



**PGECons**  
PostgreSQL Enterprise Consortium

# 2019年度活動成果報告

PostgreSQLエンタープライズ・コンソーシアム  
WG1 (新技術検証WG)

---

# パラレルクエリ検証 成果紹介

# パラレルクエリ・JITコンパイルとは

## ■ PostgreSQLでのパラレルクエリとは

- 1つのクエリを複数のプロセスで分担して並列処理すること
  - パラレルクエリで効率的に処理可能と判断した場合のみ採用
- BIなどのOLAP用途で恩恵を受けやすい
- PostgreSQL 9.6で採用され、10以降で継続的に強化されている

## ■ JITコンパイルとは

- クエリの実行時にコンパイルを実行し、高速化を図る機能
- 実行計画のコストが、設定した閾値を超えた場合に採用
  - コンパイル時にオーバーヘッドが発生
  - ある程度以上の規模のクエリでないと逆効果
- CPUリソースを多く使用する大規模クエリに有効
- 高速化の対象は、式評価 (WHERE句、ターゲットリスト、集約 など)、  
タプル分離 (ディスク上のタプルをメモリ上の表現に変換)

# 検証の目的

- PostgreSQLをOLAP用途でより実用的に使用可能とするパラメータ設定を検証するため、2種類の検証を試行
  - 検証A: 複数クライアントからのパラレルクエリ性能  
パラレルクエリ関連のパラメータを、物理CPUコア数を考慮した並列度と、大きく超える並列度に変更した上で、複数クライアントからのパラレルクエリの処理性能を比較
  - 検証B: JITコンパイル併用時の処理時間の比較  
検証Aと同内容の検証を、JITコンパイルを併用して行い、処理性能を比較

# 検証方法

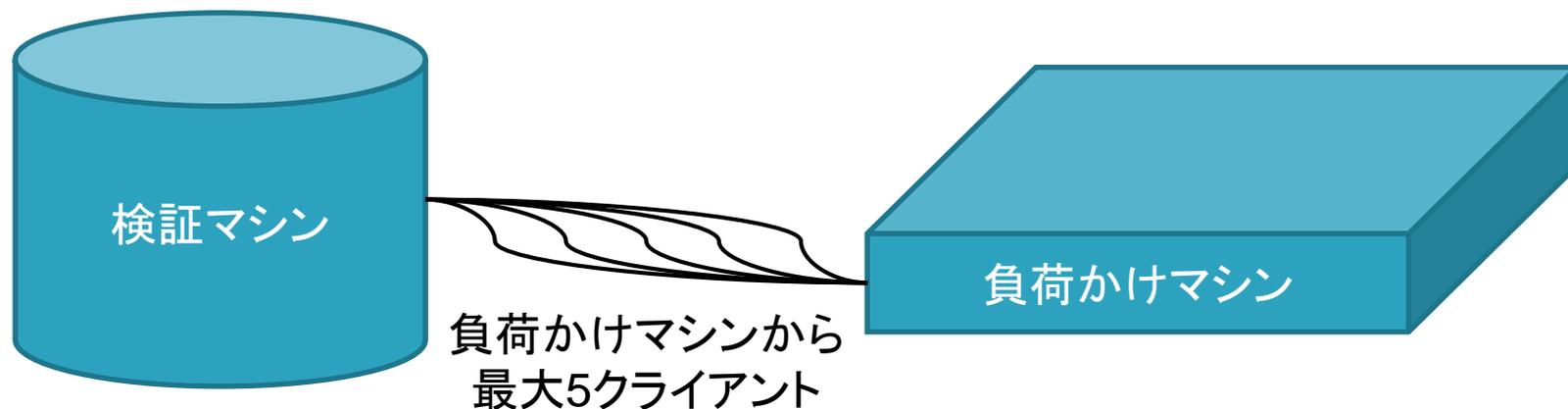
## ■ 検証方法

- 検証A・Bとともに、Star Schema Benchmark (SSB) に定義されているクエリ 4パターン / 13本のクエリを使用する
- 本検証のために作成したクライアントプログラムをクライアント上で実行し、DBサーバに対してクエリを実行する
- クエリ実行前にpg\_prewarmで全テーブルデータをメモリにロードする
- Scale Factorは 100 とする
  - Scale Factor=1で概ね1GBのデータサイズが生成される
  - テーブルに格納すると、60GB程度の物理データサイズとなる

# 検証環境

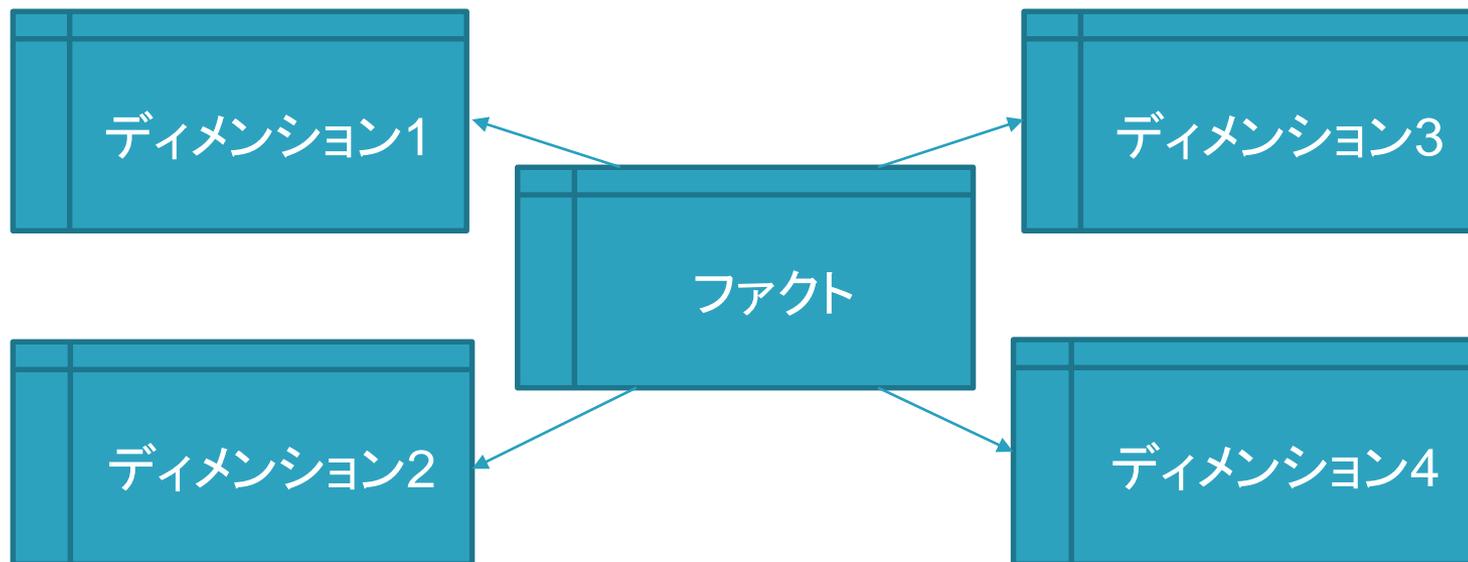
## ■ 検証環境

- Express5800/R120f-1E (Intel Xeon E5-2630 v4@2.20GHz)  
上記を2台 (測定用と負荷かけ用にそれぞれ使用)  
(日本電気株式会社 様 提供環境)
- 同一の負荷かけマシンから、最大5クライアントを起動して  
検証マシンに接続し、クエリを発行



# スタースキーマベンチマーク (SSB) とは？

- スタースキーマベンチマーク (SSB) のベースはTPC-H
  - ファクトテーブル: 分析対象となる実績値が格納
  - ディメンションテーブル: ビジネス上の軸になる詳細なマスターデータが格納



- テーブルサイズ
  - Scale Factor=100の時、全体で約60GB(テーブル物理サイズ)
    - 大部分がファクトテーブルに格納

# スタースキーマベンチマーク (SSB) とは？

- SSBでのクエリは4パターン、計13本が定義されている。
  - q1 (3本)  
ファクトテーブルとディメンションテーブル1つとの結合  
ファクトテーブルの選択率は0.008% (q1.3) ~ 1.95% (q1.1)
  - q2 (3本)  
ファクトテーブルとディメンションテーブル3つとの結合  
ファクトテーブルの選択率は0.02% (q2.3) ~ 0.8% (q2.1)  
集約、ソートあり
  - q3 (4本)  
ファクトテーブルとディメンションテーブル3つとの結合  
ファクトテーブルの選択率は0.0006% (q3.4) ~ 3.4% (q3.1)  
集約、ソートあり
  - q4 (3本)  
ファクトテーブルとディメンションテーブル4つとの結合  
ファクトテーブルの選択率は0.0009% (q4.3) ~ 1.6% (q4.1)  
集約、ソートあり

# 検証A 複数クライアントからのパラレルクエリ性能

- 複数クライアントからパラレルクエリを実行する場合、全クライアントにワーカが割り当たるようにした方が良い

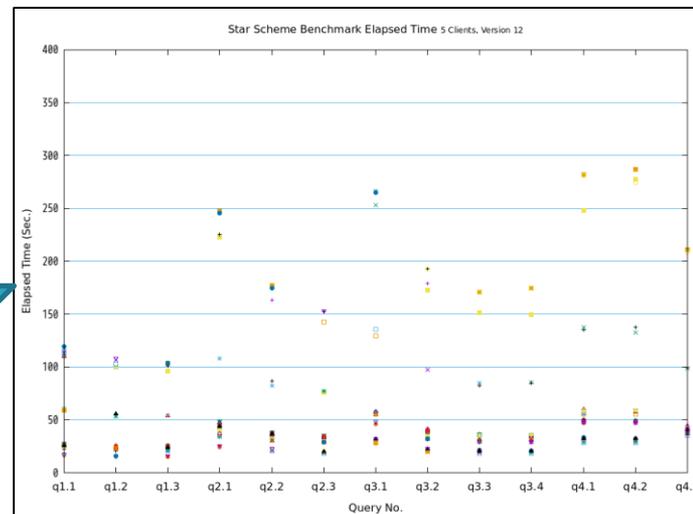
- `max_parallel_workers=9` (a) の状態と比較して、`max_parallel_workers=100` (b) の状態の方が、システム全体として高速

各クライアントにおけるベンチマーク1回の  
所要時間 平均値[秒]

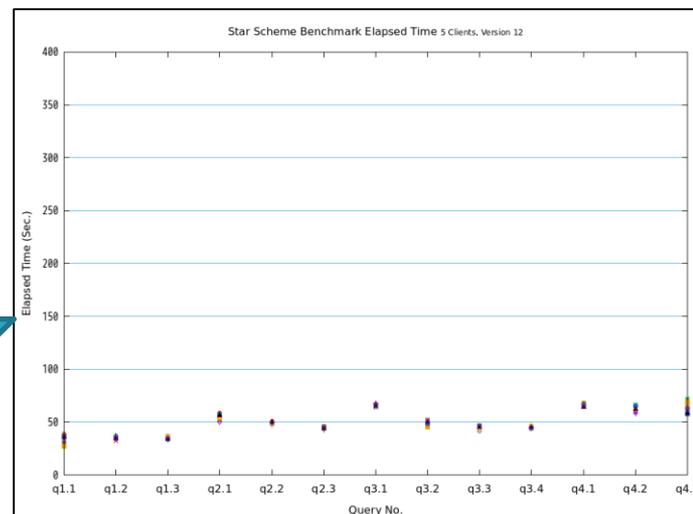
バージョン	<code>max_parallel_workers=9</code>	<code>max_parallel_workers=100</code>
v11	3627.5680	3165.4339
v12	3825.6987	3260.7030

- a. CPU物理コア数考慮 (グラフ右上)
- b. 全クライアントにパラレルワーカが割当てるよう考慮 (グラフ右下)

処理時間とワーカ数の関係 (v12, Worker=9)



処理時間とワーカ数の関係 (v12, Worker=100)

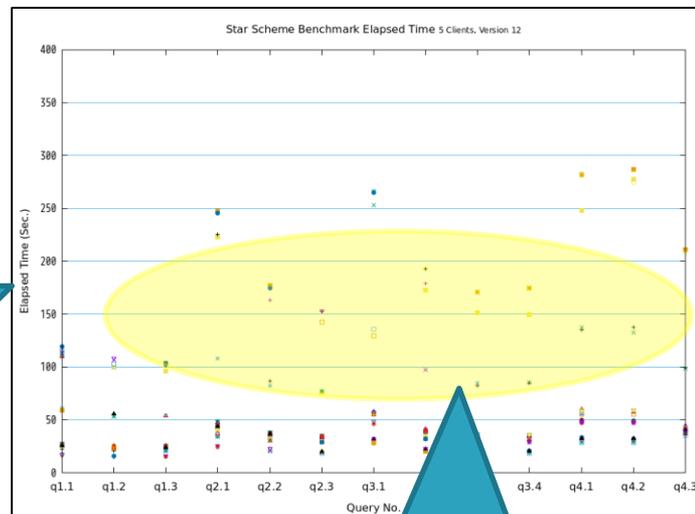


# 検証A 複数クライアントからのパラレルクエリ性能

- パラレルワーカーの要求総数が最大値を上回る場合、想定外のパフォーマンス悪化の可能性あり

- 要求数をすべて確保するか、まったく確保できないかの二択ではない
- ワーカーの割り当て順は先着順ではない
- 2つ以上のクライアントから、ほぼ同時に実行された場合、割り当てられるワーカー数が想定より少ない場合あり

処理時間とワーカー数の関係 (v12, Worker=9)



性能が安定しない

クエリ毎の平行ワーカー数  
(クライアント数 5の場合)

ワーカー数	0	1	2	3	4	5	6	7	8	9
クエリ数	91	51	0	0	314	92	0	0	33	69

↑ 予想外の確保ワーカー数 ↑

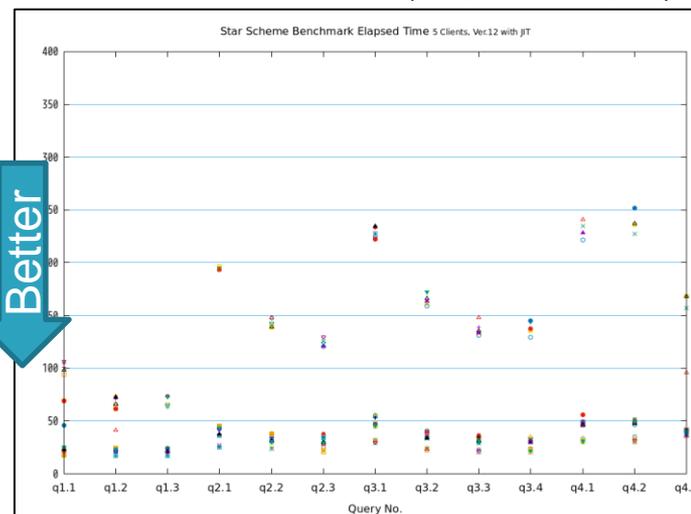
# 検証B JITコンパイル併用時の処理時間の比較

- max\_parallel\_workersの設定値の違いによる性能の傾向は検証Aと同様
- 物理コア数を超える平行ワーカーが起動する場合には、JITコンパイルを有効にした方が、性能は向上

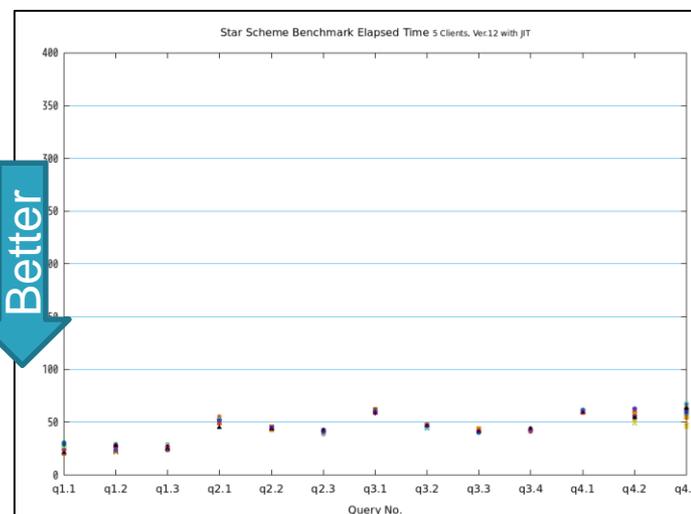
q2.1におけるクエリ所要時間 最大・最小値[秒]  
(5クライアント時, max\_parallel\_workers=100)

Ver.	最小値 (JIT OFF)	最小値 (JIT ON)	最大値 (JIT OFF)	最大値 (JIT ON)
v11	47.7680	47.6245	53.7806	50.2430
v12	49.4745	45.3404	58.4706	55.2112

処理時間とワーカー数の関係 (v12, Worker=9, JIT)



処理時間とワーカー数の関係 (v12, Worker=100, JIT)



# まとめ

- **複数クライアントからパラレルクエリを使用する場合は  
パラレルワーカー数の最大値の設定に注意が必要**
  - 複数クライアントがパラレルクエリを使用する場合は、  
全クライアントにパラレルワーカーが割り当たるように  
設定したほうが良い
  - パラレルワーカーの要求総数がパラレルワーカーの最大値を  
上回る場合、想定外のパフォーマンス悪化の可能性あり
    - この場合の、実際に割り当たるワーカー数は予測が困難なため
- **JITコンパイル併用時はパラレルワーカーの稼働数に注意**
  - パラレルワーカーの総数が物理CPUコア数を超える場合は  
効果が期待できる
  - 逆に物理CPUコア数以下に収まる場合は性能が劣化する可能性あり

---

# *Appendix*

# パラレルクエリ検証環境 (日本電気株式会社様提供)

## ■測定環境

型番	Express5800/R120f-1E
CPU	インテル Xeon E5-2630 v4@2.20GHz
メモリ	256GB
内蔵ストレージ	SSD 400GB x 8 (RAID 5構成)
OS	Red Hat Enterprise Linux 8.2

## ■負荷かけ環境

型番	Express5800/R120f-1E
CPU	インテル Xeon E5-2630 v4@2.20GHz
メモリ	256GB
内蔵ストレージ	HDD 600GB SAS x 24 (RAID 5構成)
OS	Red Hat Enterprise Linux 7.3



# PGECons

PostgreSQL Enterprise Consortium