

Technology Reports

ユーザ設定によりサービスの組合せが可能なサービス連携システム「BLOCCO」

多様化するユーザニーズに対応するため、カスタマイズ可能なコンテキストウェアネスやマッシュアップを実現するさまざまなサービスが存在するが、ユーザにとって操作が複雑すぎる場合や、そのままでは、ユーザが実現したいことに制約を受ける場合が多い。そこで、アプリケーションをブロックのように組み替えるだけで、擬似的なプログラミングを行うことが可能なサービス連携システムBLOCCO^{*1}を開発した。なお、BLOCCOは株式会社GClueと共同開発した。

サービス&ソリューション
開発部

はぎの ひろあき ふじい くにひろ
萩野 浩明 藤井 邦浩
むらかみ じゅんこ はら みらい
村上 純子 原 未来

1. まえがき

多様化するユーザニーズに対応するため、ユーザが機能をカスタマイズできるサービスが多数存在する。特に、サービスの振舞いを単一にカスタマイズするのではなく、状況に応じて異なる振舞いを設定できる、コンテキストウェアサービス[1]が増加している[2][3]。また、カスタマイズの手段として、サービスとサービスを組み合わせてマッシュアップ^{*2}[4]できるサービスも出現している[5]~[8]。しかし、これらのサービスは、ユーザにとって操作が複雑すぎる場合や、そのままでは、ユーザが実現したいことに制約を受ける場合が多い。

そこで、ユーザがアプリケーションをブロックのように組み替えるだ

けで、擬似的なプログラミングを行うことが可能なBLOCCO[9]を開発した。

BLOCCOはAndroidTM*3 OS[10]上で動作するAndroidアプリケーションであり、ドコモと株式会社GClueとが共同で開発した。ユーザはBLOCCOを利用することで、複数のAndroidアプリケーションを組み合わせて、あるアプリケーションにおいて発生・検出した事象をトリガとするサービス実行や、アプリケーション間でパラメータを受け渡し、そのパラメータを処理するようなアプリケーションの動作を、自由に設定できる。BLOCCOは、ユーザ設定に従って他のアプリケーションの状態を監視し、アプリケーション間のパラメータの仲介およびアプリケーションの自動実行を実現する。

本稿では、サービス連携システムBLOCCOの概要と、関連技術との比較を交えたBLOCCOの有効性について解説する。

2. サービス連携システム「BLOCCO」

BLOCCOはAndroid OS上で動作するAndroidアプリケーションであり、Androidマーケットからダウンロード可能である。

2.1 BLOCCOの特徴

BLOCCOでは、複数のAndroidアプリケーションを組み合わせて、プランと呼ぶサービス実行シナリオを設定できる。BLOCCOおよびBLOCCOで設定できるプランは、次の2つの特徴をもつ。

*1 BLOCCO：Bloccoは(株)GClueの商標。
*2 マッシュアップ：複数の異なるサービスやコンテンツを組み合わせて、1つのサービスを作成、提供すること。
*3 AndroidTM：米国Google, Inc.が提唱する携帯端末を主なターゲットとしたオープ

ンソースプラットフォーム。AndroidTMは、米国Google, Inc.の商標または登録商標。

(1) イベント駆動型プログラミング^{*4}

「〇〇が発生したら」のように、あらかじめ定められた条件を判定し、その条件が満たされた場合にサービスを実行する。さらに、BLOCCOはサービスを実行するだけでなく、動作条件と実行動作のセットからなるプランを自由に設定できる機能を持ち、ユーザは、動作条件と実行動作のそれぞれに、さまざまなAndroidアプリケーションを割り当てることができる。

(2) サービス間パラメータ連携

実行動作に割り当てられたアプリケーションに対して、他のアプリケーションからパラメータを引き渡すことができる。また、ユーザはそのような動作をプランに設定できる。

BLOCCOの主要な画面例を図1に示す。BLOCCOでは、プラン

の動作条件や実行動作に割り当てられるアプリケーション、および実行動作に割り当てられたアプリケーションにパラメータを提供するアプリケーションを、プラグインと総称する。さらに、動作条件に割り当てられるプラグインをイベントプラグイン、実行動作に割り当てられるプラグインをアクションプラグイン、アクションプラグインにパラメータを提供するアプリケーションをインプットプラグインと呼ぶ。

BLOCCOと各種プラグインとの間の通信は、Android OSが提供するintent^{*5}上にプロトコルを規定することで実現している。

次に、BLOCCOを用いることで可能となるプラン例を示す。

プラン例1：

スケジューラに件名「飲み会」、時間「19時から22時」、

場所「六本木」と登録しておく、1時間前に今いる場所から六本木に19時に到着できるような電車の経路検索を自動で実行してくれる。

プラン例2：

家に帰り、手書きジェスチャーアプリケーション^{*6}を起動して、画面に「〇」を描くと自動的にTwitter^{*7}に「帰宅なう」とつぶやいてくれる。

プラン例1は、iコンシェルで提供されている機能とほぼ同等の機能である。プラン例1におけるプラグインの割当てを図2に示す。スケジューラアプリケーションがイベントプラグインに割り当てられ、「予定の1時間前になったら」という条件が設定されている。アクションプラグインには、経路検索アプリケーションが割り当てられてい



図1 BLOCCOの画面例

*4 イベント駆動型プログラミング：なんらかの事象が発生、検出されたときに、それを契機として実行されるプログラムを生成すること。BLOCCOにおいては、ユーザがサービス実行シナリオを作成することを指す。

*5 intent：Android OSが提供するプログラム間のパラメータ交換手段。アプリケーション内のコンポーネント間でのやりとりや、アプリケーションをまたがるやりとりに利用される。

*6 手書きジェスチャーアプリケーション：

手書き入力認識アプリケーション。本稿では、ユーザがあらかじめ「〇」などの簡単な図形を手書きで登録でき、ユーザによる手書き入力が、あらかじめ登録されたどの図形に該当するかを判別するアプリケーションのことを指す。

る。経路検索アプリケーションが必要とするパラメータのうち、「到着駅」と「時刻」については、イベントプラグインであるスケジューラアプリケーションが、インプットプラグインも兼ねることでパラメータを提供している。また、「出発駅」のパラメータ提供には、GPSアプリケーションが割り当てられている。このように、BLOCCOを利用すると、なんらかの事象が発生したときに、あらかじめプランにアクションプラグインとして設定しておいたアプリケーションを自動的に実行することができる。同様に、例えば友人からイベントのお誘いメールが届いたときに、自動的に経路検索を実行するなど、プラグインとなるアプリケーションを組み替えるだけで、さまざまな応用を実現できる。

プラン例2におけるプラグインの割当てを図3に示す。プラン例2では、手書きジェスチャーアプリケーションがイベントプラグイン、Twitterクライアントアプリケーションがアクションプラグインに該当する。この例は、BLOCCOがアプリケーションの自動実行だけでなく、半自動実行やユーザ操作のショートカットも可能であることを示している。

BLOCCOでは、プランの実行トリガをイベントプラグインが発行するものとして定義している。これにより、Android端末などの状態を監視するアプリケーションがイベントプラグインに設定されてい

る場合は、アプリケーションの自動実行が可能であり、ユーザの操作によって実行トリガを発行するアプリケーションがイベントプラグインに設定されている場合は、半自動実行やユーザ操作のショ

ットカットとして利用できる。

このように、BLOCCOは、ユーザがAndroid端末にインストールしているアプリケーションを柔軟に組み合わせて、さまざまなプランを作成できる環境を提供する。特

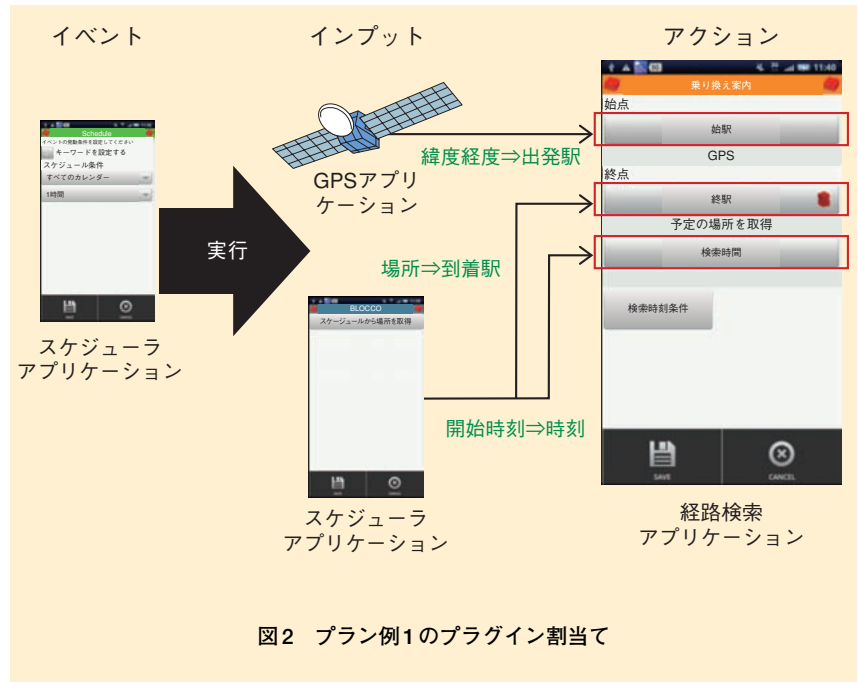


図2 プラン例1のプラグイン割当て

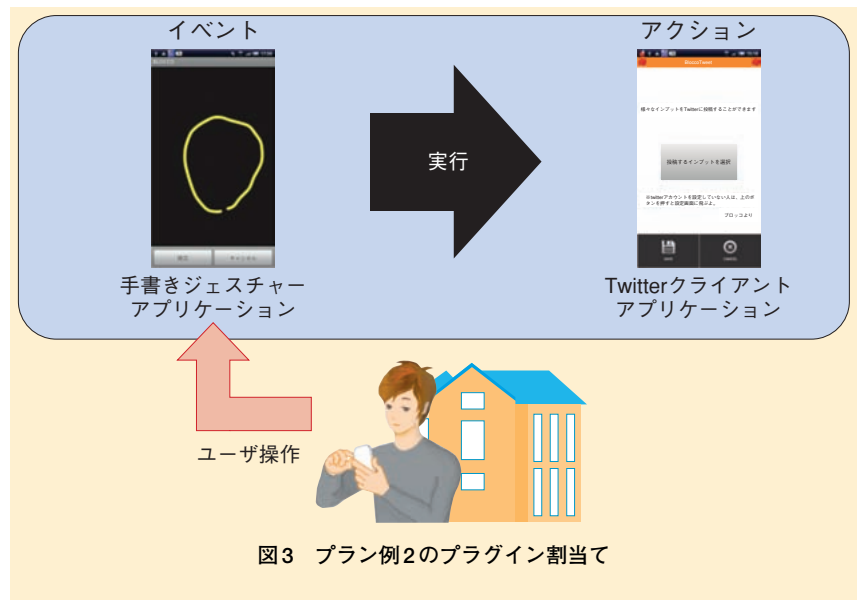


図3 プラン例2のプラグイン割当て

*7 Twitter：アメリカ合衆国また他国々におけるTwitter, Inc.の登録商標。

に、ユーザが日常的に行う動作を、自動的もしくは半自動的に行いたいというニーズとの親和性が高いと考えられる。

2.2 ユーザ操作の支援

一般に、BLOCCOのプランのようなシナリオを設定できるサービスは、PC向けに提供されているものが多く、パラメータの流れやシナリオを構成するコンポーネントの関係をグラフィカルに表示するものが多い[10]。しかし、BLOCCOはAndroid端末上でのユーザ操作を前提としているため、画面に表示できる情報量が限定される。BLOCCOでは、特にユーザがプランを設定する際、画面に表示され

る限られた情報に基づいて、多数のアプリケーションから必要なプラグインを選択し、それらの間のパラメータの流れを意識しながら設定しなければならない。そこで、BLOCCOは次に示す方法で、プラン設定時のユーザを支援する。ユーザが2.1節のプラン例1を設定する際の画面遷移を図4に示す。

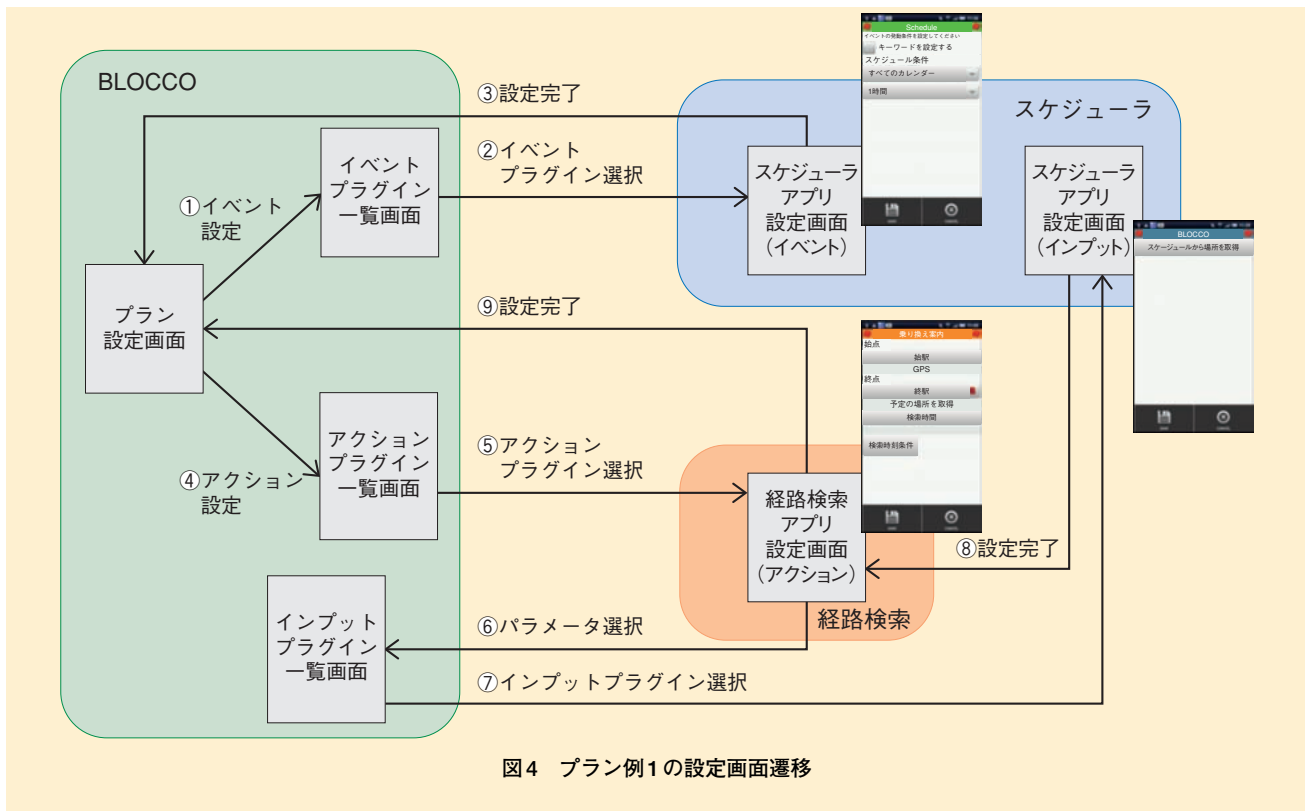
(1)プラグイン一覧の提示

BLOCCOは、プラグインとなるアプリケーションがインストールされていることを、自動的に検出できる。これによって、ユーザがプランを設定する際に、プラグインの一覧を提示することができる。プラグインの一覧は次のようなタイミングでユーザに提示される。

- ・プランにイベントプラグインやアクションプラグインを割り当てるとき
- ・アクションプラグインのパラメータに対してインプットプラグインを割り当てるとき

(2)パラメータ受渡し設定

BLOCCOの特長の1つとして、サービス連携の紐付け方が「アプリケーションtoアプリケーション」ではなく、「パラメータtoパラメータ」であることが挙げられる。例えば、2.1節のプラン例1では、経路検索アプリケーションの「出発駅」「到着駅」「時刻」のそれぞれのパラメータに対して、GPSアプリケーション、スケジューラアプリケーションの「緯度経度」「場所」「開始



時刻」が割り当てられている。そこでBLOCCOでは、ユーザがアクションプラグインの画面に表示されているパラメータから、インプットプラグインを設定できる画面遷移を採用した。ユーザがアクションプラグインのパラメータを選択すると、前述のインプットプラグイン一覧が表示される。

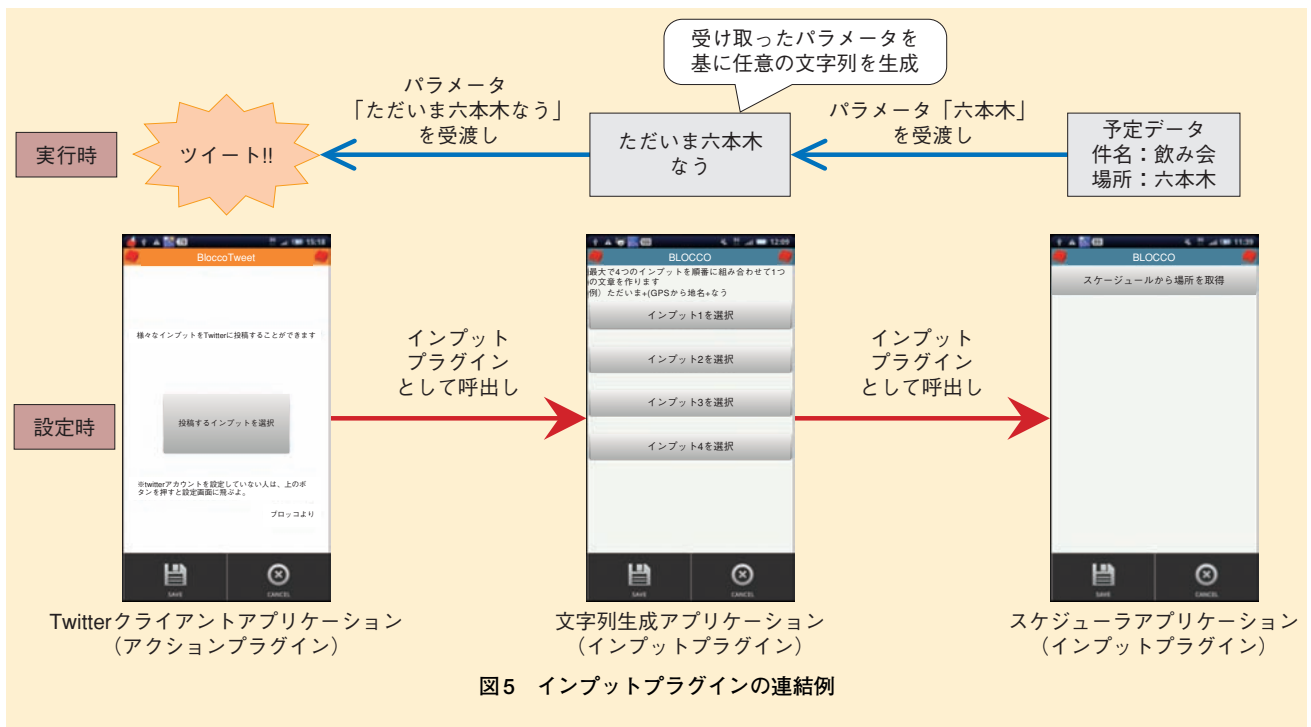
また、BLOCCOでは、インプットプラグインを多重設定することができる(図5)。本図の例では、スケジューラアプリケーションから場所情報「六本木」を取得し、その情報を使って任意の文字列「ただいま六本木なう」を生成して、Twitterクライアントアプリケーションに渡している。この場合、他のインプットプラグインからパラメータを受け取り、文字列を生成して、

他のプラグインへその文字列をパラメータとして提供するインプットプラグインが存在する。つまり、アクションプラグインであるTwitterクライアントに対して、文字列生成アプリケーションとスケジューラアプリケーションが、2重に連結されたインプットプラグインとして設定されている。このような場合も、Twitterクライアントアプリケーションのパラメータに対するインプットプラグインとして、文字列生成アプリケーションを設定し、文字列生成アプリケーションのパラメータに対するインプットプラグインとして、スケジューラアプリケーションを設定する。この流れのように、複雑なプラン設定をAndroid端末の限定された画面領域で行うことが可能となった。

3. BLOCCOに関する考察

3.1 サービス連携のコンセプト

BLOCCOは、ドコモが提案するサービス連携フレームワークに基づいて、設計・開発を行った。サービス連携フレームワークは、ユーザカスタマイズ、マッシュアップおよびコンテキストウェアネス^{*8}という3つの概念もしくは技術の組合せとして考えている。すなわち、サービス連携フレームワークとは、開発者ではなくユーザが、自由にサービスとサービスを組み合わせること(ユーザカスタマイズ)で、サービス間でのパラメータの受渡しと、それに伴うサービスの振舞いの制御(マッシュアップ・コンテキストウェアネス)を実現可能な環境や技術を指す。



*8 コンテキストウェアネス：ユーザやサービスに関する情報に応じて、動作を変更、決定するサービスをコンテキストウェアサービスといい、そのような振舞いや概念のことをコンテキストウェアネスと呼ぶ。

3.2 関連技術

ユーザによるサービスのマッシュアップを可能とする仕組みとして、Plagger[5]やIntelligentPad^{*9}[6]、Yahoo! Pipes^{*10}[7]、Accelerators[8]が存在する。また、ユーザによってカスタマイズ可能なコンテキストウェアサービスとして、Toggle Setting[3]、Locale[2]が挙げられる。これらの技術の比較を表1に示す。

この中で、特にLocaleはBLOCCOのコンセプトに近い。LocaleはAndroidアプリケーションであり、Android端末の設定を自動的に変更することができる。Localeの特徴は、設定の自由度と自動実行である。例えば、ユーザは、自分の位置がある範囲に入ったときに、Android端末のWi-Fi^{®*11}のON/OFF

を自動的に切り替えることが可能である。ユーザはこのような設定を、BLOCCOと同じように、自由に組み替えられる。また、設定を変更するトリガとなる「Event」や設定対象である「Setting」として、他のAndroidアプリケーションをプラグインできるインタフェースを有するため、端末設定に限らず、さまざまなサービスの振舞いを変更することができる。この概念は、複数サービスの組合せによるイベント駆動型プログラミングを可能とするものである。しかし、あるサービスの状態を他のサービスの実行トリガとして利用はできるが、BLOCCOのようにサービス間でパラメータを引き継いで利用することはできない。

3.3 BLOCCOの利便性

3.2節でさまざまな関連技術について説明したとおり、これまでユーザカスタマイズが可能なコンテキストウェアサービス、もしくはマッシュアップサービスは存在するが、サービスの組合せ（マッシュアップ）をユーザがカスタマイズすることでコンテキストウェアサービスを実現することはできなかった。その点において、BLOCCOはLocaleなどの関連技術と比較して、利便性・柔軟性が高いといえる。

また、近年、スマートフォン上では、本来web上で提供されているサービスでも、それに対応するアプリケーションが開発/提供されている場合が多い。したがってアプリケーション間の連携を可能とすること

表1 BLOCCOと関連技術との比較

	コンテキストウェアネス	マッシュアップ	簡易性	柔軟性	導入容易性	特徴
Plagger	△	○	×	◎	◎	webサービスとwebサービスをつなげる仕組み。設定にはperlの知識が必要
IntelligentPad	×	◎	○	△	×	ユーザがwebコンテンツを切貼りして、マッシュアップできるサービス。コンテンツが独自仕様に対応する必要がある
Yahoo! Pipes	△	○	△	△	◎	RSSを取得して、さまざまな処理が可能なサービス。GUIにより設定が容易
Accelerators	×	○	△	×	◎	webコンテンツからwebコンテンツへ単一のパラメータを引き継ぐことができるサービス。設定には知識が必要
Toggle Setting	◎	×	◎	×	—	ユーザがシチュエーションごとに端末の設定をセット化し、ワンタッチで変更可能なアプリケーション
Locale	◎	○	○	△	○	ユーザが端末設定の変更を、条件と動作のセットで詳細に設定可能なアプリケーション。他のアプリケーションをプラグイン可能
BLOCCO	◎	◎	△	○	△	アプリケーションを組み合わせて、条件と動作のセットで実行できるサービス。さらに実行時のアプリケーション間のパラメータ受渡しも設定できる

perl：インタプリタ型のプログラミング言語。一般に、文字列処理に適しているといわれる。

RSS (RDF Site Summary)：一般に、webページの更新情報を通知するために使われる、webページの要約などのメタデータ記述方式。

XML (eXtensible Markup Language)：文書を構造化するためのマークアップ言語。ユーザが独自タグを定義可能なため、拡張性が高い。

*9 IntelligentPad：財団法人さっぽろ産業振興財団の登録商標。

*10 Yahoo! Pipes：Yahoo!およびYahoo! Pipesは、Yahoo! Inc.の商標または登録商標。

*11 Wi-Fi[®]：Wi-Fi Allianceの登録商標。

は、最小限の開発コストで世の中の多くのコンテンツやサービスを網羅できる可能性が高いと考えられる。そこで、従来のマッシュアップがwebサービス、コンテンツに対するものであったのに対して、BLOCCOはAndroidアプリケーションを対象とした。BLOCCO自身もAndroidアプリケーションであるため、BLOCCOが影響をおよぼす範囲は、一見各ユーザのAndroid端末の中に閉じているように感じられるが、BLOCCOはあらゆるサービスやコンテンツを組み合わせて、サービス連携というコンセプトを実現できる、新しいプラットフォームとなり得るものである。

3.4 BLOCCOの課題

3.3節で述べたように、BLOCCOはLocaleと比較して、サービス間のパラメータ引継ぎが可能である。これにより、ユーザはより柔軟なシナリオを作成できる。しかし、一般に機能や柔軟性の高度化は、ユーザ操作の複雑化とトレードオフの関係にある。例えば、Toggle SettingとLocaleを比較すると、双方ともAndroid端末の設定を変更可能なアプリケーションであるが、Localeの方がより柔軟なシナリオを作成可能であり、それに伴って、ユーザに求める操作が必然的に複雑となっている。BLOCCOはLocaleと比較して、機能や柔軟性をさらに高度化するため、ユーザに要求する操作はより複雑なものになる。BLOCCOは、複雑な設定ができる

ユーザの使用を想定しているが、新しいプラットフォームとして展開するためには、操作や設定に慣れていないユーザにとっても利用しやすい環境を整備することが、今後の課題となる。

3.5 BLOCCOの拡張性

3.2節で紹介した、ユーザによるマッシュアップ技術の普及の背景には、web技術やwebコンテンツの汎用性の高さがある。汎用性の高い技術の集大成であるため、既存コンテンツのさまざまな流用が可能となり、その中に、ユーザによるマッシュアップがあるといえる。しかし、3.2節で解説したユーザによるマッシュアップ技術のうち、対応コンテンツに独自仕様への対応を要求するIntelligentPadは、コンセプトや技術の先進性・有用性と比較して、普及しているとはいいがたい。BLOCCOは汎用性の高いwebコンテンツではなく、アプリケーション

を対象としているため、既存のあらゆるアプリケーションがプラグインとなるのではなく、BLOCCOが規定するインターフェースを有するアプリケーションのみが、プラグインとして動作できる。そのため、対応アプリケーションを開発する際、通常アプリケーションを開発する場合と比較すると、BLOCCOに対応するための開発オーバーヘッドが発生することは避けられない。この不可避である開発オーバーヘッドを可能な限り小さくするため、BLOCCOプラグインのプロトコルインターフェースの仕様と、BLOCCOプラグインを開発するためのBLOCCO SDK (Software Development Kit) ^{*12}を、web上で一般に公開している[9]。SDKの画面を図6に示す。このSDKを利用すると、BLOCCOと通信するためのインターフェース部分のソースコードのひな形を自動生成できるため、プラグイン開発にかかるオーバーヘッドは大幅に小さくなる。

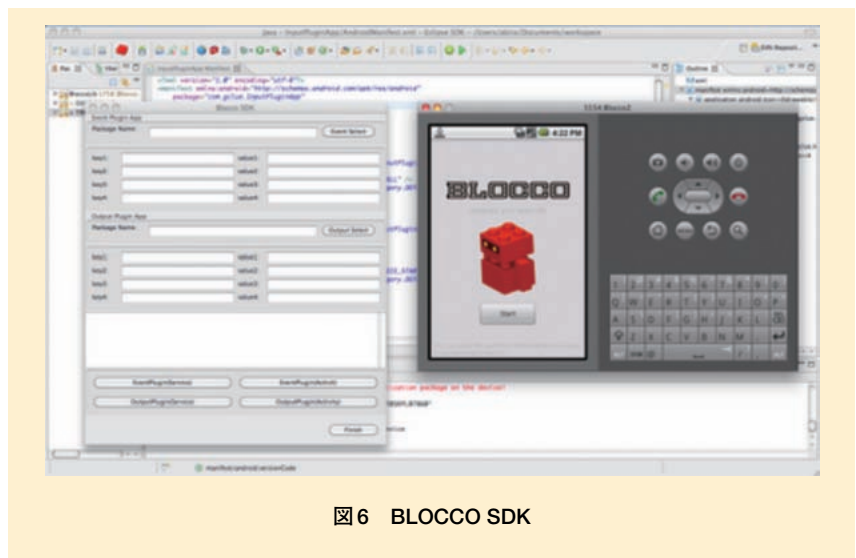


図6 BLOCCO SDK

*12 SDK：ソフトウェアを開発するためのツールもしくはツール群。

BLOCCOが本来もつ、独立するアプリケーションどうしの連携という概念に加え、プラグイン開発に必要な仕様や開発環境を公開することで、BLOCCOは高い拡張性を実現している。

4. あとがき

本稿では、サービス連携システム「BLOCCO」の概要を解説し、関連技術との比較を基にBLOCCOの利便性や拡張性について検証することで、BLOCCOのプラットフォーム

としての優位性を示した。今後は、あるユーザーが設定したプランを、他のユーザーと共有できる仕組みなどを導入するなど、設定が困難なユーザーでもBLOCCOを使用可能とする仕組みを検討する予定である。

文 献

- [1] A. K. Dey and G. D. Abowd : “Toward a Better Understanding of Context and Context-Awareness,” Proc. of the CHI2000, 2000.
- [2] “『Toggle settings』設定コントロールアプリ～こちらAndroid管制塔!～ |

andronavi (アンドロナビ) ; Toggle Setting,” Jan. 2010.

- [3] Locale for Android ホームページ.
- [4] T. O'Reilly : “What Is Web 2.0 - O'Reilly Media,” Sep. 2005.
- [5] Plagger ホームページ.
- [6] 田中 譲 : “ミームメディア・アーキテクチャ IntelligentPad とその応用,” 情報処理, Vol.38, No.9, pp.222-231, Mar.1997.
- [7] Yahoo! Inc. : “Pipes: Rewire the Web ; Yahoo! Pipes.”
- [8] Microsoft Corp. : “Internet Explorer 8 Readiness Toolkit ; Accelerators.”
- [9] Blocco ホームページ.
- [10] Android ホームページ.