



携帯電話での ネットワークゲーム制作技術

～「カルドセプトモバイル」を例に～

株式会社ドワンゴ
第三制作部 ゲームズセクション 田巻敏直
ソリューションビジネス部 アプリケーション開発セクション 荻野 洋

セッション概要

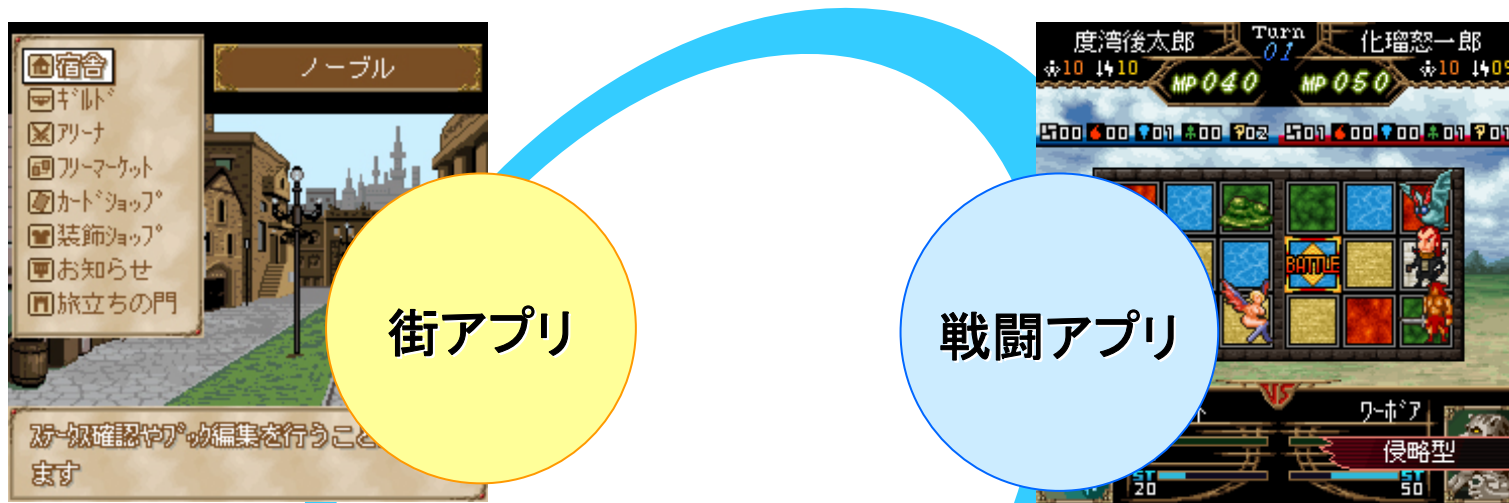
◆ コンシューマからモバイルへの落とし込み(約20分)

株式会社ドワンゴ
第三制作部 ゲームズセクション
田巻敏直

◆ 携帯電話でのネットワークゲーム制作技術(約60分)

株式会社ドワンゴ
ソリューションビジネス部 アプリケーション開発セクション
荻野 洋

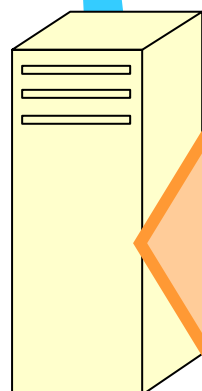
カードセプトモバイルの紹介



街アプリ

戦闘アプリ

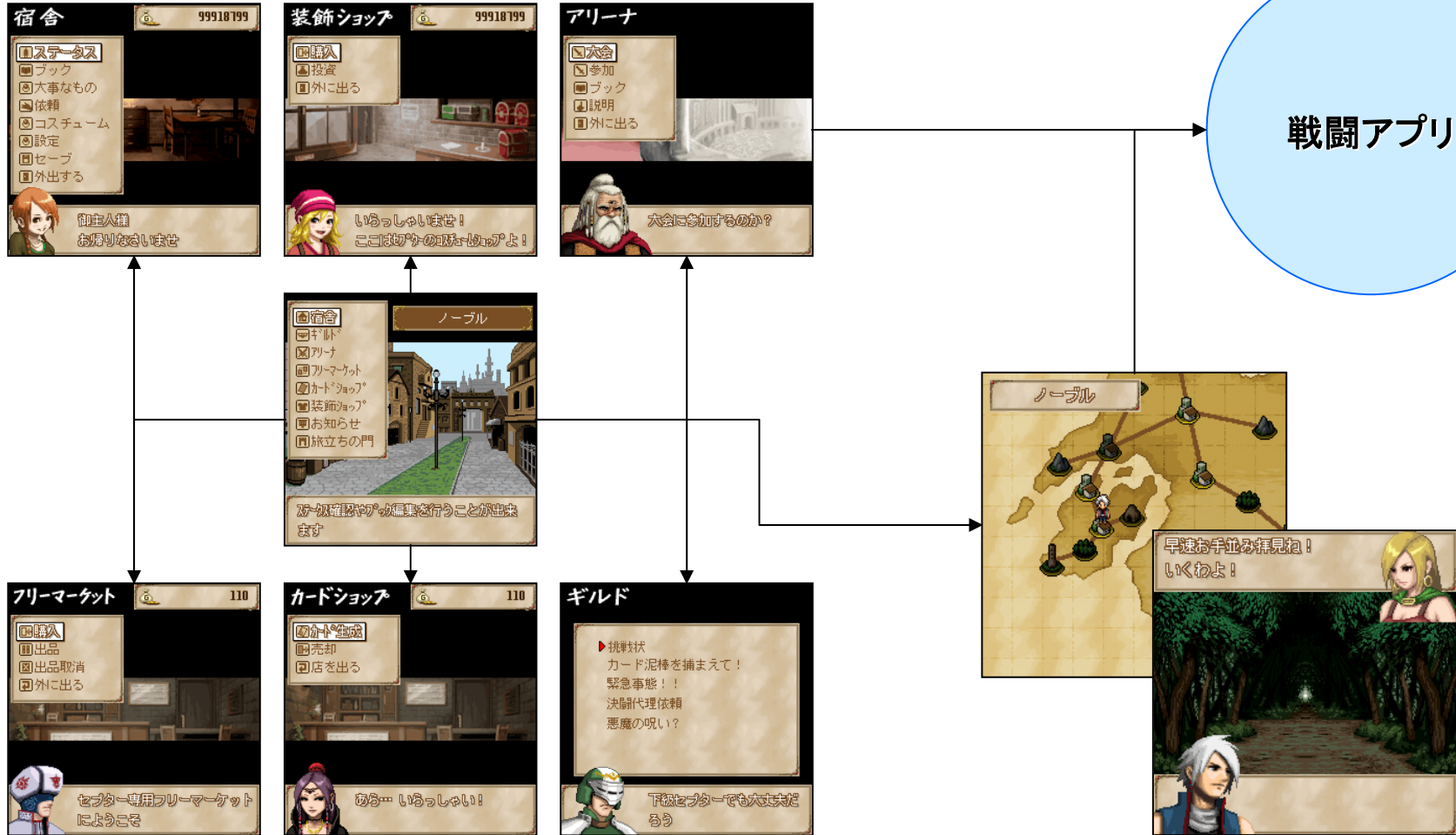
通信機能・アプリ連動機能を利用して、サービスを提供



サーバー

- ・通信対戦大会の実施
- ・フリーマーケットの実施
- ・新たな追加イベントの配信 Etc...

カードセプトモバイル 街アプリ



カードセプトモバイル 戦闘アプリ



【ファーストセット】



【ドローフェーズ】



【セットフェーズ】



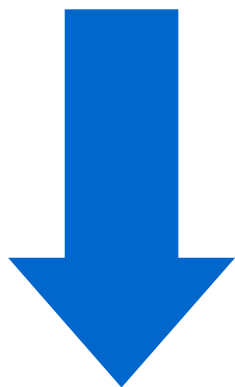
【アクションフェーズ】

全てのフェーズが終了すると、後攻側へとプレイ権利が移行

コンシューマからモバイルへ！（1）

◆ 携帯アプリ史上最高カードゲームを目指す

- 通信を最大限に活かしたシステムの構築！
- 何時でも手軽に誰でも楽しめる携帯の特性を生かす！
- 携帯アプリだからこそその要素を入れる！



モバイルに適した新しいシステムを再構築する必要がある！

コンシューマからモバイルへ！（2）

◆ カルドセプトで核となる部分を抽出

- カードを使用した対人対戦
- 事前に戦略を練り、使用カードを選択するデッキ構築作業
 - ↳ おまけで、カード収集する楽しみがあり、デッキ構築作業が更に楽しく！

◆ 企画当初は大規模ネットワークゲーム？

- ユーザー同士のコミュニケーション要素を入れたシステム

規模が大きすぎて、運営が困難、またはバランス調整が不可能になる事が予想された為に削除！

コンシューマからモバイルへ！(3)

◆ 通信戦闘実装の為の仕様作成

- 通信料 ≤ 乱数の面白み 通信対戦の【同期】
- 通信対戦中の通信切断に備えた仕様 復帰処理

◆ 誰でも楽しめるアプリを考える！

- カルドセプトの世界観、カードゲームの理解の為のイベント
- 通信対戦をせずとも長く楽しめる要素

コンシューマからモバイルへ！（４）

◆ モバイル版としてのオリジナリティ

- 誰でも簡単に参加できるオンライン要素 フリーマーケット

◆ ネットワークゲーム的な観点から

- バージョンチェック、リソースチェックの仕様入れ込み
- サーバーにイベントデータを置く事で追加配信が可能に

コンシューマからモバイルへ！(5)

◆ 携帯アプリ必須機能 どこでもセーブ

- 何時でもどこでもプレイする携帯アプリには必須機能！
- 結果としてカードやお金のデュープバグ対策にも！

◆ 通信頻度の多さを技術でカバー！

- マルチスレッドを使用する事で動作に支障なく通信可能に

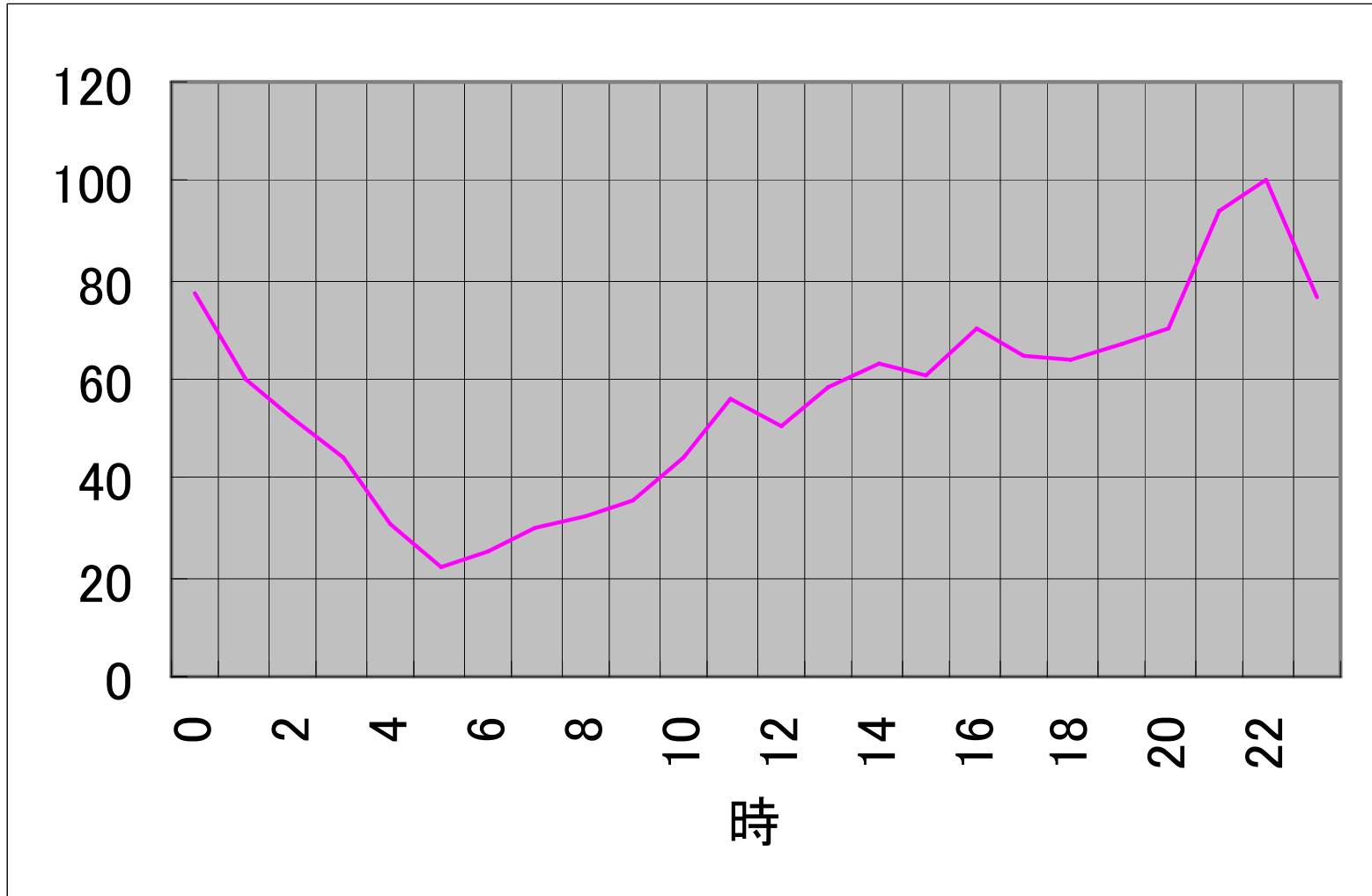
携帯アプリネットワークゲームの運営

◆ 頻繁なバージョンアップは必須！

- ユーザー意見を反映したバージョンアップ
- 起きてしまった不正に対する対応

ネットワークゲームを運営する上で
最も重要な事は鮮度を保つ事！

カードセプトモバイル1日のアクセス数



携帯電話でのネットワークゲーム製作技術

株式会社ドワンゴ

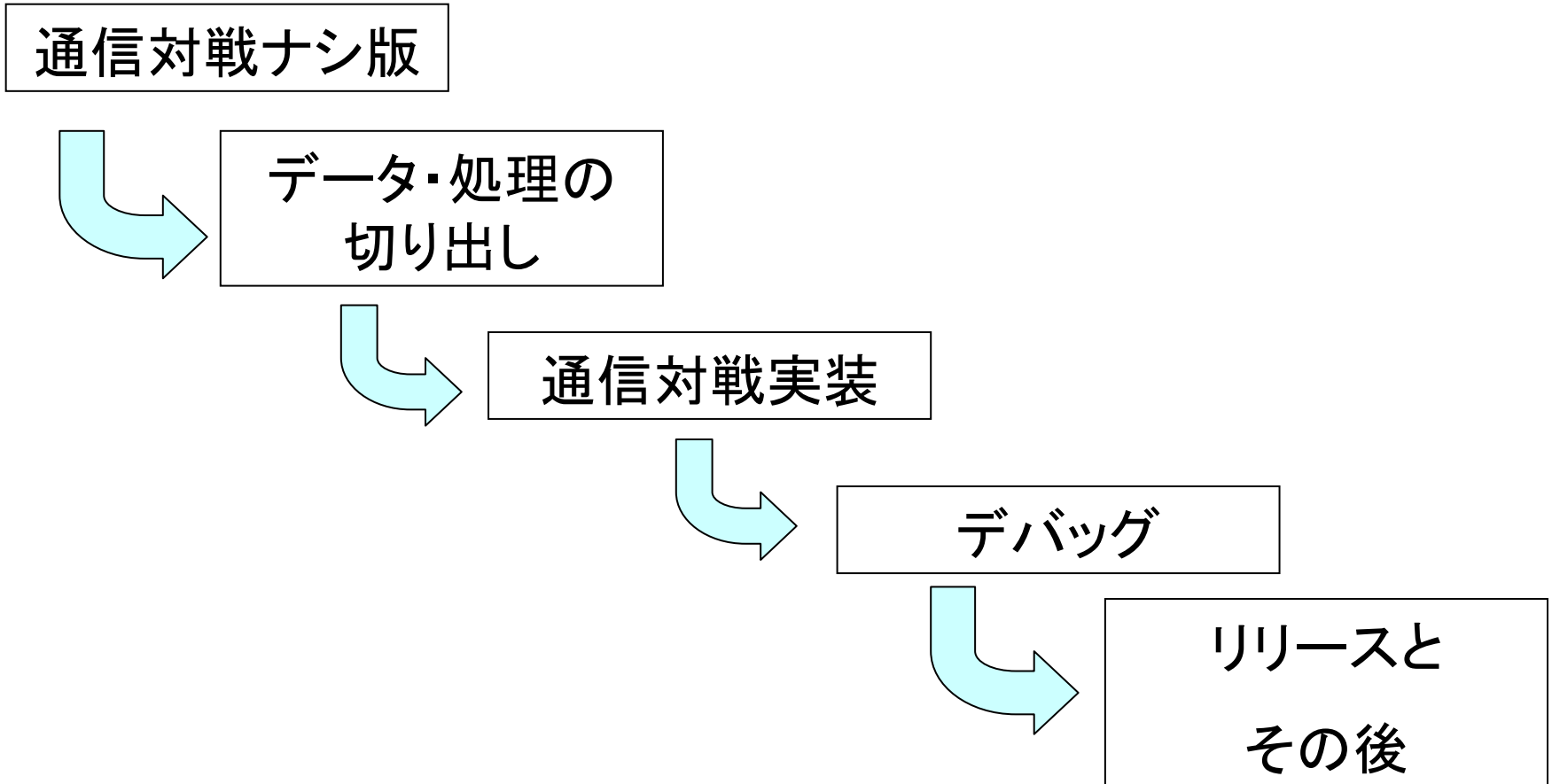
ソリューションビジネス部 アプリケーション開発セクション 荻野 洋

もくじ

- カルドセプトモバイルの実装(概要)
- 問題点と解法
- 技術

カードセプトモバイルの実装(概要)

戦闘アプリ実装手順



(1)通信対戦ナシ版

- アプリ単独で動作するモノ
 - TOKYO GAME SHOW 2005版

- メモリ管理
- 画像キャッシュ
- アプリサイズ圧縮

- 水色＝「技術」で解説している項目

(2)データ・処理の切り出し

- フェーズごとに処理を分割
 - フェーズ=プレイヤーの行動単位
 - いわゆるリファクタリング

○アプリ・サーバでのデータとメソッドの共有

○ 水色=「技術」で解説している項目

(3)通信対戦実装

- 通信が遅い
- 通信切断時の対応

- 起動時処理の調整
- 通信対戦時の処理
- ランダム同期

○ ピンク＝「問題点と解法」で解説している項目

○ 水色＝「技術」で解説している項目

(4)デバッグ

- データが途切れる
- データが混ざる
- 端末依存の問題

- ピンク＝「問題点と解法」で解説している項目

(5)リリースとその後

○バランス調整

- カードデータ調整
- 先攻・後攻の有利・不利の調整

○機能拡張

- 端末内対戦(対NPC)の途中セーブ

街アプリ

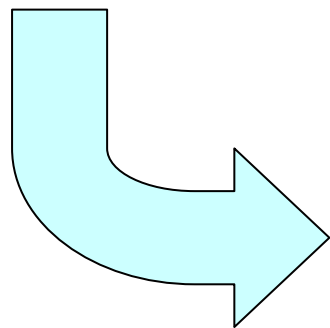
- 戦闘のマッチング
- セーブタイミング
- イベントでのデータ同期

- 水色 = 「技術」で解説している項目

問題点と解法

問題点(1)通信が遅い

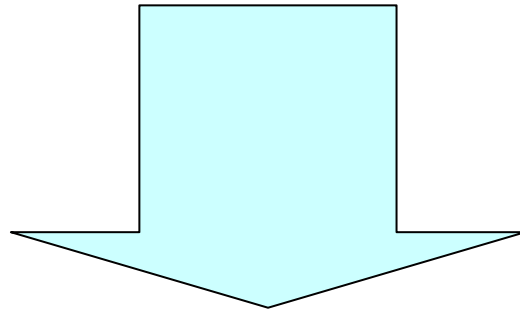
- 接続までに1～3秒
- つながってしまえばそれなりに速い
- データ量よりも通信回数が問題



- 常にスレッドで通信
- とりあえず画面で何か動かしておく

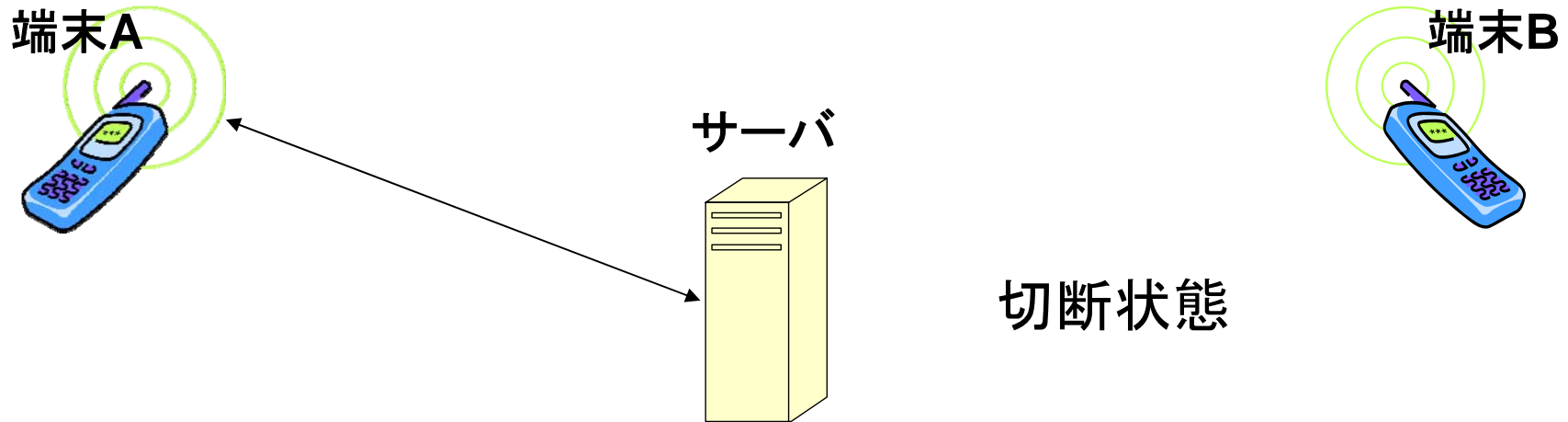
問題点(2)対戦中の通信切断の発生

- 電波状態の悪化
- バッテリー切れ
- プレイヤーによる意図的なアプリ終了



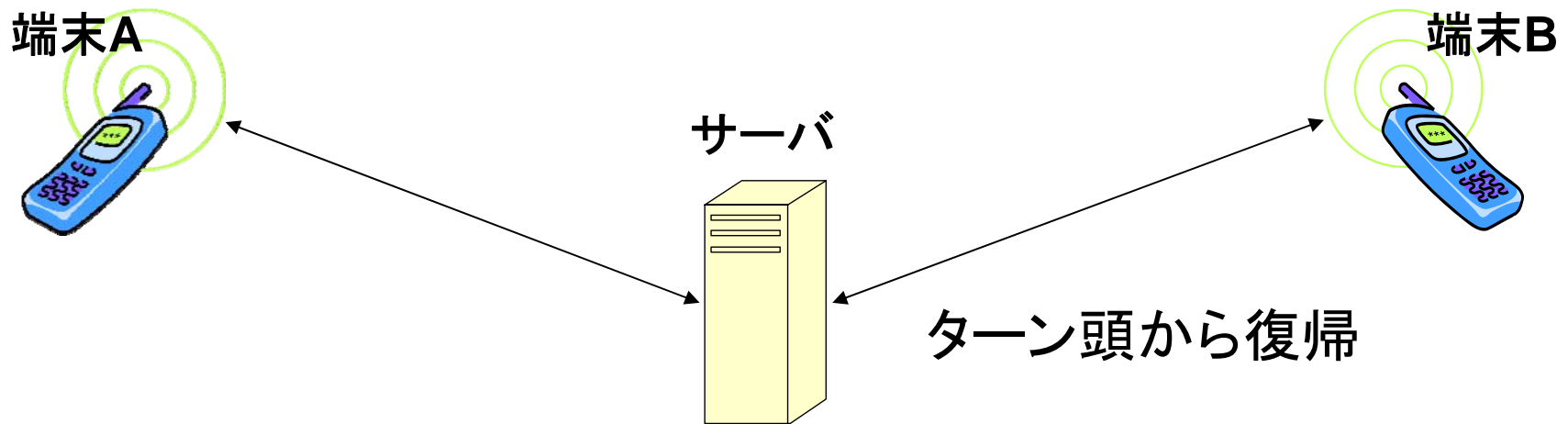
快適に対戦を継続させる仕組みが必要！

問題点(2)対戦中の通信切断の発生 対応編



- 75秒でタイムアウト(制限時間の1.25倍)
- タイムアウト後にサーバが代打ち
- 端末Aは端末Bの切断を意識しない

問題点(2)対戦中の通信切断の発生 対応編



- 端末Bが再接続
- ターンチェンジ時にサーバが復帰許可
- 復帰情報をシリアルライズして送信
 - 通信回数を減らすため

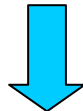
問題点(3)データが途切れる

○例

「BTL_SET_WAIT」を送りたい



「BTL_SET」まで送信した時に電波状態悪化



サーバは「BTL_SET_WAIT」ではなく「BTL_SET」という別コマンドとして解釈！！



○原因

- 電波状態の悪化
- 電源ボタンなどの押下
- 電話の着信
- Etc....

問題点(3)データが途切れる

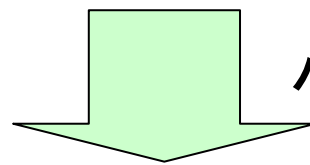
○解決策

- Content-Lengthによりチェック
- チェックサム、CRCなどの導入
- 末尾に「EOF=1」を追加
- など.....

常にデータは途切れるモノとして考えよう！

問題点(4)データが混ざる

- 「端末A」のデータと「端末B」のデータが混ざって届く場合がある！



バージョンが古かった……

サーバーのバージョンを新しくして解決

常に最新安定版を使いましょう

問題点(5)端末による問題

- ヒープが1.5Mしか無いD900i
 - メモリ消費を抑えたため問題なし
- 全体的に遅いF900i、D900i
 - オートフレームスキップで回避
- 通信時にメモリ不足で落ちるN900i、P900i
 - 起動時にメモリを100k bytes確保
 - HttpConnectionのopen前に解放 & System.gc()
 - 通信終了まで新たにメモリは確保しない
 - 通信終了後に100k bytes確保し直し

問題点(6)その他の問題

○通信しすぎは避ける

- 秒間一回程度の連続アクセスはアウト
- BREWだと3Mbytes／日の制限有り

でもサーバーの負荷テストは必要.....

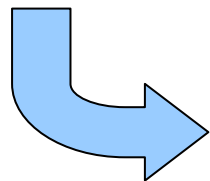
負荷テストは極力PCでやりましょう

- カルドセプトモバイルの場合
 - PC版クライアントにて2000人同時接続実験(1000対戦)
 - 数秒ごと行動する実装(瞬間同時アクセス数100程度)

技術

技術(1)メモリ管理

極力、動的に確保しない！！



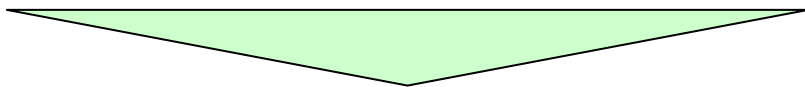
メモリ関係が弱い端末が多い

○カードセプトモバイルの場合.....

- 起動時におおむね確保(汎用画像、サウンドetc...)
- 極力オブジェクト指向で書かない
 - クラス数削減(街、戦闘共に8クラス)
- 常にメモリの確保・解放を把握

技術(2)画像キャッシュ

- 画像を全てアプリに持てない
 - カドセプト=カード・キャラ画像 合計390kbytes

- 
- × 使う画像を毎回ダウンロード
 - × 画像を減らす
 - 画像を一定枚数保存しておく(キャッシュ)

カドセプトモバイルの場合(カード画像キャッシュ)

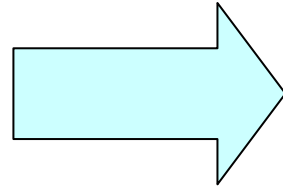
- 700バイトごとの領域とヘッダを確保
- ヘッダは画像サイズと状態(「空き」「永続」「一時的永続」「存在」)

技術(3)アプリサイズの圧縮

- proguard.....軽量化・最適化・難読化ツール
- 7zip.....圧縮率の高いzipアーカイバ

街アプリ jar サイズ

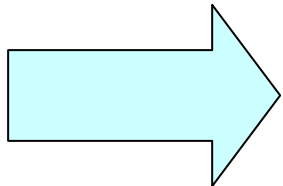
118828bytes



97911bytes

戦闘アプリ jar サイズ

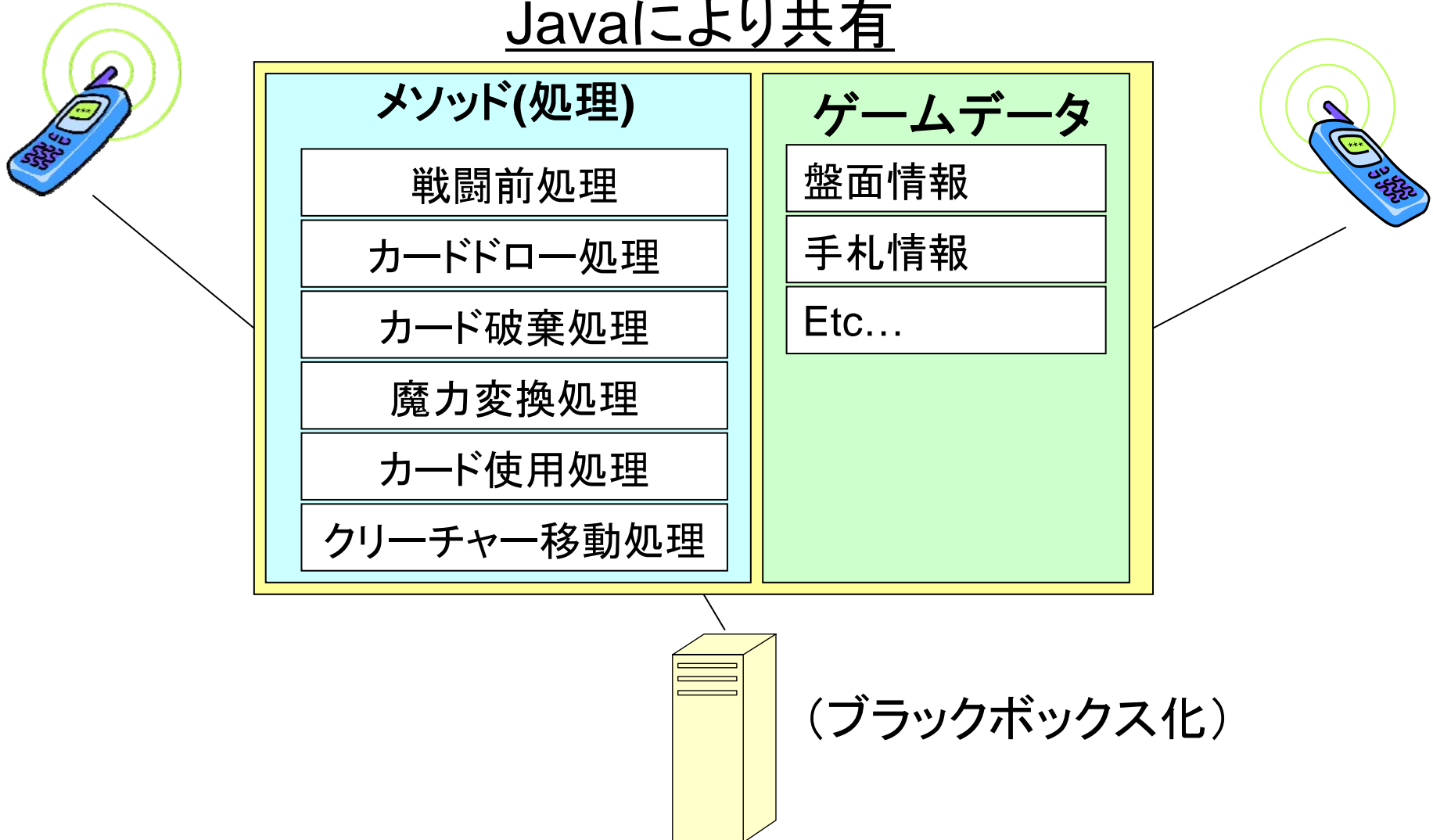
101698bytes



90770bytes

技術(4)アプリ・サーバでのデータとメソッドの共有

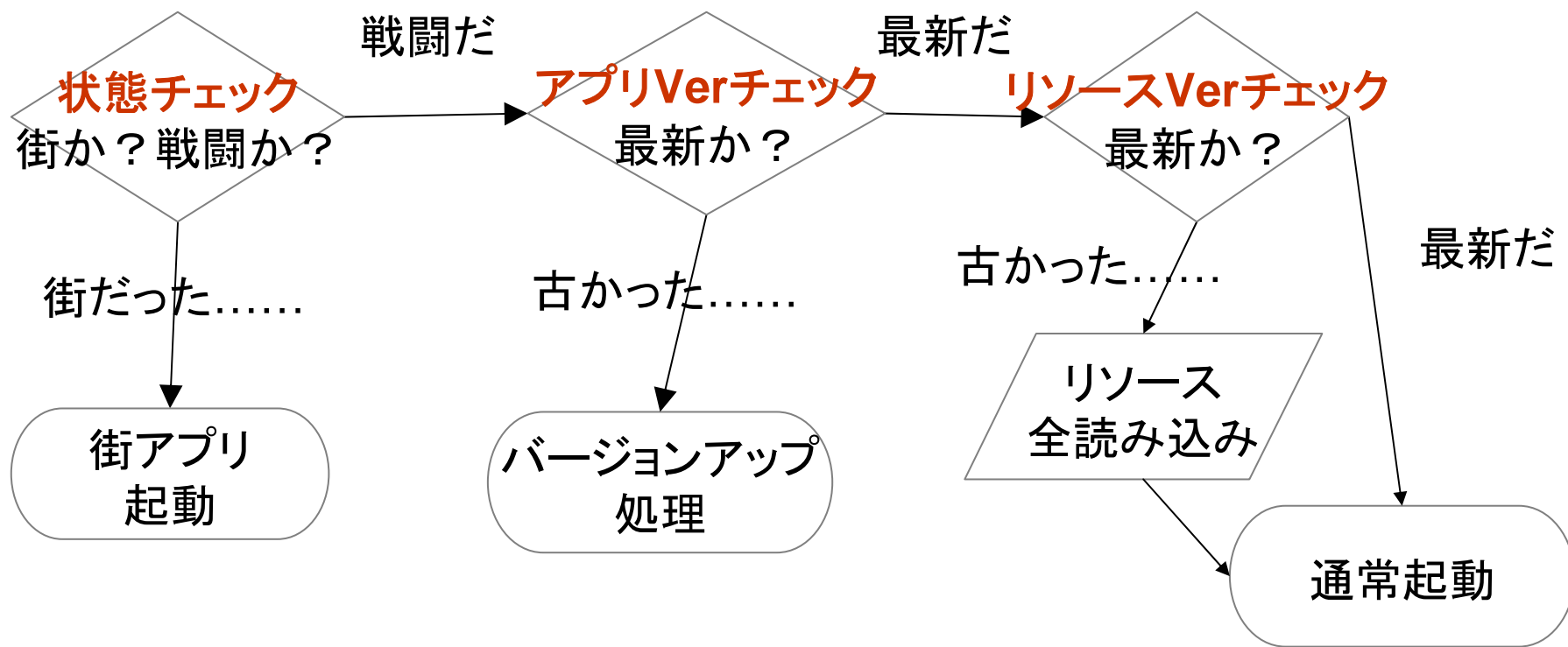
Javaにより共有



技術(5)起動時処理の調整

○状態&アプリ&リソースバージョンチェック

カルドセプトモバイル・戦闘アプリの例



技術(6)通信対戦時の処理

ターン形式の対戦

○ターン形式とは？

- プレイヤーが順番に行動(「行動側」と「待つ側」)
- 複数プレイヤーが同時に行動しない

○長所

- 同時に行動しないのでわかりやすい
- 実装がシンプル

○短所

- 待つだけの側が退屈
 - 「閲覧モード」を実装して回避(カルドセプトモバイル)

技術(6)通信対戦時の処理:実装案

○どうやって行動を送受信するか？

- サーバーが行動情報を蓄積
- その情報を端末側が必要に応じて取得
 - 常に端末側がサーバーに情報を要求(サーバー側から端末に接続する事は出来ない)

技術(6)通信対戦時の処理:実装例

一般的なターン形式対戦の実装例

行動側

行動したらデータをサーバに送る

※15秒ごとに、生存確認の通信

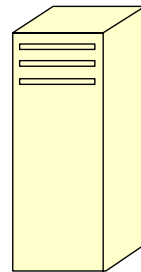


手札1を使ったヨ!

サーバー

行動側の手を1ターンぶん保存

1. 山札から引いたカード情報
2. 手札のカードを使った情報
3. 盤上のキャラクターを移動した情報



待つ側

相手の行動情報を問い合わせる(3秒ごと)



行動側はカード引いた?

技術(7)ランダム同期

○ランダムについての確認

- アルゴリズムとランダムシードが同じなら、常に同じ値を返す
- サーバと端末はjavaで動作

○復帰時.....

- サーバはランダム呼び出し回数を渡す
- 復帰側はその回数だけランダムを呼ぶ

処理の簡略化&パケットを少々節約

技術(8)戦闘のマッチング(1)

○ 対人対戦

ユーザ1	ユーザ2	条件
------	------	----

カードセプトモバイル
での例

フレンド対戦	ユーザID1+ユーザID2(昇順)
クラス対戦	クラス番号
フリー対戦	フリー用番号
大会	大会用番号

○ 条件不一致

- 自分をユーザ1に、条件に検索条件を入れてレコード作成

○ 条件一致

- 自分をユーザ2に入れて戦闘状態へ

技術(8)戦闘のマッチング(2)

○ NPC対戦の場合

- →「ユーザ1」に自分のIDを設定し、「ユーザ2」にNPCのIDを設定して戦闘状態へ。

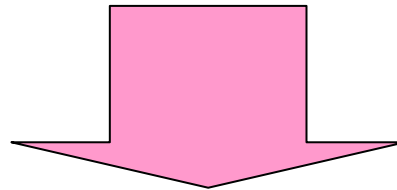
カードセプトモバイルの情報

- アプリのポーリングは3秒
- サーバーのタイムアウトは30秒
- 対戦意思確認などがあると、もっと複雑になります

技術(9)セーブタイミング

同期を取る必要がある場合は常にセーブ！

- バッテリーが切れたり.....
- 急に用事が出来たり.....



不測の事態に対応

技術(10)イベントでのデータ同期

○お金やアイテムをイベント中に入手？

- イベントをキャンセルされたら戻せない！
- 入手コマンドをサーバに送信する実装では多重取得などの問題が発生！

カドセプトモバイルの場合.....

「大事なもの」で回避

○「大事なもの」とは？

- 単なる名前付きのフラグ
- イベント終了時にお金やアイテムに変換(サーバーサイド)

まとめ

携帯アプリネットワークゲームの未来(1)

◆ Bluetooth・Wi-Fi実装による ローカルネットワークゲームの構築

- 低コストな多人数プレイゲームが可能に！
- 動的(アクション要素の高い)な対戦ゲームも可能
- ローカル接続でアプリを配布したりも出来るかも！
- MODの配布も可能！



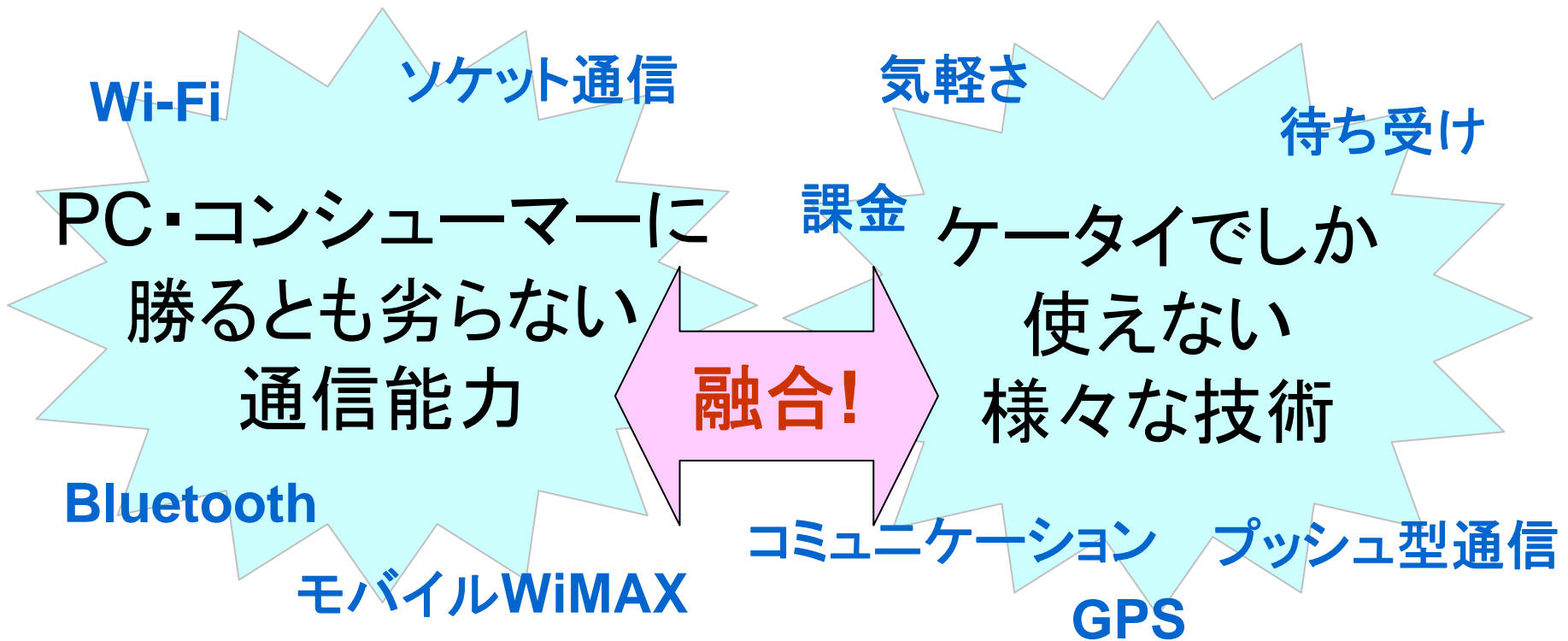
ローカルネットワークだけじゃない

携帯アプリネットワークゲームの未来(2)

◆TCP実装で更に広がるネットワーク

- ターン制ではない動的なネットワークゲームが可能に！
- 本格的な携帯アプリMMO時代の到来もそう遠くは無い
- 携帯特性である通信の手軽さ、課金システムの容易さ
- 誰でも持っている便利なコミュニケーションツール

携帯アプリネットワークゲームの未来(3)



新世代ネットワークゲームの幕開け

ご質問をどうぞ！

連絡先

本日はセッションご参加ありがとうございました。
ご質問は下記メールアドレスまでご連絡ください。

株式会社ドワンゴ

田巻敏直 toshinao_tamaki@dwango.co.jp

荻野 洋 hiroshi_ogino@dwango.co.jp