クラウドで
３Ｄレンダリング

Android Bazaar & Conference 2009 Fall
November 30 Tokyo

株式会社インタラクティブブレインズ
代表取締役　笠井秀行

1. 自己紹介

こんなことしてました

2003年 リッジレーサー

©BNG

2004年 機動戦士ガンダム3D

©BNG

2004年 ディープラビリンス

2005,6年 フィストグルーヴ1,2

2007年 はじめの一歩

©森川ジョージ/講談社/ I.B.

2008年 バーチャファイターモバイル

©SEGA

2008年 モバイルゴルフ･パンヤ！

©Ntreev

コンテンツプロバイダとして、携帯上で3Dを動かしてました。

こんなこともしてます

サーバーサイドレンダリング

モバイル3Dアバターシステム

２．Androidとクラウド

Androidといえば、クラウド。　…なぜ？

Androidのアプリは下記の主要な４種類のコンポーネントの組み合わせで構成される。

Activity	• アクティビティ • 画面やアプリそのもの
Intent Receiver	• インテントレシーバ • 情報の種類に合わせた挙動選択
Service	• サービス • バックグラウンドで動作し続ける
Content Provider	• コンテントプロバイダ • 入出力情報種類

各アプリ間はインテントのやり取りにより、起動や画面遷移を制御される。

Androidの仕組み

インテントにより、各アプリは疎結合に保たれ、マッシュアップが容易。

- 機能ごとにアプリの乗り換えが可能
- 機能単位で入れ替えが容易
- ローカル処理かクラウド処理かを気にせずサービスを構築
- そもそもミドルウェア以下のレイヤーもオープン化されていてマッシュアップできる。

- 例） FEPだけ入れ替えする Shimeji
- クラウドベースのFEP = Social IME とか容易に提供可

→ クラウドとの親和性が高い ！

Androidを使った実験的な取り組み

http://iappli.labs.nttdocomo.co.jp/

「みんなのドコモ研究室」

⇒　最新技術研究室へ

実機登録後にダウンロード

（残念ながらすでに実験は終了していますが。）

こんな感じ

ケータイ Watch

サイバード、Android向け「3D地図ビューアー」開発に協力

サイバードは、NTTドコモが実証実験を行うAndroid用アプリケーション「3D地図ビューアー」の開発に協力したと発表した。

「3D地図ビューアー」は、立体的に描かれた地図を表示するAndroid用アプリケーション。花火大会などイベント会場周辺を下見するといった利用方法が想定されている。先端的かつ試験的なアプリケーション/サービスを提供するユーザー参加型コンテンツ「みんなのドコモ研究室」で提供される。

従来は携帯電話側で3Dレンダリング処理を行うことが一般的とのことだが、今回のアプリのようなコンテンツを携帯上で処理するには、メモリやCPUに限界がある。そこで同アプリでは、インタラクティブブレインズが開発した「3次元サーバレンダリング技術」を利用し、街並みのデータはサーバー側で処理して生成され、通信経由で携帯電話上に表示される。なお、3次元地図データはゼンリンとジオ技術研究所の「Walk eye Map」を利用する。

「3D地図ビューアー」を利用できるのは「HT-03A」。導入手順は「みんなのドコモ研究室」のWebサイトで案内されており、HT-03Aのアプリケーション設定を変更した後、メールを送信してダウンロードサイトにアクセスする。

■ URL
3D地図ビューアー案内
http://trial.nttdocomo.co.jp/technology/vr/
サイバード プレスリリース
http://www.cybird.co.jp/press/content/c20090803.html
インタラクティブブレインズ プレスリリース（PDF形式）
http://www.ibrains.co.jp/pdf/20090803.pdf

(関口 聖)

2009/8/3 15:42

ケータイ Watch

【CEATEC JAPAN 2009】 ゼンリン、3D地図ビューアーなどを紹介

ゼンリンのブースでは、NTTドコモと協力して開発された「3D地図ビューアー」が展示されている。

「3D地図ビューアー」は、立体的に地図を表示することで、従来よりもわかりやすく現在地周辺を案内できることを目指したアプリ。Android搭載の「HT-03A」で動作するが、サーバー側で3D画像計算処理が行われることから、ゼンリンでは「どのような携帯電話でも高精細な3次元地図を表示できる」と説明している。

携帯電話側の加速度センサーを利用して、携帯電話の向きにあわせて3D地図の表示内容も動くようになっている。地図データはジオ技術研究所の「Walk eye Map」で、車のカーナビ向けでは既に実用化されているという。

このほか同社ブースでは、iPhone、Windows Mobile、携帯電話向けナビアプリを展示。携帯電話向けアプリは徒歩移動時の利用を想定する一方、スマートフォン向けアプリはカーナビとしての利用に対応していることが特徴という。

3D地図ビューアー

端末を上に傾けると、地図も上の方が表示される

３Ｄ地図ビューア

INTERACTIVE BRAINS CO.,LTD.

サーバサイド レンダリングシステム

NTTドコモが実証実験を開始した3次元化された実際の街並みを、ケータイ上で確認できるAndroidアプリケーション「3D地図ビューア」のサーバ側基幹部分となる主要機能の一つとして当社が開発した“クラウドエンターテイメント”テクノロジーの一部が採用されました。
※3D地図ビューアアプリは、株式会社NTTドコモが株式会社サイバードの協力を得て開発しました。

3D地図ビューア

QRコード

概要

サーバサイド3Dレンダリングシステムは、3D地図ビューアのような3Dマップに加え、その優れた3D動画の表現力により、3Dアバター、3Dゲームなど広範囲のエンターテイメントサービスに活用が可能となっております。

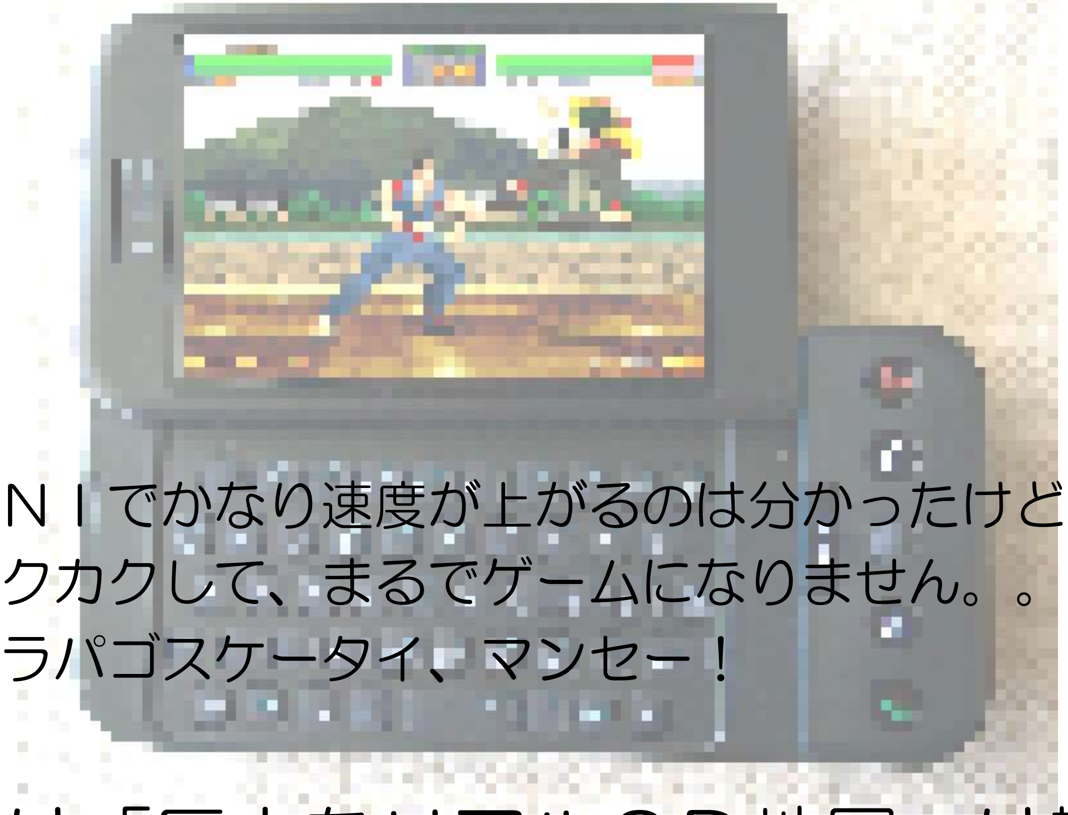
３．クラウドで３D

Androidの3Dレンダリング

- 公式でOpenGL/ES 1.0をサポート
  - 相当するソフトウェアレンダラが自前で用意されている。
  - もちろん、ドライバがあればハードウェアでもアクセラレーション可能。
  - NDK 1.6を使えば、C++(ネイティブ)から扱えるし、OpenGL/ES 1.1もサポート
- G1のCPUは、Qualcomm MSM7201A グラフィクスコアにImageon2300を搭載
  - 数万ポリゴン毎秒くらいは出せそう！（と思っていましたが …。）

で、動かしてみました。

- 試しに某有名対戦格闘をAndroidに
  - 実験的に移植してみたのですが。。。。。
  - ３Ｄ描画が遅い　＆　Dalvikによる演算がこれまた遅い。



- ＪＮＩでかなり速度が上がるのは分かったけど。。。
- カクカクして、まるでゲームになりません。。。
- ガラパゴスケータイ、マンセー！

これでは「巨大なリアル３Ｄ地図」は難しい。

クラウド側での3Dレンダリング

- １万円台強のＧＰＵボードを追加するだけでも、かなりのパフォーマンス。
  - ソフトウェアレンダリングだと、ぶっちゃけヘボいですが。
  - 標準的に、DirectX10 または OpenGL2.0サポート
  - GeForce GTS250の積和演算は500GFLOPS、HD4850では1.2TFLOPSに上る！
- クライアント環境に関わらず、いかようにもスケールアップ、スケールアウトが可能
- 数GByteの街の３Ｄデータも クラウド側なら楽々レンダリング

クラウドレンダリングでの要注意点

- マルチコンテキストがメイン
  - １台で複数のクライアントを捌く様なサーバー動作となるのが一般的。コンテクストスイッチングを意識したサーバー動作にする必要がある。
- 簡単にスケールできるとは限らない。
  - ハードウェア増強などで対応できるうちはスケールアップは容易。しかし、上記の観点からスケールアウトは工夫が必要。
- レスポンスとのトレードオフを考慮
  - それなりに帯域を食うので、ユーザーストレスを減らすための仕掛けが必要。
  - 例）ローカル側でのインタラクションを許せるような形の複数データをまとめて送るなど。

4．クラウドとの役割分担

Android側でやるべきこと

- そもそも、ローカルでしか得られない情報
  - 各種センサーの情報
    - キー・カーソル・タッチパネル　（ユーザー操作）
    - カメラ・音声入力
    - ＧＰＳ・加速度・方位・地磁気　（ロケール情報）
    - 血圧・脈拍・体温　（ヘルスチェック）
- ユーザーとのインターフェース
  - フィードバック
    - スクリーン・サウンド・振動
  - クイックレスポンスが要求される部分

クラウド側でやるべきこと

- レスポンスはそこそこでいい場合
  - 現状では、帯域、レイテンシが馬鹿にならない。
- 大規模なデータ容量が必要な処理
  - ストレージサービス
- 高速に大量な演算が必要な処理
  - データベース検索・マッチング
- クラウド側に情報を保存したい場合
  - シンクロナイズサービス
- 汎用化できる処理
  - クライアントに依存しない処理

Androidの性能

Android端末の基本スペック

- ARM9　clock 200-400MHz
- ROM　64M-128M
- RAM　64M-128M

1つのアプリケーションあたり8MB (on Dalvik)

Processor	Qualcomm® MSM7201A™, 528 MHz
Operating System	Android™
Memory	
ROM:	256 MB
RAM:	192 MB

クラウド側の性能

Amazon EC2のVM基本スペック

(Large Instanceの場合　$0.4/1時間)

- 4 **EC2** Compute Units (2 virtual cores with 2 **EC2** Compute Units each)
- 7.5 GB memory
- 850 GB instance storage (2×420 GB plus 10 GB root partition)
  64-bit platform

Android vs クラウド

INTERACTIVE BRAINS CO.,LTD.

プロセッサ名	IPS	IPS/Hz
Intel Pentium III	1,354 MIPS at 500 MHz	2.7 MIPS/MHz
Freescale MPC8272	760 MIPS at 400 MHz	1.9 MIPS/MHz
AMD Athlon	3,561 MIPS at 1.2 GHz	2.967 MIPS/MHz
Pentium 4 Extreme Edition	9,726 MIPS at 3.2 GHz	3.039 MIPS/MHz
ARM Cortex A8	***2,000 MIPS at 1.0 GHz***	***2.0 MIPS/MHz***
Xbox360 IBM "Xenon" Triple Core	19,200 MIPS at 3.2 GHz	6.0 MIPS/MHz
PS3 Cell BE (PPE only)	10,240 MIPS at 3.2 GHz	3.2 MIPS/MHz
AMD Athlon FX-60 (Dual Core)	18,938 MIPS at 2.6 GHz	7.283 MIPS/MHz
Intel Core 2 Extreme X6800	***27,079 MIPS at 2.93 GHz***	***9.242 MIPS/MHz***
Intel Core i7 Extreme 965EE	76,383 MIPS at 3.2 GHz	23.860 MIPS/MHz

てゆーか、単純に比較するべきものでもない。

⇒　互いに相補的な関係にある！

Android vs クラウド

技術	通信速度	年代
2G PDC	9.6k	1995
2G PDC Packet	28.8k	1997
3G W-CDMA	384k	2001
3G HSPA	3.6M	2007
3G LTE(Super 3G)	50M 〜 100M	2010
4G	100M 〜 1G	2011策定

LTEがくれば、クラウドとAndroidは、もっと密接につながる事ができる。

5．クラウドで、○○

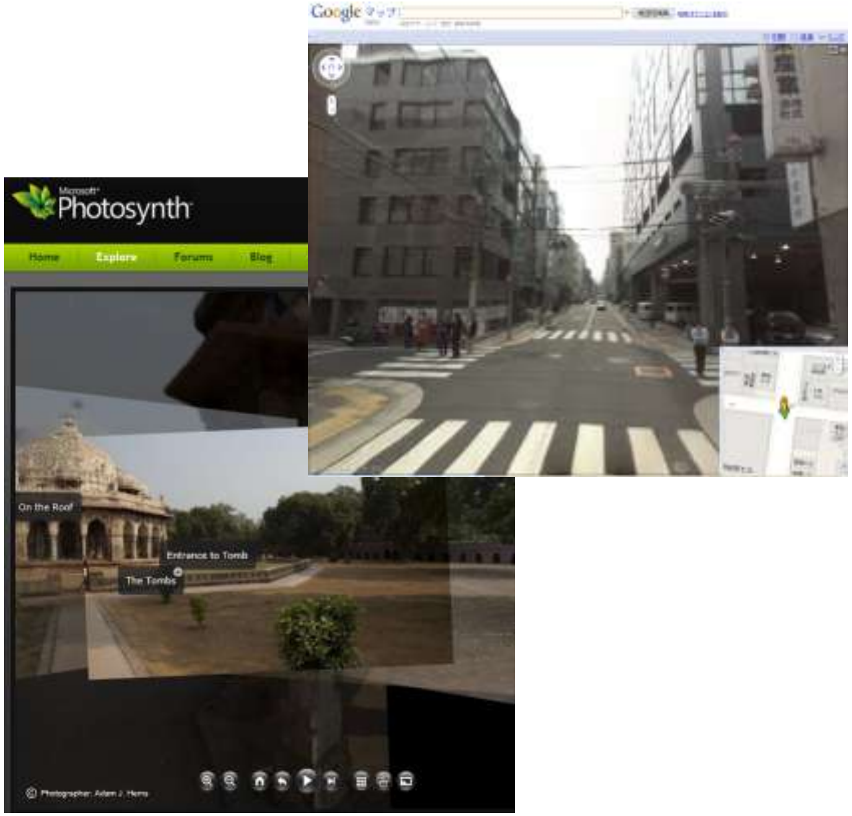
「Layer」
ARアプリの代名詞に。

「ポケナビ with 羅針盤」
AndroidでさくっとMashUpの実例。

見せ方の差はあれど、リアルタイムに必要な情報を巨大なデータベースからスクリーニングして返すことはクラウド側の仕事の基本。

「モバゲー」も「Any」も
アバターを3Dでレンダリング。

EC向けの画像変換出力も
一種の画像処理(セラーテム社)。



GoogleStreetViewや、MSのPhotoSynthは、描画のほとんどをクライアント側と協調しているが、広義のクラウドレンダリングとも言える。

実施例その３　画像認識(1)

「顔ちぇき」(J-Magic)は、顔画像を認識し類似判定。

「フォトナビワイン」(ゼータブリッジ)は、ワインラベルで、DB検索。

実施例その３　画像認識(2)

「SR Engine」(金村氏)での画像認識。初期は、サーバー連携で認識していたが、iPhone3GSではスタンドアロンに

AR的なアプローチで、「東のエデン」を髣髴とさせる。

「 shazam」は音楽のフレーズをクラウドに投げて音楽DBを検索。

「AmiVoice」(アドバンスドメディア)社の音声認識エンジン。スタンドアロンとサーバーへの分散型がある。

実施例その５　音声合成

INTERACTIVE BRAINS CO.,LTD.

携帯翻訳 J-Serverポケット
(KODENSHA)

バックエンドはPENTAXの
音声合成エンジン

ナレッジクリエーション社
の音声合成はMashUpAwards5にAPI提供

実施例その６　ストリーミング

汎用的なものは割愛。ここでは、独自にクライアント協調して動くものを紹介。

Live Server(セーバー)
携帯アプリ向けのエンジン。字幕などがつけられる。

anymo(フレックスファーム)管理情報などを一括して扱える。

実施例その７　ゲームサービス

OnLive
2009年冬にはサービス開始？

すでに、そうそうたるメンバーがコンテンツ供給を明言
OnLivElectronic Arts（EA）, UBISOFT, Take-Two Interactive Software, Warner Bros. Interactive Entertainment, THQ, Epic Games, Eidos, Atari, Codemasterse

まとめ（1）

- マッシュアップは既に十分にコモディティ化
- Webアプリ側もどんどん連携強化の方向に
  - JavaScript
  - Flash
  - Silverlight
- 更に、HTML自体がHTML5で強化
  - WebSockets
  - WebStorage
  - WebWorkers
- 低位レイヤーからこれらをさらに使いこなせる。

まとめ（2）

- クラウドと連携させたサービスが作りやすいのがAndroidの醍醐味。
  - 単にActivityを実装するのでなく、ServiceやContentProviderをうまく使いましょう。
- クラウド側とAndroid側での特性を考慮すべし。
  - Androidクライアントとクラウドでは、それぞれ得意分野が違います。
- 臆せずとりあえず作ってみるべし。
  - クラウドもAndroidも個人で簡単に扱える時代です。Webサービス・組み込みいずれもオープンな環境を利用して、やりたいようにやってみましょう。

御清聴ありがとうございました。