

一歩前へ！ 知って得する DBアプリの環境改善

キムラデービー 代表 木村明治



本文書の一部または全部の転載を禁止します。本文書の著作権は、著作者に帰属します。

アジェンダ

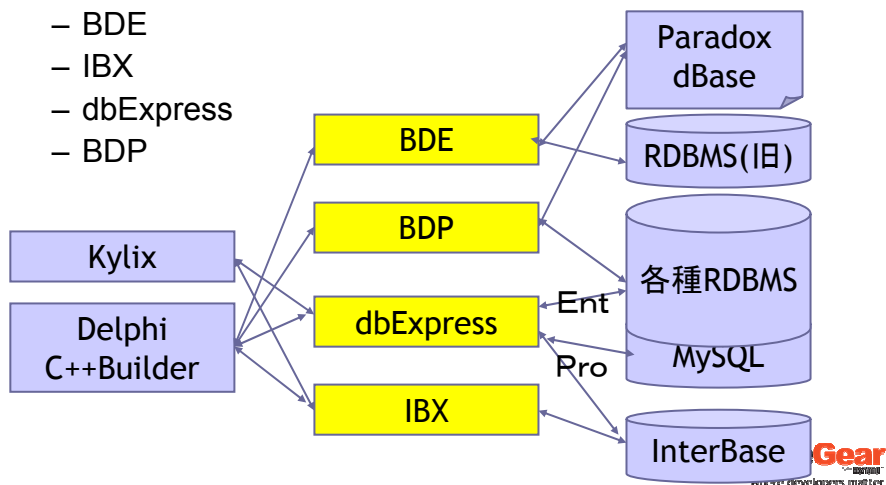
- はじめに
- 共通するところから
- 選択(SELECT)とは
- 更新(UPDATE)とは
- 挿入(INSERT)とは
- 削除(DELETE)とは
- まとめ

はじめに

データベースアクセス概略

- Borlandのデータベースアクセス

- BDE
- IBX
- dbExpress
- BDP

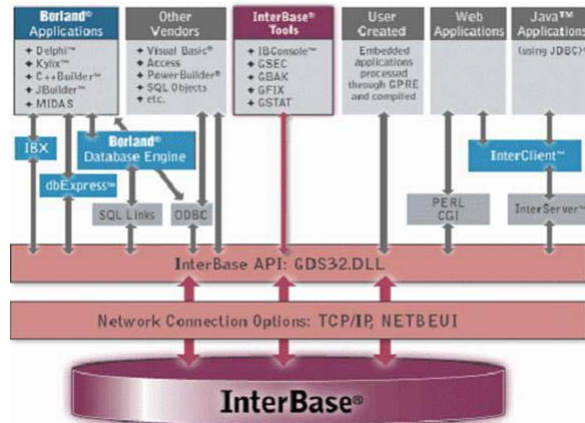


今回のセッション

- 各ミドルウェアの細部はとりあげません。
- まずは「データベースとデータベースアプリケーションの関係」
- 大きな背景を知ってもらいます。
- そして「共通認識」をもってもらいます。
- それにより、各ミドルウェアの機能が理解できるようになり、よりよくDB-DBアプリ環境が改善できることを目指します。

共通するところから

例) InterBaseへのミドルウェア接続



出典: <http://www.coadletter.com/article/0,1410,27570,00.html>

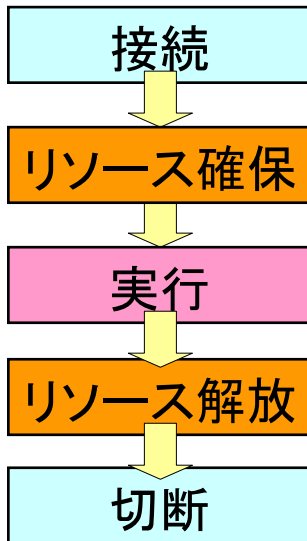
CodeGear
Where developers matter

ミドルウェアとワイヤプロトコル

- DBに接続するミドルウェアは千差万別。
- しかし、一番DBに近いところで考えると、DBとアプリケーションは「ワイヤプロトコル」でやりとりしている。
- 通常DBではワイヤプロトコルのC言語実装ライブラリを持っていて、各種ミドルウェアはその上に皮を被せる形で実装されている。
- 最近のミドルウェアでは、直接ワイヤプロトコルを扱うものもある。

CodeGear
Where developers matter

C言語実装ライブラリからみた手順

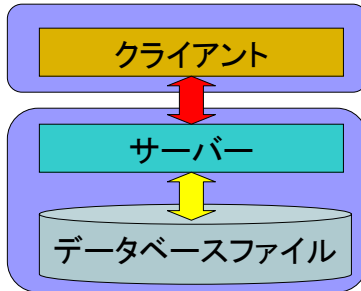


接続はコストが高い！

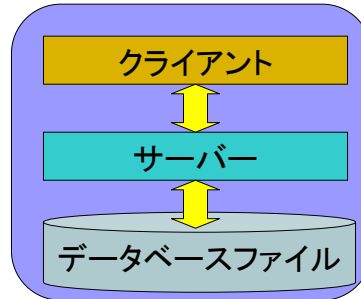
- 数秒かかることもある！
- 接続のコストを下げるには...
 - DBアプリとDBを同じマシンに置く。
 - 適用可能なら「インプロセスのライブラリ」を使う。
 - ミドルウェアや、DB自体に機能があれば「コネクション・プーリング」の機能を使う。

DBアプリとDBを同じマシンに置く

DBアプリとDBは違うマシン



DBアプリとDBを同じマシン

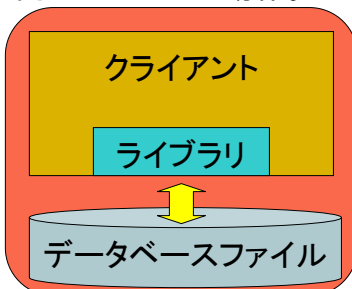


・DBによってはローカル接続がリモートよりもスピードがはやいものがある。

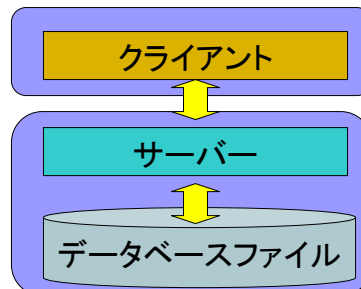
CodeGear
Where developers matter

インプロセスとC/S構成

【Embedded: インプロセス】
同一プロセスで動作。



【クライアント・サーバー構成】
独立したプロセスで動作。

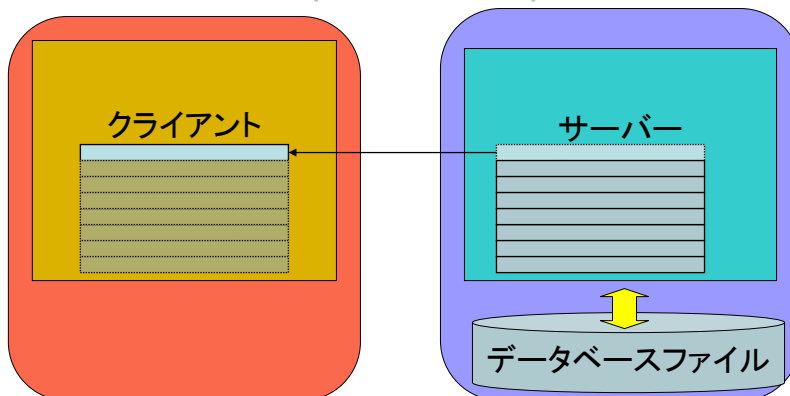


・InterBase/Firebird, MySQL, Apache Derbyなどは、両方の形態をとることが可能。

CodeGear
Where developers matter

選択(SELECT)とは

選択(SELECT)とは



・クライアント側の要求に応じて、サーバー側で行セットを作成して、それをクライアント側に順次転送していく。

いくつかの観点(1)

- 単方向カーソルか、双方向カーソルか
 - 単方向で足りるなら単方向を使う
 - 双方向はコストが高い(サーバー側で実装しているならまだよいが、クライアント側や、単純に全行キャッシュしている場合などはコストが高い)
- 不要なソートはしない。
- 行を全部読まない
 - できるだけ絞り込んだ形でSELECTする。
 - サーバー側ですべての結果行セットを生成しないように、クライアント側にすべての結果行セットを生成しないようにする。
- 不要であれば総行数をカウントしない。
 - DBによってはselect count(*)はコストが高い。

いくつかの観点(2)

- 同じ結果セットが返るSQLを何度も投げない。
 - ほとんど更新されず、総行数が知れているマスタ表など。
 - 厳密な同期が必要でなければ、一度クライアントに読み込んだものを使い回しても良い。
 - InterBase/Firebirdならば、イベントという力強い機能あり。

デモ&メモ

更新(UPDATE)とは

更新(UPDATE)とは

- 条件にあった行の値を更新する。
- 複数行、単一行
- 単一行→UIとからんで、特定の行を更新。
- 単一行のキー
 - ユニークなキー
 - 物理配置に関連するユニークな値(例:InterBase/Firebird, Oracle など)
 - 主キーやユニークインデックスなど
 - ユニークなキーがないと更新できない。
 - でも荒技でトライするものもある。

デモ&メモ

挿入 (INSERT) とは

挿入 (INSERT) とは

- 大量のINSERTは力業だ
 - インプロセスライブラリでINSERTしてしまう。
 - 専用のツールでINSERTしてしまう。
 - バッチINSERTが可能なら設定する。
- 当然オートコミットなんてもってのほかだ。
 - 今回はコミットについては詳細に説明しませんでした。

デモ&メモ

削除(DELETE)とは

削除(DELETE)とは

- 大量のDELETEは力業だ。
 - INSERTと同じフレーズ。
 - 表の行を全部消すなら
 - DBがサポートしていたらTRUNCATEでもOK。
 - 表のドロップ、再作成も現実味はある。
- 業務系なら、本当にDELETEすると困る場合もある。
 - 行削除カラムを用意して、論理的に消したりとか。

デモ&メモ

まとめ

まとめ

- ここまで聞いて頂ければ、以下のような初心者のそぼくな疑問に答えられるはず。
 - SELECTでデータの最初を取得するまでに時間がかかる。それ以降はすぐに取得できるのに。
 - Connectでなくpconnect使うのはなぜ？
 - 自分が使っているミドルウェアでは、大きな表を開くのに時間がかかる。
 - ローカル接続、インプロセスライブラリってなに？
- それをベースにDB-DBアプリ環境が改善できる!はず

デモ&メモ&トピック

Thank you