

クラウド時代の UI デザイン再検討

矢野りん[†]

卓上式パソコンは「卓上」で行う行為をそのまま操作性に置き換えていた。卓上に、道具がある。ととらえていたのである。来るべきクラウド時代、道具は雲の上にあるとしたら、それを仮想の卓上に置くことは機能的だろうか？雲（クラウド）を見るのはわたし（ユーザー）である。そしてわたしは机の前にいるとは限らない。いつか、任意の時間に、どこか、任意の場所で雲を見ている。ならばいっそ、時間と空間をそのまま操作性に置き換えたほうが機能的だと考えることはできないだろうか。本稿ではこの仮説に基づいて、UI のインタラクションデザインを再検討する。

Reexamination of User Interface design for Cloud computing era

Yano Rin[†]

A desk-top personal computer kept replacing the act of doing by "Desk-top" with GUI operation.
Will it be functional to put it on virtual desk-top if the tool is on the cloud in a approaching Cloud computing age?
It is me (user) that see the cloud. And I am not necessarily be in front of the desk. The cloud is seen in an arbitrary place somewhere at arbitrary time. Then, why don't you think that it is functional to replace time and the space with desk-top?
In this text, I try to reexamine about The interaction design of User Interface based on this hypothesis.

1. はじめに

1.1 クラウド時代におけるユーザーと IT の関わり

1980 年代「computer,for the rest of us」のキャッチフレーズと共にパソコンのコモディティ化を目指したアップルは 2008 年、携帯端末「iPhone」を発表した。iPhone は同社の OS である Mac OS X 同様、マニュアルのいらない単純さを追求した設計となっている。その単純さは機器の操作方法に限らず、各種サービスとの連携、さらにアプリケーションの操作性にもおよぶ。iPhone からアップデートしたスケジュールは mobile me という同社のサービスを通じて自宅や会社の PC にも反映される。この操作はシームレスに、とても単純な操作で完了してしまう。アップルはこの mobile me を「クラウド」と呼ぶ。

1996 年「世界中の情報を整理し、世界中の人々がアクセスできて使えるようにすること」を目的として誕生した google は、検索エンジンをはじめ、数多くのウェブアプリケーションおよびサービスを公開し続けている。メール機能、ドキュメントエディタ、ブログサービスをネットを通じていつでも、どこでも利用可能だ。google はこうしたアプリケーションのサービス群を「クラウド・コンピューティング」と説明している。

アップルと google が「クラウド」という言葉で説明するこれらのサービスは、共通した 1 つの特徴がある。その特徴とは、ユーザーにサービスを使うためのテクニックをほとんど要求しないことだ。過去、ネット上のストレージにデータをアップロードするためには FTP ソフトの利用法やインターネットの接続など複数の手続きが必要だった。こうした手続きがサービスのなかに取り込まれ、ユーザーからは見えなくなった。つまり、クラウドは多くの IT 技術をサービスの中に取り込んで、コモディティ化するキーワードなのである。こうしたサービスの流れのなかでユーザーは、アプリケーションがデータを生成する、という認識すら持たなくなるだろう。

そして現在、携帯端末がユーザーに IT 技術を届ける機器の 1 つとして高い地位を占めつつある。携帯端末は PC 以上に属人機として人を支えるターミナルの役割を担うと期待されている。

しかしながら現在の携帯端末は、IT 技術のコモディティ化に耐える操作性を持っていないのではないだろうか。本稿ではこの仮説を入り口に 1 つの可能性を検討する。

1.2 本稿の位置付け

本稿は UI のコンセプトを解説することが目的である。このため技術的な可能性の検証を含まない。また、UI コンセプトは特定のプラットフォームに依存する目的で作られていないが、適応する機器の範囲はいわゆる「スマートフォン」に限られる。ここで言うスマートフォンとは、一般的な携帯電話よりも処理能力が高く、パケット通信以外の通信方式が利用可能な携帯端末という位置付けである。

また、UI はスマートフォンを通したアプリケーションの活用を絞って検討した。通話や文字入力など機器の操作全般に対する内容はここでは触れていない。

2. GUI

2.1 GUIの要素

クラウド時代のUIデザインについてのコンセプトを述べる前に、現在広く受け入れられているPCのGUIの要素について簡単に確認したい。

1980年代パロアルト研究所で芽を出し、Apple Lisaが搭載して商用となったGUIによって、コンピュータと人とのやり取りは言語による命令から視覚情報に対するフィジカルな働きかけに変わった。

働きかけの対象は「アイコン」だ。アイコンのおかげでユーザーはアプリケーションを視覚的に認識し、ファイルの整理ができる。アイコンとして認識可能なファイルやフォルダ（ディレクトリ）の中身は「クローム」（ウィンドウ）によって「開く」という感覚を伴いつつ確認できるようになっている。さらに「メニュー」という要素からシステムに働きかけることも可能だ。アイコンとクローム、メニューの3大要素がGUIの操作の基本と整理できる。

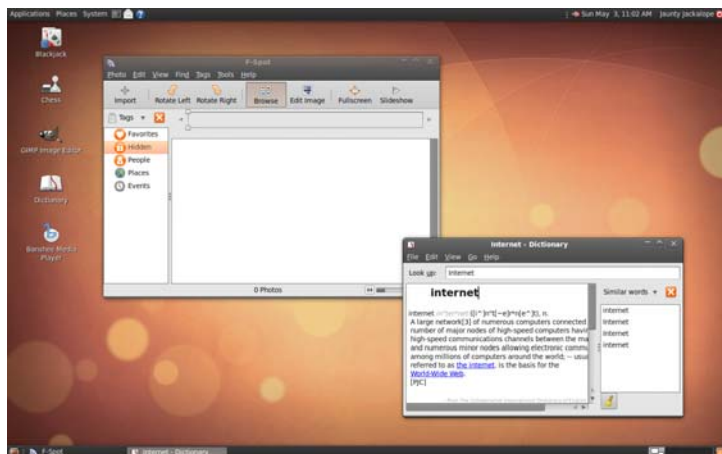


図 1 典型的な GUI の例 (Ubuntu 9.04.のスクリーンショット)

2.2 デスクトップ

GUI デザインを支えるメタファーはデスクトップ（机の上）という要素の呼び名が

示すように作業机が源だ。GUIの3大要素はすべてこの上で展開する。デスクトップ上のアイコンをクリックするとクロームが開く一連の動作は、机の上のファイルフォルダから書類を取り出す動作と違和感なくシンクロして馴染みやすい。

またデスクトップはアプリケーションを部品（ウィジェット）として扱うためにも重要な概念だ。PCの入門者が複数のアプリケーション間でファイルを受け渡すといった使いかたをするために、デスクトップというベース（基地）は欠かせない。ちなみに google は自社製のブラウザを「google クローム」と名付け、この操作の関係性における OS の存在をクラウドに置き換えている。

ところでデスクトップという操作性を十分に活用するには、ある前提をクリアする必要がある。それは表示領域を十分確保することだ。PCの性能が向上するにつれ、大きな表示解像度が求められてきた流れを見ると理解しやすい。大きい机ほど使いやすいように、その表示解像度の高さが PC の使い勝手を左右する。つまりデスクトップシステムは、広い表示領域限定の GUI なのである。それは GUI が机をメタファーとして作られた以上当然のなりゆきだ。そして広い表示領域を持つ機器は物理的なサイズを伴うことから、気軽に外に持ち出す事が難しい。この点が、携帯端末に対しデスクトップに変わる GUI のメタファーの検討が必要な理由である。

2.3 スクリーン

では、携帯端末のうち高い処理能力を持つスマートフォンは機器に対しどのような概念を与えているのだろうか。

スマートフォンの初期画面は各社共通で「home screen」と呼ばれている。iPhone や Android 携帯の場合、home screen は PC のデスクトップにきわめて近い操作性を持つ。横長の作業領域を持つが、可視領域がモバイルサイズに限定されているというイメージだ。不可視の領域はスクリーンを横方向に指でこすり、スクロールさせることで可能となる。アプリケーションの起動は home screen からアイコンをタップするか、home screen を長押しして表示するコンテキストメニューから行う。



図 2 エミュレーターで機動した Android 携帯の home screen



図 3 iPhone 3G の home screen (Apple Inc.)

一方 Windows Mobile や Symbian OS 端末の場合、一般的な携帯電話の操作感を踏襲し、home screen には機能をリスト化したデザインを採用している。アイコンよりも文字によるメニューを強調したデザインだ。アプリケーションを起動するシーンでは画面がアイコン一覧に切り替わるが、アプリケーションはデスクトップ型 GUI 同様の分類がなされている。唯一 NOKIA が N97 端末で home screen のパーソナライズ機能を搭載したが、よく使う項目以外のアプリケーションを固定的な分類によるリストから探して起動するといった使い方は、シングルファインダー時代の PC の操作感によく似ている。

これら既存のスマートフォンの操作感をアプリケーションの活用に絞って整理すると以下ようになる。

まず、ユーザーはアプリケーションを視覚的に選ぶ。次にアプリケーションをタップ

して起動し、メニューから目的の機能にアクセスする。処理を終えたら操作を終了する。こうした一連の操作感は、スマートフォンでアプリケーションを活用するシナリオに照らし合わせると使いにくい面が見えてくる。

3. コンセプトモデルの提案

3.1 シナリオ

では、スマートフォンを使って実現できると考えられる一連の操作を、「上野動物園に行く」というシナリオに沿って整理する。

1. オンラインドキュメントの閲覧：趣味で編集した動植物データのチェック
2. スナップ&シェアアプリの活用：動物の写真撮影&保存
3. バス時刻表ガジェット：バス時刻表の確認
4. コンタクトリストをチェックして電話で帰宅を知らせる

上記のシナリオを既存のスマートフォンで行う場合、次のような手続きが想定できる。

1. オンラインドキュメントの閲覧
 - home screen からドキュメント編集アプリケーションを起動
 - アプリケーションの操作画面からファイルの保存先をブラウズ
 - 目的のファイル（動植物データ）を開く
 - データを編集して保存
2. スナップ&シェアアプリの活用
 - カメラアプリを起動
 - 静止画モードで撮影
 - アップロード画面に遷移
 - アップロード画面からカテゴリを探すか、新しいカテゴリを作成する
 - 撮影データを動物の写真カテゴリにアップロード（保存）
3. バス時刻表ガジェット
 - バス時刻表サービスにアクセス
 - 現在地近くの目的地行き路線を特定
 - 現在地近くの停留所を特定
 - 時刻を確認
4. コンタクトリストをチェック

- コンタクトリストアプリ起動
- 全てのデータから目的の連絡先を探す
- 連絡先の特定
- 発信

ユーザーがそれぞれの目的を果たすまでの手続きは、モードの切り替えやデータの格納場所探しといったデスクトップシステム特有の手続きが必ず存在する。それだけ操作は煩雑になり、複数のアプリケーションを使いこなすことが難しくなる。この問題を解決するためには、まずスマートフォンの中のアプリケーションやデータを（アプリケーション）や（ゲーム）といった固定の分類から解放する必要がある。具体的にはユーザーの行動とひも付けやすい新しい分類方法を与える手法が考えられる。

3.2 ユーザーの行動と必要な機能の関係

では、前述のシナリオに基づいて「わたし」という個人の行動全体で必要と思われる機能を整理する。行動と機能を整理するには、住所というすでに整理された概念を利用する。

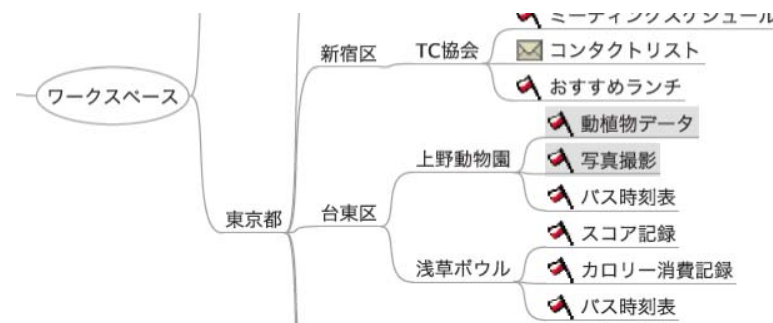


図 4 マインドマップ化した行動と必要な機能の関係。全体図は添付資料（map.pdf）に掲載

住所で表した行動範囲の下に、行動中必要か、あるいは望むアプリケーションの機能が整理できた。

3.3 プレイス、スポット、シーン

ここでユーザーの行動というコンテキストをアプリケーションへダイレクトに伝え

るため、既存の機器にはない（場所）というメタファーを採用する。メタファーから派生した用語の位置付けは以下の通り。

place

日本なら都道府県、米国なら州のような国土を区切る上位区分。国際的に行動するユーザーは設定を変更して国を **place** にしてもよい。

spot

place の下位区分。都の場合は区や市、府の場合は府とか市がこれにあたる。

scene

spot 内に存在する施設や通りを目的地として、また **spot** で行う行為を目的として設定できる。

これらの分類を Android 携帯を例に具体的な UI のアイデアに落とし込むと、以下のようなコンセプトデザインになる。



図 5 home screen からワークスペースに切り替え

home screen からワークスペースに切り替える際の手順の案。移動する人をかたどったアイコンをタップするとモードが切り替わる。

ワークスペースモードはリストでアクセスする体裁と、位置情報のうえにアプリケーションの情報を掲載する 2 パターンのモードが想定できる。コンセプトでは、趣旨をより分かりやすくするためリストのパターンに絞って考案した。

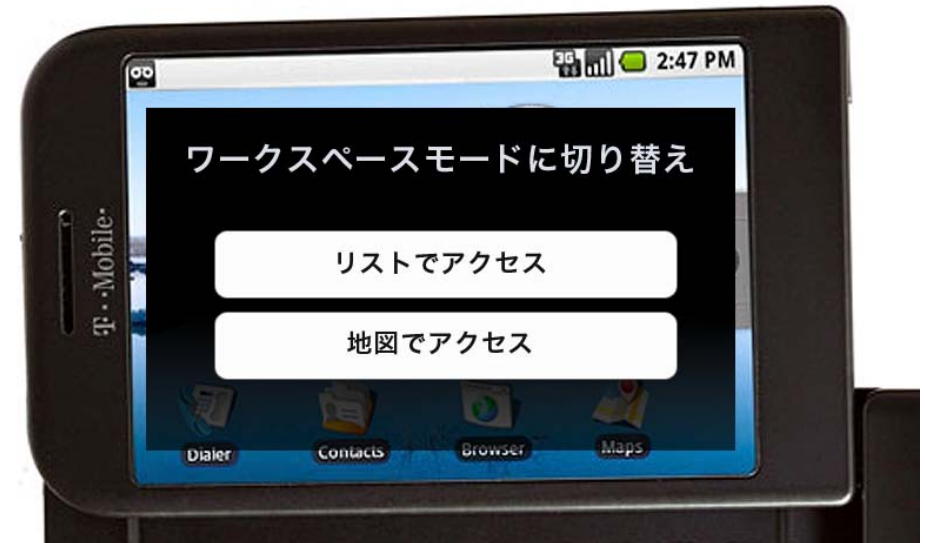


図 6 ワークスペースモードの選択肢



図 7 プレイスを選択

リストモードからプレイスを選択。プレイスの登録もこの画面で行うことができる。プレイスのスクリーンでスポットを選択。スポットはさらにシーンへと分岐する。



図 8 プレイスからスポットを選択

スポットから選んだシーンを表示。ランチャーからツール群にアクセスする。シーンのスクリーンには、シーンに沿ったおすすめのアプリケーションを確認するレコメン機能も搭載する。



図 9 プレイスを選択

目的のツールを起動すると、ユーザーのコンテキストに同期して適切なデータを表示したり、特定のモードに設定した状態を確認できる

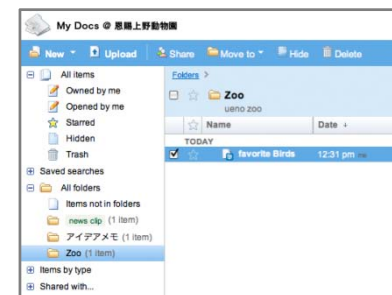


図 10 プレイスを選[a]

本コンセプトによってアプリケーションの各機能がユーザーのコンテキストにひも付けできれば、ユーザーはアプリケーションという個々の部品に階層的な操作を伴ってアクセスする必要がなくなる。さらに必要な機能だけをシーンに応じて活用するという使い方が可能になる。

ワークスペースモードで起動した場合に想定できる手続きの例は以下のとおり。

1. プレイスからスポットに移動、シーンを選択
 - オンラインドキュメントの閲覧
 - 目的のファイル（動植物データ）が開かれている
 - データを編集してシーンにひもづいた保存先にアップロード（保存）
2. スナップ&シェアアプリの活用
 - 静止画モードで撮影
 - アップロード画面に遷移
 - 撮影データを動物の写真カテゴリにアップロード（保存）
3. バス時刻表ガジェット
 - バス時刻表サービスにアクセス
 - 特定済みの現在地近くの停留所時刻を確認
4. コンタクトリストをチェック
 - コンタクトリストアプリ起動
 - シーン特定の連絡先にアクセス
5. 発信

アプリケーションの機能そのものにタグ付けするという考え方は、デジタルコンテンツにおけるフォークソノミーに近い。ただし、本コンセプトの場合はユーザーが言葉で自由にタグを与える方法でなく、住所や施設といった固定的な位置情報をタグとするため、分類の規則作りに関する裁量権をユーザーに与えない点で異なる。

本コンセプトはシーンに合ったスマートフォンの使い方がリテラシーなしでできるという利点のほかに、アプリケーション市場の活性化にも役立つ可能性を持っている。シーンごとのアプリケーションのレコメンド機能が実現すれば、現在 iPhone 向けアプリケーションだけでもマーケットプレイスに存在するアプリケーションの総数が 1 万 5000 を超えるという飽和状態の市場において、アプリケーションを選ぶという行動がしやすくなる。一般のユーザーにとってアプリケーションのマーケットに掲載のアプリケーション選ぶ方法は新着と人気ランキングリストを見ることに限られている。こうした制限は有料アプリケーションマーケット伸び悩みの要因にもなっている。ユーザーの行動がつくるコンテキストに操作性を添わせることによって、これらの問題が打開できるはずだ。また、友人とワークスペースを共有するような SNS 機能が実現できれば、行動的な友人の行動パターンを手本にスマートフォンの活用方法が広がる可能性も出てくる。

4. 実現するには

4.1 GPS との連携

コンセプトデザインは、アプリケーションランチャーのようにユーザーが事前にアプリケーションを整理する操作を前提としている。この事前準備の煩雑さはアプリケーションの利用履歴と GPS を連動させることで解消できるという考え方もできる。ただし、現在の GPS はビルや建物の中など遮蔽物の影響を受けやすく、リアルな行動をトレースしにくいという問題点もある。この問題が解消されないかぎり、地図情報を確認しながらユーザーが能動的に位置と機能をひも付けする事前準備は必要ではないだろうか。

4.2 時刻のタグ付け

コンセプトではシーンを位置情報の一種として扱っているが、時刻をシーンとする考え方もある。アプリケーション内の任意の機能を使った時刻をタグとして記憶させれば、生活パターンに応じた機能のレコメンドを端末側から受ける使い方が可能だ。朝、昼、夜、という時間をアプリケーション自体が知覚することで、時間軸上のニーズを把握し「シーン」をアプリケーション側が判断するようなくみを作れるかもしれない。

4.3 問題点

コンセプトモデルには以下のような問題点も指摘できる。

UI のアプリケーションレコメンド機能には、レコメンドをキャンセルする設定など、経済的に自立していないユーザーが無自覚に活用することを防ぐ機能が不可欠だ。

また、固定的な位置情報にアプリケーションの機能をひも付けすること自体、無理があるケースも想定できる。たとえば上野動物園でなく多摩動物公園に行くというケースを想定すると、その目的は似通ってくるため必要な操作もほぼ重複する事が予想される。コンセプトのブラッシュアップには、シーンごとに共通する部分をどのように扱うのかという問題の検討が不可欠である。

a) 図 10 は既存の google Docs の UI に基づいて作成したイメージである。現時点で google Docs モバイル版にドキュメントの編集機能はない

参考文献

- 1) Android Market
<http://www.android.com/market/>
- 2) Apple App Store
<http://www.apple.com/jp/iphone/appstore/>
- 3) Developer Android.com
<http://developer.android.com/>

付録

マインドマップ化した行動と必要な機能の関係 (map.psd)

謝辞 日本 Android の会 / 日本 Java ユーザーグループ

著者紹介

矢野りん

フリーライター・デザイナー

日本 Android の会 会員。

テクニカルコミュニケーター協会 Web ワーキンググループメンバー。

