS、VM、SQL、実行計画、ディスク構成、テーブル構成、ハードウェアなど

現状のマニュアル

14 性能に関するヒント

14.1 EXPLAINの利用

14.1.1 EXPLAINの基本

14.1.2 EXPLAIN ANALYZE

14.1.3 警告

14.2 プランナで使用される統計情報

14.2.1 単一系列統計情報

14.2.2 拡張統計情報

14.3 明示的なJOIN句でプランナを制御する

14.4 データベースへのデータ投入

14.4.1 自動コミットをオフにする

14.4.2 COPYの使用

14.4.3 インデックスを削除する

14.4.4 外部キー制約の削除

14.4.5 maintenance_work_memを増やす

14.4.6 max_wal_sizeを増やす

14.4.7 WALアーカイブ処理とストリーミングレプリケーションの無効化

14.4.8 最後にANALYZEを実行

14.4.9 pg_dumpに関するいくつかの注意

14.5 永続性がない設定

追加する内容(案)

14.6 大規模データ利用時

14.6.1 パーティショニングの利用

パーティションに分割すること、パーティション単位でTRUNCATEをすることなどを記載

14.6.2 統計情報

ALTER TABLE ... SET STATISTICSで統計収集レベルを決定することを記載

14.6.3 その他

正確な数をカウントする必要がある場合はcount(*)ではなくreltuplesを使うことなどを記載

14.7 高負荷システム利用時

14.7.1 ハードウェアの選定

メモリやCPUなど高性能なものを使用することを記載

14.7.2 OSのチューニング

マウント時にnoatimeをつける、ライトバック処理のチューニング、省電力モードの無効化、hugepageなどを記載

14.7.3 ディスク構成

pg_walディレクトリを別ディスクにしたり、テーブルスペースを用いてIOを分散することを記載

14.7.4 頻繁に実行されるクエリ

PREPARE文や、ストアドプロシージャを使用する

14.8 サードパーティ製のソフトウェア

pg_hint_planやpg_dbms_statsの紹介

今後の計画

- PostgreSQL開発者メーリングリスト(pgsql-hackers)に提案を送付、議論を実施
- 合意できたらマニュアルの改訂パッチを送付
- 議論、レビューを経て正式採用

2021年度の取り組み内容

1. 開発コミュニティへのフィードバックを目指し、エンタープライズ領域における技術的課題を詳細化 - チューニングノウハウの不足への対応
2. **負荷分散方式の現状調査とヒアリング実施**
3. IVM (Incremental View Maintenance、マテリアライズドビューの増分リフレッシュ)開発の支援活動

負荷分散方式の現状調査とヒアリング実施

- 検索、更新を含めた負荷分散方式は手法が確立されていない
 - 現状利用できる負荷分散方式の調査
 - 調査の一環としてヒアリング
 - 結果を開発コミュニティにフィードバック
- 現在利用できる負荷分散方式の調査
 - 負荷分散方式の分類
 - 参照負荷分散(Streaming replication, Pgpool-II, Aurora)
 - 更新負荷分散(FDW+パーティションテーブル、Postgres-XL、Citus)

ヒアリングの実施

- 負荷分散方式の利用実態の調査を実施
 - 現在使用中、あるいは評価中の負荷分散方式
 - その機能の評価
 - 負荷分散に関してコミュニティにフィードバックしたいこと
 - PGEcons内の3社にヒアリング
- 結果
 - 現在使用中、あるいは評価中の負荷分散方式
 - Pgpool-II、EDB failover manager、Replication manager、Azure Database for PostgreSQL Hyperscale (Citus)、FDW
 - FDWを使った例では、1日30億レコードをFDWベースのシャーディングで処理する実例あり
 - ストリーミングレプリケーションを使用している場合にはリカバリ衝突に課題
 - 上記の負荷分散方式すべてで、障害時の解析や、対応のしやすさなどで要望あり

結果をコミュニティにフィードバック

- PostgreSQL本家のWikiにヒアリング結果を掲載
 - pgsql-hackersでWiki掲載を知らせ、意見、フィードバックを募集中

結果をコミュニティにフィードバック(Wiki)

log in

page discussion view source history

Want to edit, but don't see an edit button when logged in? [Click here.](#)

PGECons CR Scaleout 2021

This is the summary of a research of scale-out solutions at 2021 and an interview to a few IT companies in Japan. They were conducted by the community relation team of PostgreSQL Enterprise Consortium. (PGECons is a non profit organization comprised of companies in Japan to promote PostgreSQL (<https://www.pgecons.org>)).

Contents [hide]

- 1 Scale-out Solutions
 - 1.1 Read load-balancing
 - 1.2 Read/Write load-balancing
 - 1.2.1 FDW + partitioned table
 - 1.2.2 Postres-XL
 - 1.2.3 Citus
 - 1.2.4 Others
 - 1.2.4.1 Greenplum
 - 1.2.4.2 TimescaleDB
 - 1.3 Summary
- 2 Interview to IT Companies in Japan
 - 2.1 About Read load-balancing
 - 2.1.1 Streaming replication and Pgpool-II
 - 2.1.2 Hyperscale (Citus)
 - 2.1.3 JDBC driver
 - 2.2 About Read/Write load-balancing
 - 2.2.1 FDW-based sharding
 - 2.3 Other feedback from interviewee companies about scaling
 - 2.3.1 Multi-master replication
 - 2.3.2 Write scaling using Redis as a cache
 - 2.3.3 Auto-scaling
 - 2.3.4 Configuration Simplicity
 - 2.4 Summary

Scale-out Solutions

There are two types of scale-out solutions: one is for read load balancing, the other is for read/write load balancing. The read load balancing is based on replication and load

https://wiki.postgresql.org/wiki/PGECons_CR_Scaleout_2021
から抜粋

結果をコミュニティにフィードバック(メール)

Subject: Research and Interview on scale-out solutions in Japan
From: Yugo NAGATA <nagata@sraoss.co.jp>
To: pgsql-hackers@postgresql.org
Date: Tue, 1 Feb 2022 10:29:13 +0900

Hello,

I made a wiki page to describe a research on scale-out solutions and an interview on scale-out solutions to a few IT companies in Japan.

https://wiki.postgresql.org/wiki/PGECons_CR_Scaleout_2021

This research and interview were conducted by the community relation team of PostgreSQL Enterprise Consortium [1]. We want to share the results and hope that our feedback would be beneficial to both PostgreSQL users and developers.

[1] PGECons is a non profit organization comprised of companies in Japan to promote PostgreSQL (<https://www.pgecons.org>). SRA OSS, Inc. is one of the members.

Regards,
Yugo Nagata

--

Yugo NAGATA <nagata@sraoss.co.jp>

2021年度の取り組み内容

1. 開発コミュニティへのフィードバックを目指し、エンタープライズ領域における技術的課題を詳細化 - チューニングノウハウの不足への対応
2. 負荷分散方式の現状調査とヒアリング実施
3. **IVM (Incremental View Maintenance、マテリアライズドビューの増分リフレッシュ)開発の支援活動**

IVMの開発支援

■ IVMとは

- マテリアライズドビューをリアルタイムで高速に更新する機能
- PostgreSQL開発コミュニティに提案中
- CR部会メンバー企業は、レビューに参加するなどして本体組み込みのIVM開発を支援

■ 2021年度の活動状況

- 開発コミュニティに積極的に提案したが、残念ながらPostgreSQL 15 (2022年冬リリース予定)には入らず
- IVM開発コミュニティでは、拡張モジュールとしてのIVMをリリース
- CR部会内で拡張モジュールIVMの説明会を実施

拡張モジュールpg_ivm

- コミュニティに提案しているIVMの拡張モジュール版
 - 現状のPostgreSQLでもIVMを利用したいとの声があった
 - pg_ivmを使ってもらうことにより、IVMへの理解、フィードバックを期待
 - GitHubで公開中 (すでに100 stars以上を取得)
 - https://github.com/sraoss/pg_ivm

pg_ivmの使用例

ビューを定義

```
test=# SELECT create_immv('immv',  
    'SELECT a.aid, b.bid, a.abalance, b.bbalance  
    FROM pgbench_accounts a JOIN pgbench_branches b USING(bid)');
```

NOTICE: created index "immv_index" on immv "immv"

```
create_immv
```

```
-----
```

```
10000000
```

```
(1 row)
```

ベーステーブルを更新

```
test=# UPDATE pgbench_accounts SET abalance = 1234 WHERE aid = 1;
```

```
UPDATE 1
```

```
Time: 15.448 ms
```

更新結果がビューに反映されている

```
test=# SELECT * FROM immv WHERE aid = 1;
```

```
aid | bid | abalance | bbalance
```

```
-----+-----+-----+-----
```

```
1 | 1 | 1234 | 0
```

```
(1 row)
```

拡張モジュールpg_ivm

- 現在使用できるSELECT
 - 内部結合、外部結合
 - DISTINCT
 - 集約は未対応
- 今後の予定
 - 現在はPostgreSQL 14のみ対応だが、今後対応バージョンを増やしていきたい
 - 使用できるSELECT文を増やしていきたい
- 要望、コメントなどは以下で
 - GitHubのIssues、Pull request
 - 日本PostgreSQLユーザ会のpgsql-jpメーリングリスト(日本語)

おわりに

2022年度に向けて

- 現在の取り組みを継続
 - エンタープライズ領域での技術的課題の詳細化とフィードバックを継続
 - 実行計画を制御できないために発生する課題の収集
 - Incremental View Maintenanceの開発支援
- コミュニティへのフィードバック活動を活性化
 - チューニングノウハウをマニュアルに追加
 - 実行計画についての課題を投稿して解決へと導く
 - Incremental View Maintenanceの議論へのフィードバック

皆様へのお願い

- 今年度（2022年度）は、ヒント句や Incremental View Maintenanceなどの PostgreSQLの改善活動の支援に取り組む予定です
 - 是非皆様のご意見をお寄せください

皆さんからのフィードバックが
よりよいPostgreSQLを作る
原動力になります

- 一緒にフィードバック活動しませんか？



PGECons

PostgreSQL Enterprise Consortium