PGCon 2013 参加レポート

22.Jun.2013.
日本PostgreSQLユーザ会・夏セミナー

NTT OSSセンタ
坂田　哲夫

目次

- PGConの開催概要
- 発表内容–傾向と対策
- 興味深い発表を数点、ご紹介
- コミュニティ動向—エンディングから

参考資料;こちらもお勧めします!

江川さん@NTTデータ 作

『10大ニュースで振り返るPGCon 2013』、

(第26回しくみ＋アプリケーション勉強会資料)、

http://www.slideshare.net/daichiegawa/jpug10pg-con2013

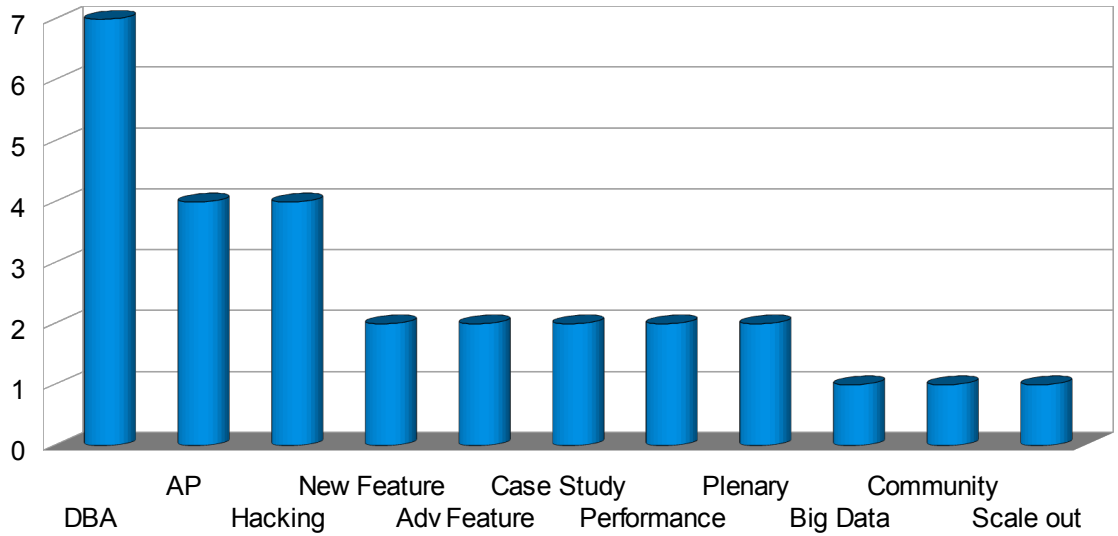
今年の開催の概要

- 場所：カナダ国・オタワ市
- 期間：2013年5月21日～25日
- 参加者：256（昨年は215）
- 行事一覧

	午前	午後	夜
21日	Tutorial	Tutorial	(なし)
	Cluster Summit	*XC-day*	
22日	Tutorial	Tutorial	Pick up
	Developer meeting		
23日	Talk	Talk	Reception
24日	Talk	Talk	Hacker Lounge
25日	Unconference		

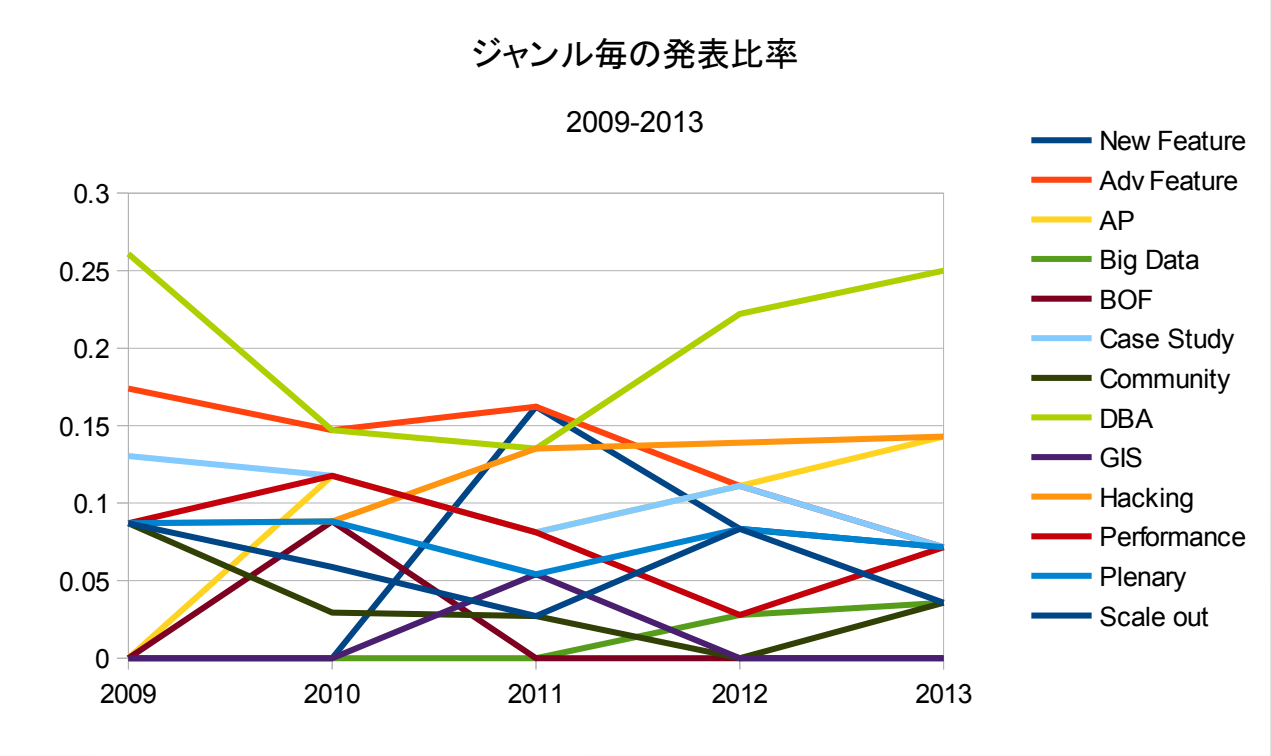
今年の発表(talk)の傾向

- 3パラレル・31件の発表
- ジャンル別の発表件数
  - DBA・AP・Hackingが多い (3者で半数)
  - 目だったテーマ
    - HA向け機能の紹介が2件
    - 開発者向けのSQL機能関連が3件



ここ数年の傾向

- ジャンル別の発表(talk)割合
  - 増えたもの：DBA(管理者)・Hacking
  - 減ったもの：Advanced Feature, Case study



傾向と対策 Acceptされるために

- 倍率は約3倍
  - 一人で数件出す人も
- Abstractでは内容をアピール
  - トレンドにあった話題
  - 包括的な視点
  - キャッチー(聞いてみたくなる)

注目の講演

- 発表資料が公開されているセッションを紹介
- 高可用システムについて二題（トレンド）
- SQL関連で三題（包括的）
- キャッチーの例

高可用サービス二題　その一

- Implementing High Availability
  - Dimitri Fontaine
- 可用性全般についての解説
  - 単なる故障時のfail overではなく、サービスを継続させるための全般的な注意事項
    - 単純なC/S構成から順を追って複雑なクラスタへ
  - 性能の安定性
    - 更新スケール、シャーディング
  - トランザクションの持続性
    - WALストリーミング

高可用サービス二題　その一(続)

- データ持続性(論理的な情報の保持)
  - バックアップ戦略の検討
    - 日・週・月・年などの単位
    - 論理バックアップ
  - バックアップと同様にリストアも重要
    - リストアの所要時間を含めて検討
- データ可用性(物理的なデータの維持)
  - データを消失しない
    - 物理バックアップ、PITRとWarm stand by

高可用サービス二題　その一(続)

- サービスの検討ー負荷の分離
  - オンラインサービスとバックオフィス(データの分析業務など)で分けよ
  - データは論理レプリケーション(pgqがお薦め)
- 待機サーバの設置ーストリーミングレプリケーション
- 更新スケール性ー増える負荷への対処
  - シャーディングでDBを(水平)分割

高可用サービス二題　そのニ

- PostgreSQL High Availability with Corosync/Pacemaker
  - Nikhil Sontakke
- Streaming Replication(SR)による高可用構成の紹介
  - 狭義の高可用システムに焦点
  - SRによるデータ持続性 + Linux-HAによるクラスタ管理(fail over)を紹介
    - JPUG(PG Day 2012)でも紹介された構成

高可用サービス二題　その二（続）

- Pacemakerの紹介
  - おおまかなアーキテクチャを解説
  - 管理されるソフトウェア(e.g. PostgreSQL)と連携するために、resource manager (RA) というソフトウェアが必要
    - 主なソフトウェアにはRAが用意されている
      Apache, drbd, PostgreSQL
- PostgreSQL用RAを用いた高可用クラスタの構成
  - 設定ファイルの解説
  - モニタ機能の紹介

SQL関連 三題

- チューニング
  - SQL HINTS, TIPS, TRICKS AND TUNING
    - Susanne Ebrecht
- 最適化
  - Query Planning Gone Wrong
    - Robert Haas
- 利便性向上
  - Estimating query progress
    - Jan Urbanski

SQLチューニング

- SQL HINTS, TIPS, TRICKS AND TUNING
  - Susanne Ebrecht (SQL標準化委員)
- 包括的な解説—単にSQLチューニングにとどまらず、DB設計までさかのぼって効率化の手法を概説
  - HW特性(メモリvsディスク)
  - 正規化の功罪
  - DB設計・運用(インデックスの作成法)
  - AP開発(SQL書換、Explainの解読と支援ツール)
  - 「講演を聞かずに資料だけ読んでも無駄」との警告
    - 説明っぽい記述は少ない→想像で補う

SQLチューニング（続）

- INDEXのTIPS
  - Create index concurrnetly のススメ
  - Multicolumn indexの使い方
- SQLの書換
  - 相関副照会→共通表式へ書換(80倍高速化)
- Explain解読
  - 出力(テキスト)の解読方法
  - 解読支援(可視化)ツールの紹介

SQL最適化

- MLでの相談事例168件の紹介
  - タイプごとに紹介。対処できるものも。
    - 設定(23)、本来遅いもの(23)、心得違い(22)、プランナ誤り(83)、バグ(14)
- 設定ミスの例
  - パラメタ設定不良(seq_page_cost, random_page_cost etc)
  - インデックス付与の不足
  - work_mem不足

SQL最適化（続）

- プランナの誤り(83件)
  - 概念誤り(28件)：等価で効率の良いプラン候補が生成できない
  - 見積誤り(55件)：見積値に誤りがある
    - 行数見積の誤り
      - 複数のカラムがある絞り込みで、カラムどうしの値の相関が活用できない
    - コスト見積
      - SELECT * FROM foo WHERE a=1 ORDER BY b LIMIT n;
      - インデックスbをスキャン→a=1で絞り込み　または
      - インデックスaでa=1を取り出してbでソートしてn件取り出す
      - 後者の方がより効率的になるときに、前者が選ばれる傾向がある
      - 複合もしくは関数インデックスを付与すると対処できることが多い
  - そのほか種々のパターンがあります

開発者必見かも

SQL関連の利便性向上

- SQL実行の進捗度合いが知りたい
  - 長時間実行しているSQLがあるとき
    - 今止めるべきか、もう少し待つべきか？
    - あと何時間待てば結果がわかるのか？
- 進捗の推定はできるか？
  - Explainの出力と対応付けたい
    - ○○%完了、あと○○秒かかる　など
  - 望ましい推定方法とは？
    - 時間とともに推定量が(適切な粒度で徐々に)増大
    - 低オーバヘッド、HW独立

SQL関連の利便性向上（続）

- 基本アイディア
  - Chaudhuri et al, “Estimating progress of SQL queries”, ACM SIGMOD, 2004.
  - Plan木の各ノード返すタプル数で進捗を表現する
    - 全ノードがタプルを返せば完了
    - GetNex()モデル
  - 各ノードについて
    - 返却する全タプル数を推定し、
    - 進捗率：返却されたタプル数/上記推定量

ここが難しい

続きはWebで

目指せ、10億テーブル

- PostgreSQL上で10億個の表を作ってみた
- Billion Table Project
  - Alvaro Hernandez Tortosa
- 最初に表を量産したのは、Josh Burkes氏
  - 1つの表のサイズには上限がある(32TB)
  - 表の数には上限がない→8Mテーブル作れた
- 動機は？
  - そこに山があるから
  - Josh Burkesには負けたくない

ちょっと聞いてみたい

もちろん、ネタ

目指せ、10億テーブル　(続)

- 10億へのいばらの道
  - ファイルシステムがパンクする
    - reiserfsで対処
  - 速度向上の対策
    - Fsync off
    - WALはRAMディスク(tempfs)に出力
    - VACUUMは止められない(XID周回対策)

ネタと言いつつ、
結構マジ

目指せ、10億テーブル　(続)

- 結果はどうなったか？
  - ちゃんと10億テーブルできたのか？
  - いったい何時間かかったのか？
  - 何バイトのDBができたのか？
  - psql “\dt”にどれくらい時間がかかるのか？

続きはWebで

ウケを取る


Black Hole Foreign Data Wrapper
Andrew Dunstan
andrew@dunslane.net
andrew.dunstan@pgexperts.com
PGX
POSTGRESQL

Blackhole foreign data wrapper

- まったくデータを取り出すことができない、書き込んでも消えてしまう、ブラックホールのようなforeign data wrapperの作り方を紹介
- 作りは極めてシンプル
  - 読み出し、書き込みとも何もしない
  - 読み出しはempty setを返す
- 何の役にも立たない…のではなくFDWを開発する際の最低限のスケルトンを提示する 本当は真面目なプロジェクト
- ソースコードも公開
  - https://bitbucket.org/adunstan/blackhole_fdw

コミュニティ動向

- 楽しいエンディングセッション
  - おなじみオークション
  - 恒例、新バージョンのポスター
  - 驚きの人事！

スライドショー
（内容は割愛）

Q&A

- Q1：コミュニティでの開発の方向性などの動向は？
- A1：developer meetingでは、EnterpriseDB社が問い合わせの並列化に着手することを宣言した。問い合わせ本体の並列化は難易度が高いので、当初はソートの並列化などから開発する見込み（藤井さん）
- Q2：カンファレンス参加者の年齢層は？
- A2：有名人には年長者もいるが、新コミッタのDavis氏のような若手もいる。年齢の分布は広い

おわり