# 絵文字パッケージ対応 IME 開発ガイド

Ver. 1.1.1

オムロン ソフトウェア株式会社

2012/4/10

Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 目次

*1. 本ドキュメントについて* 5

*2. 参照資料* 5

2.1. 参照ドキュメント 5

*3. 略語・記号の定義* 5

*4. 機能概要* 6

*5. 構成* 6

*6. 絵文字パッケージ* 7

6.1. 絵文字パッケージでできること 7

6.2. 絵文字パッケージの機能レベル 7

6.3. 絵文字パッケージのI/F 8

*7. Application Programming Interface* 9

7.1. 入力/表示シーケンス 9

*8. IME Interface* 12

8.1. キャリア絵文字入力I/F（A機能） 12

8.2. デコメ絵文字一覧入力I/F（B機能） 12

8.3. デコメ絵文字予測/変換I/F（C機能） 14

8.4. IMEへ提供する汎用インタフェース 15

*9. 【実装手順】アプリからの絵文字パッケージ利用手順* 17

9.1. 絵文字表示対応 【Level A/B/C】 17

9.1.1. 絵文字入力可能なエディタの生成 17

9.1.2. EmojiAssistに組込み、絵文字表示を可能にする 17

9.1.3. 文字列からデコメ絵文字を取得 17

*10. 【実装手順】IMEからの絵文字パッケージ利用手順* 19

10.1. IMEのキャリア絵文字一覧入力機能追加【Level A】 19

10.2. IMEのキャリア絵文字予測/変換入力対応【Level A】 19

10.3. IMEのデコメ絵文字一覧起動・入力の追加【Level B】 19

10.3.1. デコメ絵文字一覧の起動 19

Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

10.3.2. 入力されたデコメ絵文字の伝送 20

10.4. IMEのデコメ絵文字予測/変換入力対応【Level C】(option) 20

*11. 【実装手順】デコメ絵文字読み情報更新手順 21*

11.1. デコメ絵文字辞書情報更新の基本手順 21

11.2. シーケンス図 22

*12. 【実装手順】デコメ絵文字の一覧表示の更新 25*

12.1. 一覧表示アプリケーションの表示更新【Level C】 25

*13. 【実装手順】IMEに関する付加情報 26*

13.1. Android Framework全画面モード時の既存不具合対応 26

13.2. スリープ時の注意事項 26

13.2.1. 対応策（アクティビティアプリケーション向け） 26

13.2.2. 対応策（(サービスアプリケーション向け（IME を含む)) 27

*14. 【API】インタフェース仕様 28*

14.1. TextView/EditText（絵文字入力の有効化） 28

14.2. EmojiAssistクラス 29

14.2.1. インスタンス取得 29

14.2.2. TextView登録 30

14.2.3. TextView削除 31

14.2.4. TextView全削除 32

14.2.5. アニメーション開始 33

14.2.6. アニメーション停止 34

14.2.7. デコメ絵文字Span設定 35

14.2.8. 絵文字データ取得 36

14.2.9. ExtractEditText差し替え 37

14.3. DecoEmojiSpanクラス 38

14.3.1. URI取得 38

14.4. EmojiDrawableクラス 39

14.4.1. 絵文字判定 39

14.4.2. 絵文字判定(コード) 40

14.5. 絵文字一覧アプリDecoEmojiList 41

14.5.1. Intent起動 41

14.5.2. 返値取得 42

14.6. 絵文字マネージャDecoEmojiManager 43

14.6.1. 絵文字マネージャのインタフェースIDecoEmojiManagerのAPI 44

14.6.2. 利用クラス 48

14.6.3. 定数 49

Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

14.6.4. DecoEmojiManager発行インテント 50

*15. 【データ仕様】デコメ絵文字読み付加情報フォーマット 52*

Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 1. 本ドキュメントについて

本ドキュメントは、Android PF 上で絵文字の入力・表示を可能にするソリューション「絵文字パッケージ」と連携する機能を IME に組み込むためのガイド（手引書）です。

本ガイドに沿って作成された IME では、キャリア絵文字、デコメ絵文字の一覧選択入力および予測/変換入力と表示を行うことが可能となります。

# 2. 参照資料

## 2.1. 参照ドキュメント

# 3. 略語・記号の定義

| | |
|---|---|
| IMF | Input Method Framework |
| IME | Input Method Editor |
| 絵文字 | キャリア絵文字とデコメ絵文字の総称。 |
| キャリア絵文字 | 携帯電話事業者ごとに定められた、文字コードを持つ絵文字。 |
| デコメ絵文字 | 文章中にキャリア絵文字と同様に用いられる画像。<br>携帯電話事業者により様々な呼称があるが、本書では代表して「デコメ絵文字」と呼ぶ。<br>※「デコメ絵文字」は、株式会社 NTT ドコモの登録商標です。 |



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 4. 機能概要

絵文字パッケージは、Android プラットフォームに、携帯電話で普及している絵文字(キャリア絵文字およびデコメ絵文字)の入力と表示の機能を加えるソリューションです。標準的な Android プラットフォームにこのパッケージをインストールすることによって、アプリケーションに共通の絵文字機能を付け加えることができます。アプリケーションは、標準的なアプリケーションフレームワーク(IMF、TextView、WebView)を使い、絵文字の入出力を行うことができます。文字入力アプリ(IME)は、絵文字辞書機能を備えることによって、絵文字パッケージと連携して絵文字を入力できるようになります。

# 5. 構成

絵文字パッケージに IME を加えた絵文字関連のモジュール構成は図 1 絵文字関連モジュール構成（IME 関連のみ抽出）のようになります。

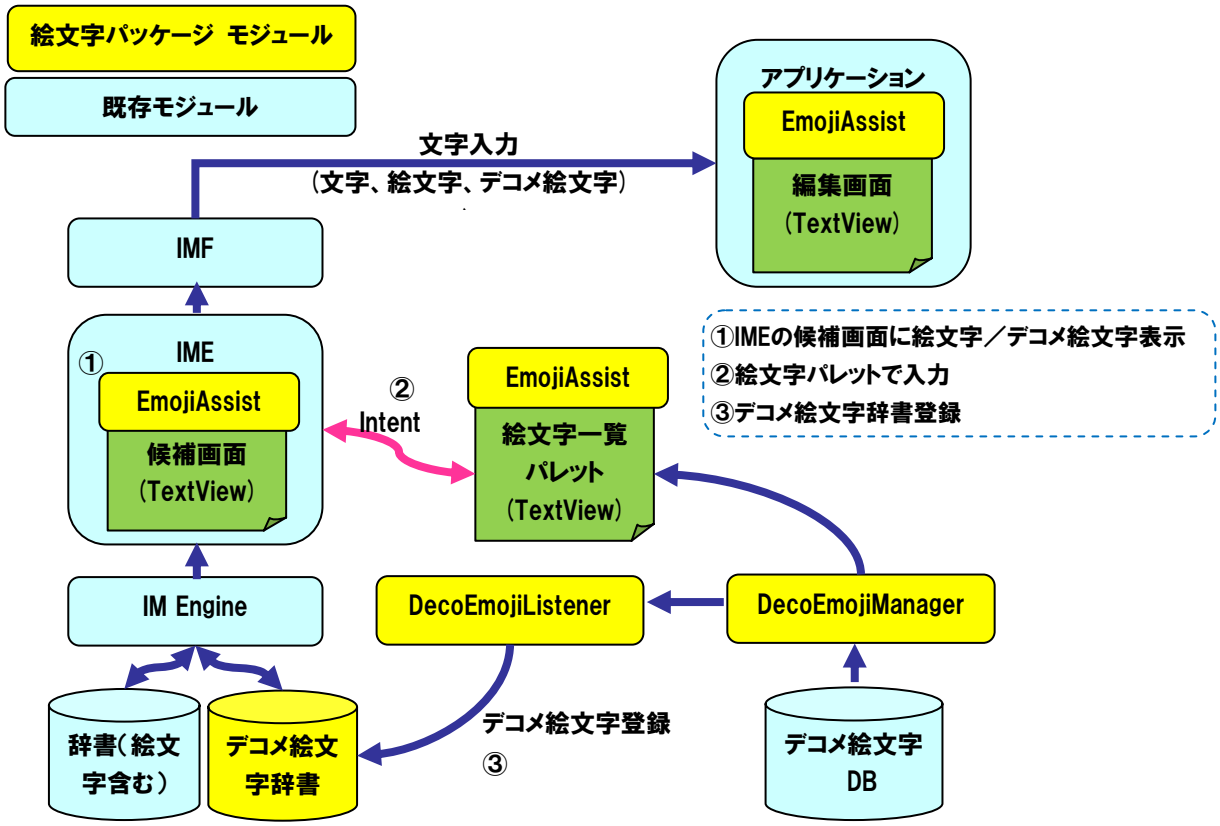



図 1　絵文字関連モジュール構成（IME 関連のみ抽出）



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 6. 絵文字パッケージ

Android 絵文字パッケージは、Android プラットフォームに絵文字（キャリア絵文字、デコメ絵文字）の入力および表示機能を加えるソリューションです。

## 6.1. 絵文字パッケージでできること

- 標準的な Android 端末は、絵文字パッケージをインストールすることによって、アプリケーションに共通の絵文字機能を付け加えることができます。
  - A) Android 端末の FW に絵文字機能を組み込んでいるか否かに関わらず、絵文字パッケージをインストールすることでアプリ間で共通の絵文字を利用することができます。
  - B) Android 端末にあらかじめインストールしたデコメ絵文字またはダウンロードしたデコメ絵文字を一覧表示し、アプリから入力することを可能にします。
  - C) デコメ絵文字に読みを付けることができ、IME の予測/変換機能を用いて入力することも可能になります。
- アプリケーションは、絵文字パッケージを利用する機能を組み込むことによって、標準的なアプリケーションフレームワーク （IMF、TextView、WebView）を使い、絵文字の入出力を行うことができます。
- ユーザは、通常文字、キャリア絵文字、デコメ絵文字を特別に意識せずにシームレスに利用して、豊かな表現を行うことができます。
- 絵文字パッケージは IM サービス（IME）と連携して、絵文字の入力を可能にします。絵文字パッケージが備える全ての機能を Android 端末で可能にするには、IME は絵文字の一覧選択機能と、絵文字辞書機能を備えることが必要です。

## 6.2. 絵文字パッケージの機能レベル

絵文字パッケージには次の 3 つの機能レベルがあります。

(レベル A)キャリア絵文字対応
- IM: キャリア絵文字の一覧および予測/変換入力
- アプリ: キャリア絵文字の表示

(レベル B)デコメ絵文字対応
- IM: デコメ絵文字の一覧からの入力
- アプリ: デコメ絵文字の表示

(レベル C)デコメ絵文字予測/変換入力
- IM: デコメ絵文字の予測/変換入力
- アプリ: デコメ絵文字の表示と追加



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 6.3. 絵文字パッケージの I/F

絵文字パッケージは、アプリケーション、絵文字データを格納する DB 、および IME との間の 3 つのインタフェースを持っています。機能レベルごとに必要となる I/F を下表に示します。

**表 1　絵文字パッケージ IF 一覧**

| 絵文字パッケージ<br>機能レベル | 1.アプリケーション<br>I/F | 2. DB I/F | 3. IME I/F |
|---|---|---|---|
| (A)キャリア絵文字<br>入力/表示 | ・絵文字有効フラグ<br>(allowEmoji) | ・絵文字フォント<br>データライブラリ | ・絵文字辞書 |
| (B)デコメ絵文字一覧<br>入力/表示 | ・デコメ絵文字<br>有効フラグ<br>(allowDecoEmoji)<br>・デコメ絵文字<br>ID(URI) | ・カテゴリ分類<br>(ディレクトリ) | ・一覧表示 |
| (C) デコメ絵文字変換<br>入力/表示 | ・(B に同じ) | ・デコメ絵文字読み<br>拡張フォーマット | ・デコメ絵文字辞書更新情報 |

IMEは、絵文字パッケージに対して絵文字を利用するアプリケーションという側面と、絵文字パッケージと連携してアプリケーションの絵文字入力を可能にするサービスという側面を持っています。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 7. Application Programming Interface

絵文字を扱うアプリケーションは、TextView を用いて、絵文字を含む文字列を入力および表示します。WebView では表示のみが可能であり、たとえば HTML メールの Preview 画面として用いられます。

絵文字は文字コードの問題などから、使えるアプリケーションを限る必要があります。このため、キャリア絵文字を扱えるかどうか、デコメ絵文字を扱えるかどうかのフラグ（TextView の属性）を API として以下のように定義しています。

・allowEmoji フラグ
　TextView において、キャリア絵文字の入力、表示を可能にする属性（Hidden API）
・allowDecoEmoji フラグ
　TextView において、デコメ絵文字の入力と表示を可能にする属性（本パッケージで定義した API）

・デコメ絵文字 ID
　アプリケーションは、入力/表示において通常文字と絵文字を区別する必要はありません。
　しかし、メールへの添付など、デコメ絵文字コンテンツファイルを意識して扱う場合があります。
　デコメ絵文字 ID はこれをファイルとして扱うための DB 内の ID(URI)です。

## 7.1. 入力/表示シーケンス

絵文字/デコメ絵文字の入力と表示を行うアプリは以下の流れで実行します。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

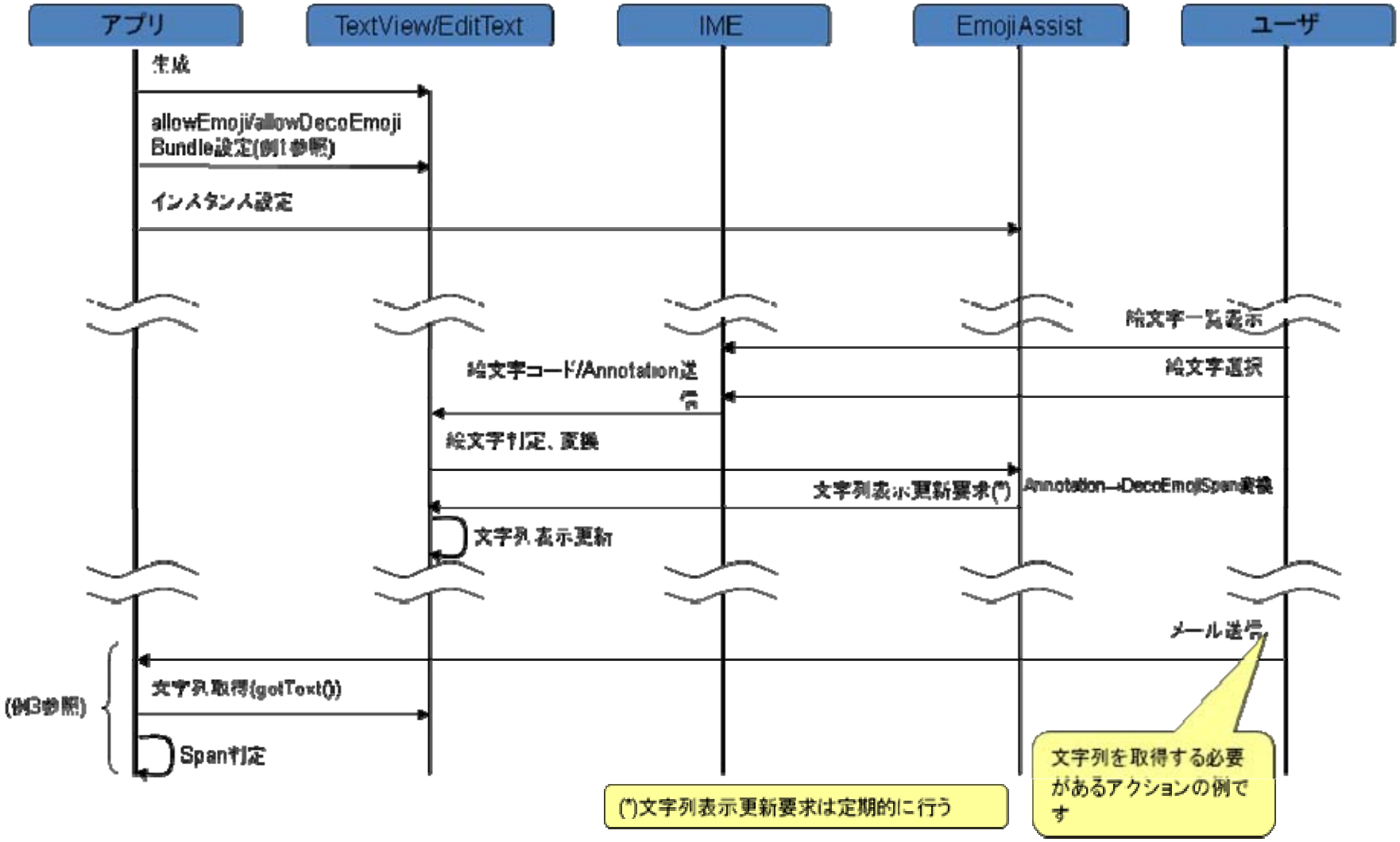



図 2　アプリ実行時のシーケンス



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

① TextView に絵文字フラグをバンドル

TextView に、絵文字の入力と表示を可能にするフラグを立てます。これは、TextView のインスタンスに Bundle パラメータとして、追加することによって行ないます。

“allowEmoji”

キャリア絵文字の入力と表示を可能にします。

“allowDecoEmoji”

デコメ絵文字の入力と表示を可能にします。

② インスタンス生成

絵文字表示を行う TextView のインスタンスを、絵文字制御ライブラリ EmojiAssist に引き渡します。

③-1 文字入力

1. TextView を通じて文字列の入力を行ないます。（ getText ）
   デコメ絵文字は通常文字、キャリア絵文字と同様、文字列の中の 1 文字として入力されます。文字コードは、別に定めます(*1)。通常表示に用いられない文字が利用されます。このコードは、定義によって変化する可能性があるため、このコードのみによって、デコメ絵文字を識別（デコメ絵文字として認識）してはなりません。
   *1:現在の定義は Unicode 0x3013、“＝”
2. 表示について
   TextView では入力が行われるとともに表示は行われているので、アプリケーションが表示のために行うことはありません。

③-2 デコメ絵文字の URI の読み込み

デコメ絵文字の保存や外部出力（たとえば、HTML への変換）のためには、ファイルとしての識別子が必要となります。

1. スパン属性の識別
   デコメ絵文字は DecoEmojiSpan というスパン属性をもちますのでこれで識別します。
2. URI の読み込み
   デコメ絵文字のバンドル情報から、URI を読み込みます。
3. ファイルの読み込み
   ファイルの実体は ContentProvider または直接ファイル入出力を用いて読み込みます。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 8. IME Interface

絵文字パッケージは IME と連携して、絵文字の入力を可能にします。IME は、絵文字の一覧選択機能と、絵文字辞書機能を備えることが必要です。

・一覧選択機能
- キャリア絵文字一覧選択機能
  キャリア絵文字の DB から一覧 Preview を表示し、選択する機能です。携帯電話事業者ごとに定義されます。
- デコメ絵文字一覧選択機能
  デコメ絵文字の DB から一覧 Preview を表示し、選択する機能です。

・絵文字辞書機能
- キャリア絵文字辞書
  キャリア絵文字をその読みから予測/変換するための辞書です。携帯電話事業者ごとに定義します。
- デコメ絵文字辞書
  デコメ絵文字をその読みから予測/変換するための辞書です。DB に格納されたコンテンツから、自動的に生成されます。（本パッケージで I/F を定義）

## 8.1. キャリア絵文字入力 I/F（A 機能）

A-1.一覧表示（パレット）入力

(1) 一覧表示アプリを用意します。
- IME からインテントで呼び出します。
- 選ばれたキャリア絵文字のコードを返却します。
- 選ばれた履歴を一定量保持し、先頭に表示します。

(2) 表示定義ファイル
- 表示順序定義
  一覧に表示するカテゴリと、表示順序は、一覧表示アプリのリソースとして、XML ファイルに定義します。

A-2.予測/変換入力

(1) キャリア絵文字を含む予測/変換辞書を用意します。
キャリア絵文字は文字コードを持ちますので、辞書を用意すれば入力は可能です。
（IME 毎にフォーマットは異なります。）

(2) IME 内の候補画面に絵文字を出力させるために、EmojiAssist を利用します。

## 8.2. デコメ絵文字一覧入力 I/F（B 機能）

(1) 一覧表示（パレット）入力
- 一覧表示アプリを用意します。
  - IME からインテントで呼び出します。
  - 選ばれたデコメ絵文字の URI を返却します。
  - 選ばれた履歴を一定量保持し、先頭に表示します。
- IME は、パッケージが提供するライブラリにより、URI をデコメ絵文字属性付き文字に変換して、アプリケーションに返します。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

・カテゴリ名定義に従い、デコメ絵文字は分類して表示されます。

(2) 表示定義ファイル

・カテゴリ名定義

- カテゴリ区分はサブディレクトリによって区別します。このディレクトリ名と表示する区分タイトルの対応を絵文字パッケージの設定として定義します。

**図 3　絵文字一覧の UI**



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 8.3. デコメ絵文字予測/変換 I/F（C 機能）

(1) 予測/変換入力

・デコメ絵文字用予測/変換辞書を用意します。

- デコメ絵文字は文字コードを持ちませんので、特別な辞書を用意することが必要です。（IME 毎にフォーマットは異なります。）
- デコメ絵文字辞書には、次の項目を含むことを想定します。
  - ・デコメ絵文字 ID（URI）
  - ・読み
  - ・品詞（オプション）

・IME 内の候補画面に絵文字を出力させるために、EmojiAssist を利用します。

(2)デコメ絵文字辞書の更新

・IME に対し、デコメ絵文字が追加されるとパッケージは DecoEmojiListener を通じて次の情報を提供します。

- デコメ絵文字 ID（URI）
- 読み
- 品詞

・IME はこの情報を受けて、デコメ絵文字辞書に情報を追加します。

・デコメ絵文字が削除されたとき、ID の削除を通知します。

- IME はデコメ絵文字辞書からデコメ絵文字を削除します。

・デコメ絵文字が変更されたとき、削除と追加を通知します。

- IME は、デコメ絵文字辞書の削除と追加を行ないます。

・IME は Android 端末の稼働中に切り換えられたり、新たにインストールされたりすることがあるため、絵文字パッケージのデコメ絵文字データベースとデコメ絵文字辞書が不一致になることがあります。このため、IME と絵文字パッケージとは適切なタイミングで同期を取る必要があります。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 8.4. IME へ提供する汎用インタフェース

1. A 機能（キャリア絵文字入力）

EmojiAssist ライブラリを提供し、これを TextView とともに候補画面に適用することによって、IME の候補画面でも画像表示を可能にします。
【インタフェース】
・絵文字表示ライブラリ（EmojiAssist）
・絵文字一覧表示アプリ（DecoEmojiList）

2. B 機能（デコメ絵文字一覧入力）

絵文字一覧(パレット)アプリをインテントインタフェースで提供します。IME は、デコメ絵文字を求められたとき、このアプリを呼び出してデコメ絵文字を入力することが可能です。
このアプリから返還されたデコメ絵文字の URI を、デコメ絵文字属性付きの文字コードに変換してアプリに引き渡すことにより、EmojiAssist がこれを解釈して TextView アプリに絵文字表示が行われるとともに、アプリにデコメ絵文字の URI を引き渡すことができます。
【インタフェース】
・絵文字表示ライブラリ（EmojiAssist、getDecoEmojiText）
・絵文字一覧アプリ（DecoEmojiList）

3. C 機能（デコメ絵文字の予測／変換入力）

デコメ絵文字に付属する読みを検出し、その ID(URI など)とともに IME に通知します。これを受けて、IME はデコメ絵文字辞書を生成することができます。
また、予測変換で得られたデコメ絵文字を、アプリに引き渡すためのデコメ絵文字属性付きの文字コードに変換する API を提供します。
【インタフェース】
・絵文字表示ライブラリ（EmojiAssist、getDecoEmojiText）
・絵文字マネージャ（DecoEmojiManager）

[説明]
1. 2. において、絵文字一覧を、絵文字一覧アプリを使わず、IME 内に組み込むことも可能です。絵文字一覧アプリは、Reference Implementation として、ソースコードも開示します。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

表 2　IME の絵文字対応とインタフェース

| IMEに実装を期待する機能 | | | IME固有機能 | 絵文字パッケージの用意するインタフェース | | | |
|---|---|---|---|---|---|---|---|
| | | | | (1)<br>絵文字表示Lib<br>(EmojiAssist) | (2)<br>絵文字一覧アプリ<br>(DecoEmojiList) | (3)<br>デコメ絵文字文字変換<br>(getDecoEmojiText) | (4)<br>デコメ絵文字辞書更新通知<br>(DecoEmojiManager I/F) |
| 凡例 | ●:絵文字パッケージインタフェースを利用する<br>◎:独自に機能を用意する | | IME側で機能を実装 | 候補画面(TextView)に絵文字を表示する | 絵文字一覧を表示し、文字を選択入力する | デコメ絵文字属性付き文字を生成する | 端末に登録されたデコメ絵文字を検出し、デコメ絵文字読み情報の追加、削除をIMEに通知する |
| A-1<br>絵文字<br>一覧入力 | 絵文字一覧を表示し、絵文字を選択入力する | 1.絵文字一覧を呼ぶ【注1】 | | | ● | | |
| A-2<br>絵文字<br>予測/変換<br>入力 | 通常文字と同様に絵文字(文字コードあり)の予測/変換を行う | 1.絵文字を含む辞書を用意する | ◎ | | | | |
| | | 2.候補画面(TextView)に絵文字を表示する | | ● | | | |
| B<br>デコメ絵文字<br>一覧入力 | 絵文字一覧を表示し、デコメ絵文字を選択入力する | 1.絵文字一覧を呼ぶ【注1】 | | | ● | | |
| | | 2.デコメ絵文字属性付き文字コードを生成し、文字列に含めてアプリに出力する。 | | | | ● | |
| C<br>デコメ絵文字<br>予測/変換<br>入力 | 通常文字と同様にデコメ絵文字の予測/変換を行う | 1.IMEの固有の機能を用いて、デコメ絵文字予測/変換辞書(入れ物)を用意する | ◎ | | | | |
| | | 2.デコメ絵文字属性付き文字コードを生成し、文字列に含めてアプリに出力する。 | | | | ● | ● |
| | | 3.候補画面(TextView)にデコメ絵文字を表示する | | ● | | | |
| | | 4.デコメ絵文字更新通知を受けて、デコメ絵文字予測/変換辞書(内容)を更新する | | | | | ● |

【注1】IME固有の絵文字一覧を実装することも可能。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 9. 【実装手順】アプリからの絵文字パッケージ利用手順

キャリア絵文字、デコメ絵文字が入力可能なアプリ開発において必要となる対応内容です。

## 9.1. 絵文字表示対応 【Level A/B/C】

### 9.1.1. 絵文字入力可能なエディタの生成

EditText で絵文字を入力する場合は、絵文字入力許可フラグを設定します。
具体的には、”allowEmoji”(キャリア絵文字)、”allowDecoEmoji”(デコメ絵文字)の Bundle パラメータを true にすることで、設定した EditText での絵文字入力が可能になります。

サンプルコード:

```
EditText text = new EditText(this);
Bundle b = text.getInputExtras(true);
b.putBoolean("allowEmoji", true);
b.putBoolean("allowDecoEmoji", true);
```

### 9.1.2. EmojiAssist に組込み、絵文字表示を可能にする

絵文字を表示するEditTextの生成時、そのインスタンスをEmojiAssistに組み込みます[1]。
EmojiAssist は絵文字パッケージの一部として提供されるプラグイン機能であり、次の機能を持ちます。

(1) 設定した EditText を定期的に更新(invalidate)し、アニメーション絵文字をアニメ動作させます。
(2) 設定した EditText の文字列内容が(IME からの入力などで)更新されたとき、更新内容にデコメ絵文字が含まれるかどうかをチェックし、含まれる場合は EditText で処理可能なオブジェクト(DecoEmojiSpan : android.text.style.ReplacementSpan の継承クラス)に変換します。DecoEmojiSpan は絵文字パッケージとして提供されるクラスで、デコメ絵文字の表示を行います。

サンプルコード:

```
// TextView のインスタンスを取得する
TextView emojiView1 = findViewById(R.id.emoji_view1);
TextView emojiView2 = findViewById(R.id.emoji_view2);

// EmojiAssist を TextView に割り当てる
EmojiAssist ea = EmojiAssist.getInstance();
ea.addView(emojiView1);
ea.addView(emojiView2);

// EmojiAssist のアニメ動作を開始させる
ea.startAnimation();
```

### 9.1.3. 文字列からデコメ絵文字を取得

EditTextの文字列を取得して処理を行う場合、デコメ絵文字(DecoEmojiSpan)が含まれるかどうかをチェッ

[1] システム標準アプリで使用されているTextViewおよびその継承部品(ステータス、アプリ一覧など)については、それらのアプリ自体を改造しない限りEmojiAssistを組み込むことができないため、キャリア絵文字のアニメ表示、およびデコメ絵文字の表示ができません。キャリア絵文字は、EmojiFactoryにより静止画が表示されます。デコメ絵文字は「=」が表示されます。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

クし、デコメ絵文字が含まれる文章に対応する[2]ことが必要な場合があります。デコメ絵文字の実体は画像ファイルですので、DecoEmojiSpanからデコメ絵文字ファイルのURI情報を取り出します。

サンプルコード:

```
  android.text.Spanned sp = (Spanned)edittext.getText();
  DecoEmojiSpan[] decoEmojiSpans = sp.getSpans(0, sp.length(),
DecoEmojiSpan.class);
  String uri = decoEmojiSpans[0].getURI();
```

[2] メーラであれば、HTMLメール形式に変換するなど。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 10. 【実装手順】IME からの絵文字パッケージ利用手順

キャリア絵文字、デコメ絵文字が入力可能なIME開発において、一般的なアプリで必要となる「9【実装手順】アプリからの絵文字パッケージ利用手順」に加えて、IMEとして対応が必要となる内容を記載しています。

## 10.1. IME のキャリア絵文字一覧入力機能追加【Level A】

キャリア絵文字一覧は、IME に機能追加して実現するか、絵文字一覧入力アプリ DecoEmojiList を IME 側から起動する動線を追加します。絵文字一覧アプリは Intent により起動されるアプリとして実装されています。

## 10.2. IME のキャリア絵文字予測/変換入力対応【Level A】

IME の予測/変換入力機能に、キャリア絵文字も候補に含めるためには、IME にて下記の対応を行う必要があります。

(a) キャリア絵文字辞書を追加する。
(b) 候補表示において、絵文字パッケージのキャリア絵文字表示のための対応を行う。

## 10.3. IME のデコメ絵文字一覧起動・入力の追加【Level B】

### 10.3.1. デコメ絵文字一覧の起動

デコメ絵文字一覧は、IMEに機能追加して実現するか、絵文字一覧入力アプリDecoEmojiListをIME側から起動する動線を追加する必要があります[3]。絵文字一覧アプリはIntentにより起動されるアプリとして実装されています。

サンプルコード:

```
  final PackageManager packageManager = ims.getPackageManager();
   // ims...InputMethodService object
  Intent intent = new Intent();
  intent.setClassName(DecoEmojiListStarter.NEXT_FULL_PACKAGE_NAME
              , DecoEmojiListStarter.NEXT_FULL_ACTIVITY_NAME);

  List<android.content.pm.ResolveInfo> activityCount =
      packageManager.queryIntentActivities(intent,
PackageManager.MATCH_DEFAULT_ONLY);
  if (0 < activityCount.size()) {
      // Exist Activity.
      int[] currentImeHeightArray = getCurrentImeHeight();
      intent = new Intent();
      intent.setClass(ctx, DecoEmojiListStarter.class);  // (*)
      intent.addFlags(Intent.FLAG_ACTIVITY_NEW_TASK);
```

[3] 同様の機能を、IMEではなくメーラ等のアプリで実装することも考えられます。その場合、絵文字一覧入力アプリからのレスポンスを当該アプリで受け付け、受け取ったキャリア絵文字/デコメ絵文字を編集画面に反映するなどの処理を追加する必要があります。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

```
        intent.putExtra(DecoEmojiListStarter.EXTRA_CURRENT_IME_HEIGHT_PORT
                    , currentImeHeightArray[0]);
        intent.putExtra(DecoEmojiListStarter.EXTRA_CURRENT_IME_HEIGHT_LAND
                    , currentImeHeightArray[1]);
        ctx.startActivity(intent);  // (*)
    }

    // (*)ctx は Context のサブクラス；Activity や Service のオブジェクトを指定する
```

### 10.3.2. 入力されたデコメ絵文字の伝送

絵文字一覧入力アプリは、入力されたデコメ絵文字の列を、その起動元(通常 IME)に対し次の形式の Bundle 情報として返します。

| 型 | 内容 |
|---|---|
| List | android.net.Uri の List |

IME でデコメ絵文字を扱うには、IME で Bundle 内の List から URI(デコメ絵文字を示す)を取り出し、デコメ絵文字属性付き文字コードに変換して EditText に伝達するように修正する必要があります。

サンプルコード:

```
EmojiAssist ea = EmojiAssist.getInstance();
// URI からデコメ絵文字属性付き文字を生成する
CharSequence decoemoji = ea.getDecoEmojiText(uriString);
```

## 10.4. IME のデコメ絵文字予測/変換入力対応【Level C】(option)

IME の予測/変換入力機能に、デコメ絵文字も候補に含めるためには、下記の対応を行う必要があります。

(a) 予測/変換に用いるデコメ絵文字辞書を追加する。
(b) 候補表示において、絵文字パッケージのデコメ絵文字表示のための対応を行う。
(c) デコメ絵文字の追加/削除の通知を DecoEmojiManager から受け取る機能を追加する。

絵文字パッケージでは、IME でデコメ絵文字の予測/変換を可能にするためのインタフェースを用意しています。

・絵文字マネージャ(DecoEmojiManager)
　デコメ絵文字読み情報(辞書更新)通知インタフェース
・絵文字表示ライブラリ(EmojiAssist)
　候補画面(TextView)表示、デコメ絵文字生成インタフェース

詳しくは、「11【実装手順】デコメ絵文字読み情報更新手順」を参照ください。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 11. 【実装手順】デコメ絵文字読み情報更新手順

## 11.1. デコメ絵文字辞書情報更新の基本手順

辞書情報更新の関係モジュール

(1) 絵文字マネージャ（DecoEmojiManager）
デコメ絵文字の登録、変更、削除を検知すると、IME に対して、更新情報を載せて Intent で通知を行います。絵文字マネージャではあらかじめ IME が特定できないため、ブロードキャストを行います。
(2) IME / DecoEmojiListener
絵文字マネージャからの Intent を受け取って、デコメ絵文字の辞書を更新します。Intent は、辞書更新情報をブロック化して分割して送信されるため、辞書の更新後、残りの更新がないか絵文字マネージャに問い合わせて、次の更新情報の通知を待ちます。絵文字マネージャへの問い合わせ/要求は、絵文字マネージャサービスへの Bind（aidl）インタフェースによるプロセス間通信で行います。

（実装例）
DecoEmojiListener は IME 側で Intent のレシーバの役割を担います。
１．DecoEmojiListener（Listener と省略）は絵文字マネージャからの Intent のレシーバ機能を持つ。IME が起動しているか否かに関わらず、絵文字マネージャからの更新通知（Intent）を受けられることが望ましい。絵文字マネージャの絵文字検出と通知は IME とは非同期に行われるため、IME が起動していないときの通知を逃すことで辞書に不一致が生じるためである。（IME 起動時に、起動していない間の通知を問い合わせる実装も可能。）
２．一方で、Listener がデコメ絵文字辞書を更新するとき、IME とは、デコメ絵文字辞書を共有することになるため、両者で競合管理を行うことが必要なことがある。
３．IME と Listener を、同一プロセス（別スレッド）とするか別プロセスとするかはインプリメントに依存する。

**表 3　モジュール間のメッセージ**

| メッセージ | 通知 | 経路 | IME の処理 | 備考 |
|---|---|---|---|---|
| ①デコメ絵文字読み情報更新通知 | M→I | Intent | デコメ絵文字辞書の更新（追加と削除）<br>IME 不特定時のブロードキャストと<br>②③に応える IME 向けユニキャストがあるが処理は同じ | 最大 100 件の通知 |
| ②追加更新チェック | I→M | Bind | 更新した最終 ID を通知し、追加の更新通知を要求する | |
| ③デコメ絵文字辞書初期化要求 | I→M | Bind | 全読み情報の再送を求める | 初期処理 |
| ④デコメ絵文字辞書初期化通知 | M→I | Intent | デコメ絵文字辞書のリセットを行う | 初期処理 |

M：絵文字マネージャ、　I：IME

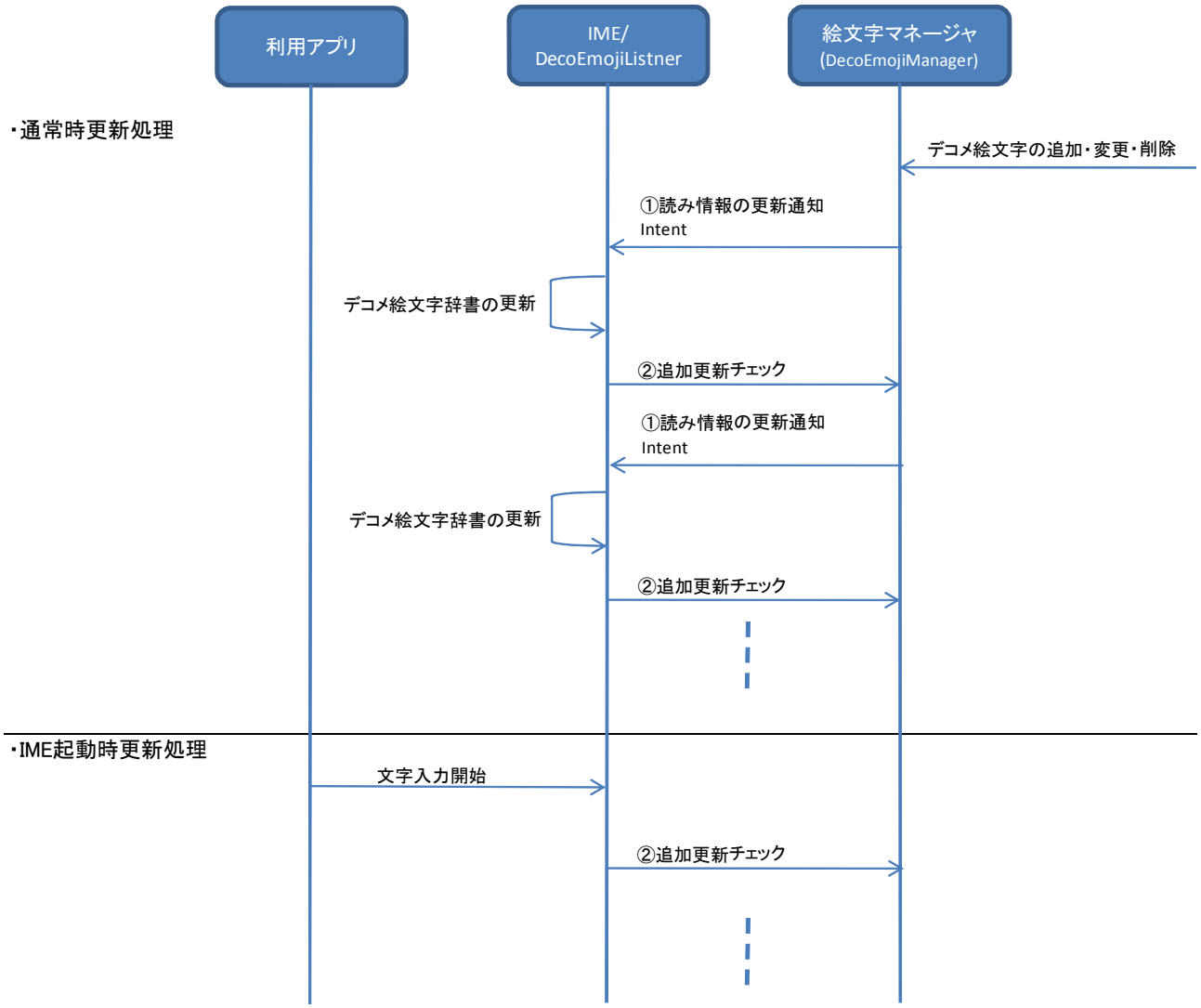


Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 11.2. シーケンス図

想定している実装シーケンスを示します。

(1) 通常更新処理

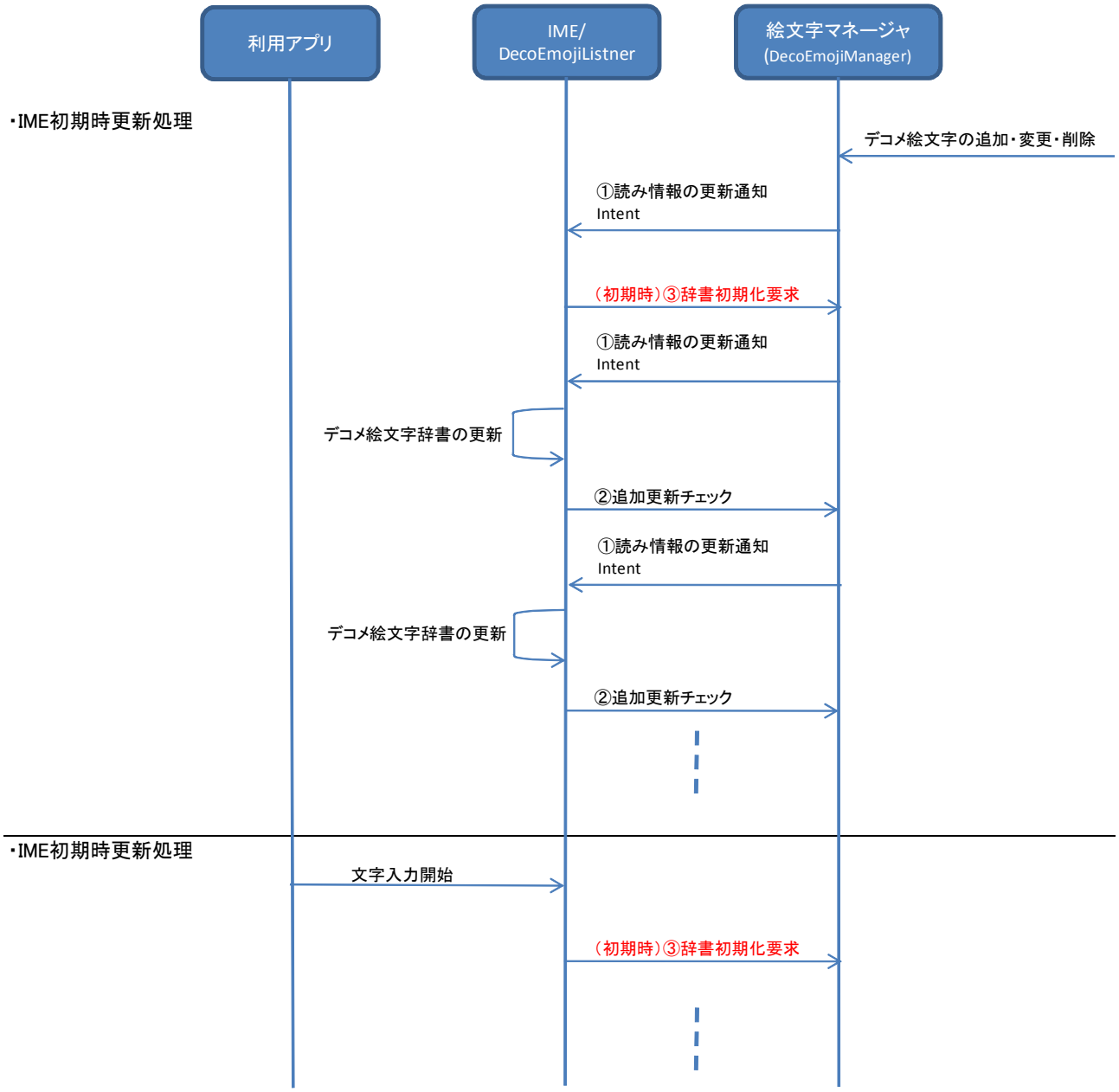





Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

(2) IME 初期処理

IME が初めてデコメ絵文字辞書を作成するとき、絵文字マネージャに全更新データを求めます。

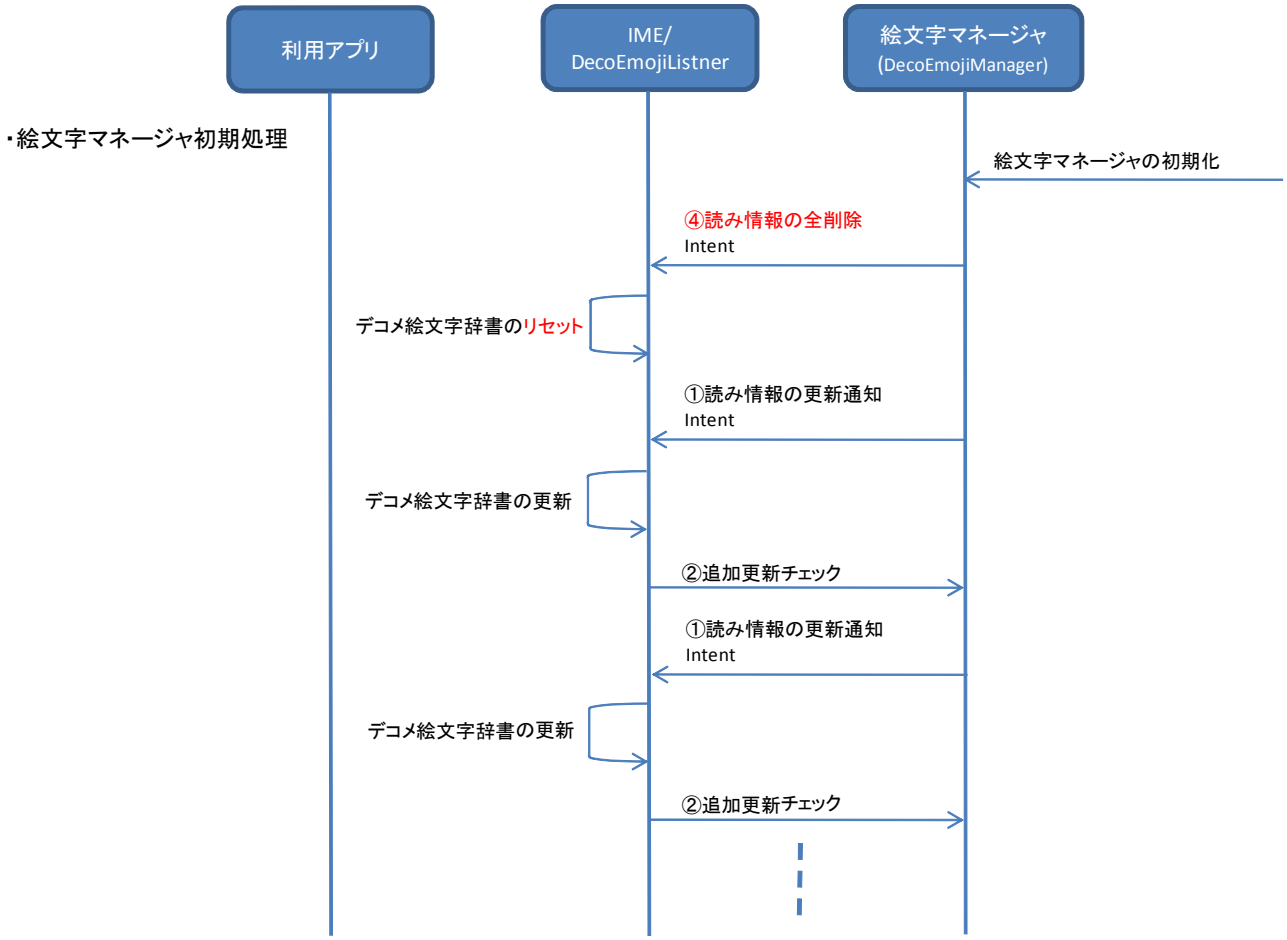





Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

(3) 絵文字マネージャ初期処理

絵文字マネージャ自身が初期化されたとき、IME に、デコメ絵文字辞書の再作成を求めます。






Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 12. 【実装手順】デコメ絵文字の一覧表示の更新

【対応バージョン：絵文字パッケージ V1.1】

絵文字マネージャがデコメ絵文字の更新を検出した時に、このことを Broadcast する機能が提供されました。

このことにより、デコメ絵文字の一覧表を表示したり、デコメ絵文字の付加情報を利用する IME などのサービスアプリケーションが、デコメ絵文字の更新の通知を受けて、一覧表などの情報を更新することが可能となりました。

## 12.1. 一覧表示アプリケーションの表示更新【Level C】

絵文字の一覧表示など、自アプリ内にデコメ絵文字の情報を取り込んで処理をしているアプリの場合、動的に変化しているデコメ絵文字の更新情報を取り込む手段がなく、表示内容を最新の状態に保つことが難しいという課題がありました。

絵文字マネージャ（DecoEmojiManager）は、登録先ディレクトリ内のデコメ絵文字の更新（追加／変更／削除）を検出すると、Broadcast でこれを通知します。

また、Bundle データとして、更新を検出した（更新データを DB に登録した）Timestamp を通知します。

一覧表示アプリケーションは、この Broadcast を receive してデコメ絵文字情報を取り込み、自らの表示内容を更新することができます。

デコメ絵文字の一覧表の更新方法にはいくつかの方法があります。

(1) 絵文字一覧アプリを用いている場合、絵文字一覧アプリを再起動し、一覧情報を取り直す。
詳細は「14.5絵文字一覧アプリDecoEmojiList」を参照ください。

(2) Bind(aidl)インタフェースを用いて、絵文字マネージャから一覧情報を取り直す。
詳細は「14.6絵文字マネージャDecoEmojiManager」を参照ください。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 13. 【実装手順】IME に関する付加情報

## 13.1. Android Framework 全画面モード時の既存不具合対応

Android Framework には IME の全画面モード時の既存不具合が存在します。(V2.3 現在)

・ 全画面モードに適用される ExtraEditText 内で文字の編集中に、Span 属性が削除されることがある。

その不具合対応として、IME 側に ExtractEditText の差し替え処理およびそこで必要となるリソースの追加を実装する必要があります。

絵文字パッケージでは、EmojiAssist 内に ExtraEditText の代替え処理を用意し、iWnn IME の V2 をベースとした追加リソースおよび ExtractEditText の差し替え処理の修正差分を用意しています。カスタム対応された iWnnIME に適用する場合は、その差分をカスタム版に反映する必要があります。

他の IME にも、参考情報としてこれを提供することは可能です。

追加リソース:

・res¥drawable¥emoji_extract_edit_text.xml
・res¥drawable-hdpi¥emoji_keyboard_textfield_selected.9.png
・res¥drawable-hdpi¥emoji_textfield_disabled.9.png
・res¥layout¥emoji_extract_view.xml

サンプルコード:

```
  @Override public View onCreateExtractTextView() {
      View v = super.onCreateExtractTextView();
      extractEditText =
(ExtractEditText)EmojiAssist.replaceExtractEditText((ViewGroup)v,
       R.layout.emoji_extract_view);
      return v;
  }
```

## 13.2. スリープ時の注意事項

絵文字パッケージでは EmojiAssist (jar) において、絵文字表示中の場合、一定時間ごとに描画処理を行っています。そのため、描画処理を明示的に停止させなければ、スリープ時に消費電力が大きくなります。

スリープ時に行うべき処理を以下に記載します。

### 13.2.1. 対応策 (アクティビティアプリケーション向け)

EmojiAssist を使用するアクティビティアプリケーション(一覧表示アプリなど)においては、 スリープモード遷移時に onPause() 関数がコールされます。この関数内において、EmojiAssist に対し、以下のように描画処理を停止させる必要があります。

```
@Override
protected void onPause() {
    --- 中略 ---
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

```
        mEmojiAssist.stopAnimation();
        --- 中略 ---
    }
```

なお、描画処理を開始／再開させる場合には、EmojiAssist#startAnimation() をコールする必要があります。
startAnimation()、stopAnimation() の使用例につきましては、絵文字パッケージ内のリファレンスソースコード「DecoEmojiList/src/DecoEmojiList.zip」をご参照下さい。

### 13.2.2. 対応策 ((サービスアプリケーション向け (IME を含む))

EmojiAssist を使用するサービスアプリケーション (IME を含む)の場合、スリープモード遷移時に onPause() 関数はコールされないため、BroadcastReceiver にてスクリーン OFF を検知し、EmojiAssist に対し、以下のように描画処理を停止させる必要があります。

・BroadcastReceiver において、「ACTION_SCREEN_ON」を検知し、startAnimation() をコールする
・BroadcastReceiver において、「ACTION_SCREEN_OFF」を検知し、stopAnimation() をコールする

以下に、BroadcastReceiver を使ったサンプルコードを記載します。

```
    private BroadcastReceiver mReceiver = new BroadcastReceiver() {
        @Override
        public void onReceive(Context context, Intent intent) {
            if (mIsEmojiAssistWorking) {   // EmojiAssist が動作している場合
                if (intent.getAction().equals(Intent.ACTION_SCREEN_ON)) {
                    mEmojiAssist.startAnimation();
                } else if (intent.getAction().equals(Intent.ACTION_SCREEN_OFF)) {
                    mEmojiAssist.stopAnimation();
                }
            }
        }
    };
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 14. 【API】インタフェース仕様

以下、各コンポーネントについて、インタフェース仕様を規定する。

なお、文中のパッケージ名については事業者ごとに異なる場合があるため、該当部分を”<carrier>”として表記している。

例："jp.co.omronsoft.android.decoemojimanager_<carrier>"

## 14.1. TextView/EditText（絵文字入力の有効化）

アプリで使用する TextView(EditText)で、キャリア絵文字、デコメ絵文字を入力したい場合は、下記のように絵文字有効フラグ（allowEmoji）及びデコメ絵文字有効フラグ（allowDecoEmoji）を Bundle する必要がある。それぞれ個別にも指定可能である。

```
EditText text = new EditText(this);
Bundle b = text.getInputExtras(true);
b.putBoolean("allowEmoji", true);
b.putBoolean("allowDecoEmoji", true);
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 14.2. EmojiAssist クラス

パッケージ名 : jp.co.omronsoft.android.emoji

### 14.2.1. インスタンス取得

| API 名 | getInstance | | | | |
|---|---|---|---|---|---|
| 説明 | EmojiAssistのインスタンスを取得する。 | | | | |
| **引数** | | | | | |
| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
| | | | | | |
| | | | | | |
| | | | | | |
| **返値** | | | | | |
| No | 型 | 説明 | | | |
| 1 | EmojiAssist | EmojiAssist のインスタンス | | | |
| **前提**<br>なし<br><br>**概略説明**<br>EmojiAssistのインスタンスを取得する。 | | | | | |



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.2. TextView 登録

| API 名 | addView | | | | |
|---|---|---|---|---|---|
| 説明 | TextViewを登録する。 | | | | |
| **引数** | | | | | |
| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
| 1 | view | TextView | I | Must | 登録するTextView |
| | | | | | |
| | | | | | |
| **返値** | | | | | |
| No | 型 | 説明 | | | |
| | | | | | |

**前提**
getInstanse()で取得した EmojiAssist で実行すること。

**概略説明**
TextViewをEmojiAssistに登録する。

**補足**
同一プロセス内で絵文字を描画されうるTextViewは全てEmojiAssistに登録すること。
アニメーション動作中に実行してもよい。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.3. TextView 削除

| API 名 | removeView |
|---|---|
| 説明 | 登録したTextViewを削除する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| 1 | view | TextView | I | Must | 削除するTextView |
| | | | | | |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| | | |

**前提**
TextView が登録済みであること

**概略説明**
登録したTextViewをEmojiAssistから削除する。

**補足**
アニメーション動作中に実行してもよい。
登録されていないviewが指定された場合は、無視される。
終了処理などで登録したTextViewの更新が不要になる場合は、removeViewなどで登録を解除すること。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.4. TextView 全削除

| API 名 | clearView |
|---|---|
| 説明 | 登録したTextViewを全て削除する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| | | |

**前提**
なし

**概略説明**
登録したTextViewをEmojiAssistから全て削除する。

**補足**
アニメーション動作中に並行実行してもよい。
終了処理などで登録したTextViewの更新が不要になる場合は、clearViewなどで登録を解除すること。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.5. アニメーション開始

| API 名 | startAnimation |
|---|---|
| 説明 | 絵文字のアニメーションを開始する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| | | |

**前提**
なし

**概略説明**
絵文字のアニメーションを開始する。addViewで登録したTextViewに対して定期的にinvalidate()するタイマーを開始させる。TextViewのアニメーションが不要になるときはstopAnimationにより停止させる必要がある。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.6. アニメーション停止

| API 名 | stopAnimation |
|---|---|
| 説明 | 絵文字のアニメーションを停止する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| | | | | | |
| | | | | | |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| | | |

**前提**
なし

**概略説明**
絵文字のアニメーションを停止する。
スリープモード遷移時など、アニメーションが不要になるときは、本 API により停止させること。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.7. デコメ絵文字 Span 設定

| API 名 | getDecoEmojiText | | | | |
|---|---|---|---|---|---|
| 説明 | デコメ絵文字属性（Annotation）を設定した文字を取得する。 | | | | |
| **引数** | | | | | |
| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
| 1 | uri | String | I | Must | デコメ絵文字の画像ファイル格納位置を指す URI |
| | | | | | |
| | | | | | |
| **返値** | | | | | |
| No | 型 | 説明 | | | |
| 1 | CharSequence | デコメ絵文字 Span を設定した文字 | | | |

**前提**
なし

**概略説明**
IMEからアプリケーションに出力するデコメ絵文字属性（android.text.Annotation）を設定した文字を取得する。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.8.絵文字データ取得

| API 名 | getEmojiImage |
|---|---|
| 説明 | キャリア絵文字の画像データ(Binary)を取得する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| 1 | code | int | I | Must | 対象文字のAndroidPUA |
| 2 | textsize | float | I | Must | フォントサイズ（文字の高さ）[pixel] |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| 1 | byte[] | 絵文字データ(Binary)(失敗時：NULL) |

**前提**

なし

**概略説明**

指定された文字コードのキャリア絵文字について、指定サイズにもっとも近いサイズのGIF画像データ(Binary)を取得する。

例：キャリア絵文字として、下記の5種類が用意されている場合；

・12pixel:　14 pixel以下の時に表示する画像
・16pixel:　15～18 pixelの時に表示する画像
・20pixel:　19～22 pixelの時に表示する画像
・24pixel:　23～26 pixelの時に表示する画像
・28pixel:　27 pixel以上の時に表示する画像

サイズ17を指定すると
・15～18 pixelの時に表示する画像
が取得される。
なお、標準の絵文字パッケージでは48x48の画像のみ用意されている。

サンプルコード:

```
StringBuffer htmlBuf = new StringBuffer(html); // html は HTML ソースを所持する String

for (int i = 0; i < htmlBuf.length(); i++) {
  // HTML から絵文字コードを検出する
  int theCode = htmlBuf.codePointAt(i);
  if (Character.isHighSurrogate(htmlBuf.charAt(i))
      && EmojiDrawable.isEmoji(theCode)) {
    byte[] imgData = ea.getEmojiImage(theCode, 24);  /* ea は EmojiAssist のインスタンスとする */

    // imgData を GIF ファイル名 imgFile に書き込む(詳細略)

    // HTML を生成する
    String mappingHTML = "<img src=¥"" + imgFile + "¥">";

    // 当該キャリア絵文字の文字コード部分を、img タグに置換する
    htmlBuf.replace(i, i+2, mappingHTML);

    // サロゲートペアを mappingHTML に置換したため、文字数分インクリメントする
    i = i + 2 - 1 + mappingHTML.length;
  } // else do nothing
}
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.2.9. ExtractEditText 差し替え

| API 名 | replaceExtractEditText |
|---|---|
| 説明 | ExtractEditTextを差し替える。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| 1 | view | ViewGroup | I | Must | ExtractEditTextを持つView |
| 2 | id | int | I | Must | 差し替えるExtactEditTextのリソースID |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| 1 | ExtractEditText | 差し替えた ExtractEditText 継承クラスのインスタンス(失敗時：NULL) |

**前提**

なし

**概略説明**

全画面表示で使用されるExtractEditText クラスを、任意の継承クラスに置換する(IME向け)

**補足**

フレームワークの全画面表示用部品ExtractEditTextは絵文字表示において不具合が発生するため、絵文字パッケージではその不具合を回避するjp.co.omronsoft.android.inputmethodservice.EmojiExtractEditTextを用意しています。IMEで全画面への切り替えを検出した際、EmojiExtractEditTextに差し替える必要があります。フレームワークのバージョンアップによりExtractEditTextが改善されれば、本対応は不要になります。

サンプルコード:

```
public class MyIME extends InputMethodService {
  private ExtractEditText mExtractEditText;
  ＜中略＞
  @Override public View onCreateExtractTextView() {
      View v = super.onCreateExtractTextView();
      EmojiAssist assist = EmojiAssist.getInstance();
      mExtractEditText = assist.replaceExtractEditText((ViewGroup)v,
R.layout.emoji_extract_view);
      return v;
  }
  public ExtractEditText getExtractEditText() {
      return mExtractEditText;
  }
  ＜以下略＞
```

サンプルコード(emoji_extract_view.xml):

```
<jp.co.omronsoft.android.inputmethodservice.EmojiExtractEditText
        xmlns:android="http://schemas.android.com/apk/res/android"
        android:scrollbars="vertical"
        android:gravity="top"
        android:minLines="1"
        android:inputType="text"
        android:background="@drawable/emoji_extract_edit_text"
    >
</jp.co.omronsoft.android.inputmethodservice.EmojiExtractEditText>
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 14.3. DecoEmojiSpan クラス

デコメ絵文字情報を保持し、描画を受け持つクラスである。
パッケージ名：jp.co.omronsoft.android.text.style

### 14.3.1. URI 取得

| API 名 | getURI | | | | |
|---|---|---|---|---|---|
| 説明 | デコメ絵文字の画像ファイルを指すURIを返却する。 | | | | |
| **引数** | | | | | |
| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
| | | | | | |
| | | | | | |
| | | | | | |
| **返値** | | | | | |
| No | 型 | 説明 | | | |
| 1 | String | URI | | | |

**前提**
なし

**概略説明**
デコメ絵文字の画像ファイルを指すURIを返却する。

サンプルコード：

```
CharSequence charSeq = text.getText();
SpannableStringBuilder ssb = SpannableStringBuilder.valueOf(charSeq);
DecoEmojiSpan[] decoEmojiSpans = ssb.getSpans(0, ssb.length(), DecoEmojiSpan.class);
String uri = decoEmojiSpans[0].getURI();
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 14.4. EmojiDrawable クラス

パッケージ名 : jp.co.omronsoft.android.text

### 14.4.1. 絵文字判定

| API 名 | isEmoji |
|---|---|
| 説明 | 引数の文字列に絵文字が含まれているか否かを判定する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| 1 | text | CharSequence | I | Must | 判定対象文字列 |
| | | | | | |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| 1 | static boolean | true : 絵文字が含まれている |

**前提**
なし

**概略説明**
引数の文字列に絵文字が含まれているか判定し、結果を返却する。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.4.2.絵文字判定(コード)

| API 名 | isEmoji |
|---|---|
| 説明 | 引数の文字が絵文字か否かを判定する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| 1 | code | int | I | Must | 判定対象文字 |
| | | | | | |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| 1 | static boolean | true：絵文字である |

**前提**
なし

**概略説明**
引数の文字が絵文字かどうか判定し、結果を返却する。
引数にはString.codePointAt()等で取得したコードを渡す。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 14.5. 絵文字一覧アプリ DecoEmojiList

絵文字一覧アプリは、IME や一般アプリから Intent によって起動され、選択された絵文字を結果として返却する。

絵文字一覧アプリを使用するアプリは、リファレンスアプリケーション DecoEmojiList のソースコードのうち下記のファイルをアプリのビルド環境にコピーして利用する必要がある。

DecoEmojiListResultInfo.java

### 14.5.1. Intent 起動

IME もしくはアプリから絵文字一覧アプリを起動させるには、下記の通り Intent を発行する。

サンプルコード:

```
Intent intent = new Intent();
// クラス名にデコメ絵文字一覧アプリを指定
intent.setClassName("jp.co.omronsoft.android.decoemojilist_<carrier>",
"jp.co.omronsoft.android.decoemojilist_<carrier>.DecoEmojiList");(*1)
// 呼び出す Activity 以外の Activity をクリアして起動させる設定を付与
intent.setFlags(Intent.FLAG_ACTIVITY_CLEAR_TOP);
// デコメ絵文字一覧アプリ上で表示する絵文字タグ切替値の設定
intent.putExtra("switch_emoji_flg", 1); // 0:both、1:deco、2:carrier
// 縦画面時のデコメ絵文字一覧アプリを表示する高さの領域を設定
intent.putExtra("current_ime_height_port", heightP);
// 横画面時のデコメ絵文字一覧アプリを表示する高さの領域を設定
intent.putExtra("current_ime_height_land", heightL);
// デコメ絵文字取得方法を設定
intent.putExtra("current_input_type", 0); // 0:複数候補選択後にアプリに返す、1:候補を
一つ選択した時点でアプリに返す
// デコメ絵文字一覧アプリ起動
startActivityForResult(intent, 0);   // request code = 0
```

(*1) パッケージ名。<carrier>は事業者によって異なる。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.5.2. 返値取得

起動した絵文字一覧アプリにおいて選択された絵文字情報を取得するには、受け取った Intent の Result から List を取り出す。

サンプルコード:

```
@Override
protected void onActivityResult(int requestCode, int resultCode, Intent data) {
    if (requestCode == 0 && resultCode == RESULT_OK) { // req. code = 0,
        ArrayList<DecoEmojiListResultInfo> result = data
                .getParcelableArrayListExtra("uri_list");

        DecoEmojiListResultInfo resultInfo;
        for (int i = 0; i < result.size(); i++) {
            resultInfo = result.get(i);
            if (resultInfo.getTag() == 1) {  // 1:deco, 2:carrier
                // デコメ絵文字
                CharSequence charSequence = resultInfo.getUri();
                 :
            } else {
                // キャリア絵文字(String の各文字がキャリア絵文字の文字コード)
                CharSequence charSequence = resultInfo.getCode();
                 :
            }
        }
    }
    finish();
}
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

## 14.6. 絵文字マネージャ DecoEmojiManager

絵文字マネージャ DecoEmojiManager は、デコメ絵文字の画像ファイルと ID を管理するクラスである。

システムで共通に利用するデコメ絵文字は、所定のディレクトリ(登録先ディレクトリ)に格納される必要がある。登録先ディレクトリに登録されたデコメ絵文字は、絵文字パッケージに管理され、アプリケーションから絵文字として利用(入力、表示)することができる。端末の別アプリ(メーラ、ブラウザやコンテンツ管理アプリ)が新規にデコメ絵文字を入手して端末(SD カードなど)に保存したり削除したりする場合、その更新を登録先ディレクトリに加えることによって、絵文字パッケージに通知することができる。

デコメ絵文字の一覧を表示したり、デコメ絵文字をアプリケーションに入力するサービスを提供するアプリ(IME)や、デコメ絵文字の付加情報を利用するアプリは、DecoEmojiManager サービスを通じてそれらの情報にアクセスすることができる。

DecoEmojiManager は、サービスとして動作する。サービスに接続する場合は下記のように bind する必要がある。

なお、DecoEmojiManager のインタフェースおよび関連する定数値を利用するアプリは、ライブラリ DecoEmojiInterface.jar をリンクする必要がある。

また、デコメ絵文字予測/変換入力機能を含む場合は、DecoEmojiManager から IME に対し、Intent により情報を通知する。

サンプルコード:

```
// サービス接続
Intent intent = new Intent(IDecoEmojiManager.class.getName());
bindService(intent, mServiceConn, Context.BIND_AUTO_CREATE);

// bindService のコールバック関数
private ServiceConnection mServiceConn = new ServiceConnection() {
    // サービス接続時のコールバック関数
    public void onServiceConnected(ComponentName name, IBinder binder) {
        mDecoEmojiInterface = IDecoEmojiManager.Stub.asInterface(binder);
    }
    // サービス切断時のコールバック関数
    public void onServiceDisconnected(ComponentName name) {
        mDecoEmojiInterface = null;
    }
};

// サービス切断
unbindService(mServiceConn);
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.6.1. 絵文字マネージャのインタフェース IDecoEmojiManager の API

DecoEmojiManager のサービスとして提供されている Bind(aidl)インタフェースの API について説明する。
パッケージ名 : jp.co.omronsoft.android.decoemojimanager_<carrier>.interfacedata

#### 14.6.1.1. URI 取得

| API 名 | aidl_getDecoUri | | | | |
|---|---|---|---|---|---|
| 説明 | デコメ絵文字のURIを取得する。 | | | | |
| **引数** | | | | | |
| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
| 1 | decoemoji_id | int | I | Must | デコメ絵文字ID … 取得対象デコメ絵文字のデコメ絵文字ID<br>デコメ絵文字ID : 0～9999 |
| 2 | uriInfo | DecoEmojiUriInfo | I/O | Must | URI … 対象デコメ絵文字のURIを格納した DecoEmojiUriInfoのオブジェクト |
| | | | | | |
| **返値** | | | | | |
| No | 型 | 説明 | | | |
| 1 | int | IDecoEmojiConstant.OK : 正常終了<br>IDecoEmojiConstant.NG_INITIALIZING : 初期化処理中<br>IDecoEmojiConstant.NG_ID : 「decoemoji_id」の値が範囲外<br>IDecoEmojiConstant.NG_URI_INFO : URI の取得中に問題が発生（DB エラーを含む） | | | |

**前提**
DecoEmojiManager により初期データが構築されていること

**概略説明**
指定されたデコメ絵文字IDに紐づくデコメ絵文字のURIを返却する。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.6.1.2. カテゴリ一覧取得

| API 名 | aidl_getCategoryList | | | | |
|---|---|---|---|---|---|
| 説明 | デコメ絵文字のカテゴリ情報一覧を返却する。 | | | | |
| **引数** | | | | | |
| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
| 1 | maxDisplaycnt | int | I | Must | 1画面に表示するデコメ絵文字の数（0 < maxDisplaycnt <= 72） |
| 2 | categoryList | List<DecoEmojiCategoryInfo> | O | - | 取得したカテゴリ情報のList |
| | | | | | |
| **返値** | | | | | |
| No | 型 | 説明 | | | |
| 1 | int | IDecoEmojiConstant.OK：正常終了<br>IDecoEmojiConstant.NG_MAX_DISPLAY_CNT：第 1 引数「maxDisplaycnt」の値が範囲外<br>IDecoEmojiConstant.NG_CATEGORY_INFO：カテゴリ情報の取得中に問題が発生した（DB エラーを含む） | | | |
| **前提**<br>DecoEmojiManager により初期データが構築されていること<br><br>**概略説明**<br>デコメ絵文字のカテゴリ情報一覧を返却する。 | | | | | |



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.6.1.3. URI 一覧取得

| API 名 | aidl_getDecoUriList |
|---|---|
| 説明 | 引数のカテゴリIDに紐づくデコメ絵文字のURI一覧を返却する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| 1 | categoryId | int | I | Must | 取得対象カテゴリのID（1 <= カテゴリID <= intの最大値） |
| 2 | pageIndex | int | I | Must | 取得対象のページ番号（0オリジン） |
| 3 | uriList | List<String> | O | - | 指定されたページに表示するデコメ絵文字のURIのList |

**返値**

| No | 型 | 説明 |
|---|---|---|
| 1 | int | IDecoEmojiConstant.OK：正常終了<br>IDecoEmojiConstant.NG_ID：第 1 引数「categoryId」の値が範囲外<br>IDecoEmojiConstant.NG_PAGE_CNT：第 2 引数「pageIndex」で指定されたページが存在しない<br>IDecoEmojiConstant.NG_URI_INFO：URI の取得中に問題が発生した（DB エラーを含む）<br>IDecoEmojiConstant.NG_DISP_CNT：表示数が未設定 |

**前提**

DecoEmojiManager により初期データが構築されていること
利用前に、aidl_getCategoryList()を呼び出し、あらかじめ表示数を設定しておくこと（未設定の場合、エラーとなる）。

**概略説明**

引数のカテゴリIDに紐づくデコメ絵文字のURI一覧を返却する。

サンプルコード:

```
IDecoEmojiManager mIDecoEmojiManagerIF = null;
ServiceConnection serviceConn = new ServiceConnection() {
  public void onServiceConnected(ComponentName componentName, IBinder iBinder) {
    mIDecoEmojiManagerIF = IDecoEmojiManager.Stub.asInterface(iBinder);
     :
  }
}
 :
Intent intentBindSearvice = new Intent(IDecoEmojiManager.class.getName());
bindService(intentBindSearvice, serviceConn, Context.BIND_AUTO_CREATE);
    :
List<DecoEmojiCategoryInfo> categoryInfoList = new
ArrayList<DecoEmojiCategoryInfo>();
List<String> decoEmojiUriList = new ArrayList<String>();
mIDecoEmojiManagerIF.aidl_getCategoryList(60, categoryInfoList); // 60 個まで
mIDecoEmojiManagerIF.aidl_getDecoUriList(categoryInfoList[0].getCategoryId(),
        0, decoEmojiUriList);
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.6.1.4. デコメ絵文字更新チェック

| API 名 | aidl_checkDecoEmoji |
|---|---|
| 説明 | 引数のデコメ絵文字IDより後に追加更新されたデコメ絵文字を抽出し、追加分のデコメ絵文字辞書更新情報をブロードキャスト送信する。 |

**引数**

| No | 引数名 | 型 | I/O | Must/Opt. | 説明 |
|---|---|---|---|---|---|
| 1 | decoemoji_id | int | I | Must | デコメ絵文字ID … IME側で最後に追加したデコメ絵文字ID<br>デコメ絵文字ID：-1(*1), 0～9,999 |
| 2 | ime_name | String | I | Must | IMEパッケージ名 … 呼出元IMEのパッケージ名 |
| | | | | | |

**返値**

| No | 型 | 説明 |
|---|---|---|
| 1 | int | IDecoEmojiConstant.OK：正常終了<br>IDecoEmojiConstant.NG_INITIALIZING：初期化処理中<br>IDecoEmojiConstant.NG_DIC_INFO：読み/品詞の取得中に問題が発生（DB エラーを含む）<br>IDecoEmojiConstant.NG_ID：「decoemoji_id」の値が範囲外<br>IDecoEmojiConstant.NG_IME_NAME：「ime_name」の値が null |

**前提**

DecoEmojiManager により初期データが構築されていること

**概略説明**

引数のデコメ絵文字IDより後に追加されたデコメ絵文字を抽出し、追加分のデコメ絵文字辞書操作情報をブロードキャスト送信する。

(*1) 全デコメ絵文字データの再取得処理を行う場合は、第一引数(decoemoji_id) に -1 を指定する。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.6.2. 利用クラス

パッケージ名 :jp.co.omronsoft.android.decoemojimanager_<carrier>.interfacedata

#### 14.6.2.1. DecoEmojiUriInfo クラス

| メソッド | 引数 | 内容 |
|---|---|---|
| DecoEmojiUriInfo() | なし | オブジェクトの生成 |
| DecoEmojiUriInfo(Parcel in) | in:Parcel 指定 | in から文字列を読み出し、URI として設定する |
| getUri() | なし | 保持する URI を返す |
| setUri(String uri) | uri：URI | URI を設定する |

#### 14.6.2.2. DecoEmojiCategoryInfo クラス

| メソッド | 引数 | 内容 |
|---|---|---|
| DecoEmojiCategoryInfo () | なし | オブジェクトの生成 |
| DecoEmojiCategoryInfo (Parcel in) | in:Parcel 指定 | in から id(int)、日本語カテゴリ名(文字列)、英語カテゴリ名（文字列）、ページ数(int)を読み出して設定する |
| int getCategoryId () | なし | 保持するカテゴリの ID を返す |
| String getCategoryName_jpn() | なし | 保持するカテゴリ名(日本語)を返す |
| String getCategoryName_eng() | なし | 保持するカテゴリ名(英語)を返す |
| int getPageCnt() | なし | 保持するページ数を返す |
| void setCategoryId(int category_id) | category_id: カテゴリ ID | カテゴリ ID を設定する |
| void setCategoryName_jpn(String category_name) | category_name：カテゴリ名 | カテゴリ名(日本語)を設定する |
| void setCategoryName_eng(String category_name) | category_name：カテゴリ名 | カテゴリ名(英語)を設定する |
| void setPageCnt(int pagecnt) | pagecnt ：ページ数 | ページ数を設定する |

#### 14.6.2.3. DecoEmojiAttrInfo クラス

| メソッド | 引数 | 内容 |
|---|---|---|
| DecoEmojiAttrInfo () | なし | オブジェクトの生成 |
| DecoEmojiAttrInfo(Parcel in) | in:Parcel 指定 | in から ID(int)、読み(文字列配列)、品詞（byte 配列）、表記(文字列配列)を読み出して設定する |
| int getId() | なし | ID を取得する |
| void setID(int id) | id：ID | ID を設定する |
| String getName(int idx) | idx:インデックス | idx 番目の読みを取得する |
| byte getPart(int idx) | idx:インデックス | idx 番目の品詞を取得する |
| void setName(int idx, String name) | idx:インデックス name： 読み | idx 番目の読みを設定する |
| void setPart(int idx, byte part) | idx:インデックス part： 品詞 | idx 番目の品詞を設定する |
| String getNote(int idx) | idx:インデックス | idx 番目の表記を取得する |
| void setNote(int idx, String note) | idx:インデックス note： 表記 | idx 番目の表記を設定する |



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.6.3. 定数

#### 14.6.3.1. DecoEmojiManager エラー値

| 名前 | 意味 | 値 |
|---|---|---|
| IDecoEmojiConstant.OK | 正常終了 | 0 |
| IDecoEmojiConstant.NG_INITIALIZING | エラー：初期データ構築中 | -1 |
| IDecoEmojiConstant.NG_DIC_INFO | エラー：読み/品詞の取得中に問題が発生した（DB エラーを含む） | -2 |
| IDecoEmojiConstant.NG_ID | エラー：「decoemoji_id」の値が範囲外 | -3 |
| IDecoEmojiConstant.NG_OPERATION | エラー：「flag」の値が範囲外 | -5 |
| IDecoEmojiConstant.NG_IME_NAME | エラー：「ime_info」の値が null | -6 |
| IDecoEmojiConstant.NG_URI_INFO | エラー：URI の取得中に問題が発生（DB エラーを含む） | -10 |
| IDecoEmojiConstant.NG_DIC_FULL | エラー：登録済の読み/品詞の件数が上限(20000)に達した | -11 |
| IDecoEmojiConstant.NG_ID_FULL | エラー：登録済のデコメ絵文字の件数が上限(10000)に達した | -12 |
| IDecoEmojiConstant.NG_NOT_REGISTERED_ID | エラー：登録されていない URI を指定 | -18 |
| IDecoEmojiConstant.NG_URI | エラー：「uri」の値が不正 | -19 |
| IDecoEmojiConstant.NG_REGISTERED_URI | エラー：登録済の URI を指定 | -20 |
| IDecoEmojiConstant.NG_TAG | エラー：引数の拡張タグ名が null | -21 |
| IDecoEmojiConstant.NG_TAG_INFO_NULL | エラー：引数の拡張情報リストが null | -22 |
| IDecoEmojiConstant.NG_TAG_INFO_READ | エラー：拡張情報取得中に問題が発生 | -23 |

#### 14.6.3.2. DecoEmojiManager ブロードキャスト関連

| 名前 | 意味 | 値 |
|---|---|---|
| IDecoEmojiConstant. BROADCAST_DATA_TAG | データ | "DecoEmojiInformation" |
| IDecoEmojiConstant. BROADCAST_ID_TAG | ID | “Id” |
| IDecoEmojiConstant. BROADCAST_TYPE_TAG | タイプ | “Type” |
| IDecoEmojiConstant.BROADCAST_TIMESTAMP_TAG | 更新タイムスタンプ | "TimeStamp" |
| IDecoEmojiConstant.ACTION_DECOEMOJI_RESULT | デコメ絵文字辞書更新情報 | "jp.co.omronsoft.android.decoemojimanager_<carrier>" |
| IDecoEmojiConstant.ACTION_CHANGE_DATABASE | デコメ絵文字情報更新情報 | "jp.co.omronsoft.android.decoemojimanager_<carrier>.CHANGE_DATABASE" |



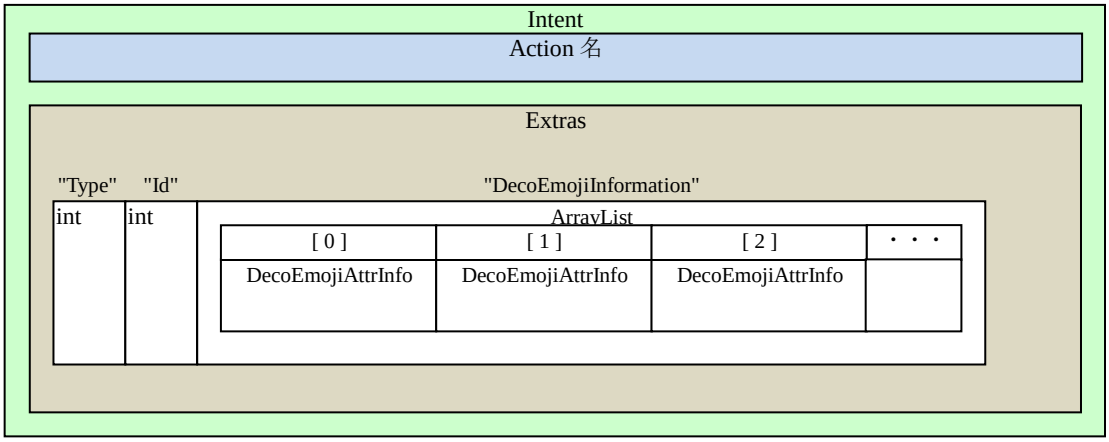
Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

### 14.6.4. DecoEmojiManager 発行インテント

DecoEmojiManager から送出される Intent（Broadcast）の詳細について説明する。

#### 14.6.4.1. デコメ絵文字辞書更新情報

Intent の構造




| データ名称 | | 入出力内容 |
|---|---|---|
| Action 名 | | ★デコメ絵文字更新チェック呼出に対する応答の場合<br>呼出元 IME のパッケージ名<br>★上記以外の場合<br>IDecoEmojiConstant.ACTION_DECOEMOJI_RESULT |
| Extras | Type | 更新区分。格納する区分は下記の通り。<br>追加:0　削除:2　全削除:4　更新データなし:9 |
| | Id | 受信済デコメ絵文字 ID。デコメ絵文字更新チェック呼出に対する応答時のみ有効な値を格納する。<br>★デコメ絵文字更新チェック呼出に対する応答の場合<br>デコメ絵文字更新チェック呼出時、引数「decoemoji_id」で渡されたデコメ絵文字 ID<br>★上記以外の場合<br>-1 固定 |
| | DecoEmojiInformation | DecoEmojiAttrInfo の ArrayList。<br>DecoEmojiAttrInfo には、更新対象デコメ絵文字 ID、読み一覧、品詞一覧が含まれる。 |

サンプルコード:

```
public void onReceive(final Context context, Intent intent) {
  Bundle bundle = intent.getExtras();
  int type = bundle.getInt(IDecoEmojiConstant.BROADCAST_TYPE_TAG);
  if (type != IDecoEmojiConstant.FLAG_FINISH) {
    List<DecoEmojiAttrInfo> receivedatalist =
            bundle.getParcelableArrayList(IDecoEmojiConstant.BROADCAST_DATA_TAG);
  }
}
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

#### 14.6.4.2. デコメ絵文字情報更新情報

| データ名称 | | 入出力内容 |
|---|---|---|
| Action 名 | | IDecoEmojiConstant.ACTION_CHANGE_DATABASE |
| Extras | IDecoEmojiConstant.<br>BROADCAST_TIMESTAMP_TAG | 更新画像のタイムスタンプデータ |

サンプルコード:

```
public void onReceive(final Context context, Intent intent) {
  // intent 識別子を取得
  String action = intent.getAction();
  if (action.equals(IDecoEmojiConstant.ACTION_CHANGE_DATABASE)) {
      // IntentBundle を取得する
      Bundle bundle = intent.getExtras();
      // 更新時の TimeStamp を取得する。
      String timestamp =
    bundle.getString(IDecoEmojiConstant.BROADCAST_TIMESTAMP_TAG);
  }
}
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

# 15. 【データ仕様】デコメ絵文字読み付加情報フォーマット

デコメ絵文字に付加する「読み付加情報」のデータフォーマットを以下に記載します。タグは将来拡張する可能性があります。絵文字パッケージでは、デコメ絵文字として用いられる GIF および JPEG ファイルにこれを格納することによって、デコメ絵文字がプリインストールまたはダウンロードされた Android 端末で、デコメ絵文字を IME の予測・変換で用いられるように自動的に登録します。読み付加情報は、画像ファイルの本来の利用を妨げない領域に書きこまれます。

◆付加情報定義フォーマット

<emoji>
<label><name>読み</name> [<part>品詞</part>][<note>表記</note>]</label>
[<label><name>読み</name> [<part>品詞</part>][<note>表記</note>]</label>]
[<category>カテゴリ</category>]
</emoji>

◆付加情報文字列の上限

上限は 2048bytes とする。

◆文字コード

文字コードは UTF-8 とする。

◆タグ格納例

<emoji><label><name>えがお</name><note>笑顔</note></label>
<label><name>わらう</name><part>動詞</part></label>
<category>感情</category></emoji>



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

◆付加情報要素一覧

| タグ名 | 項目 | 出現性<br>(個数) | 内容 |
|---|---|---|---|
| emoji | 絵文字定義 | 必須<br>(1) | 画像ファイルを絵文字として扱うための付加定義情報を記述する。<br>一つの画像ファイルに対し、emoji は 1 つだけ記述可能とする。<br>emoji の構成要素として、label のみ必須とする。<br>emoji には、複数の label を含めることができる。 |
| label | ラベル | 必須<br>(1-10) | 読み情報のセットをくくるためのタグ。絵文字は読み情報を複数持つことができる。（最大 10 個。）label 内で name は省略できない。 |
| name | 読み | 必須<br>(1) | 読み文字列を格納する。（24 文字まで）<br>読みは原則ひらがなのみで記述する。<br>辞書の読みとして漢字等を使用したい場合は、それらの文字を含めてもよい。(フォーマットとしては特に制限は設けない。)ただし、予測・変換などを行った場合の動作は IME に依存する。通常の IME であれば、読みはひらがなのみが有効である。 |
| part | 品詞 | 省略可<br>(0-1) | 品詞名を格納する。<br>日本語向け品詞一覧は別途定義する。<br>品詞を指定しない場合は part を省略可とする。<br>（品詞を指定することにより、IME がより適切な予測・変換を行うことを可能にする。） |
| note | 表記 | 省略可<br>(0-1) | 表記文字列を格納する。（24 文字まで）<br>表記を指定しない場合は note を省略可とする。<br>（IME が本タグで記述されている文字列を解析し、最適な品詞情報を取得し、自然なつながりを実現することを可能にする。） |
| category | カテゴリ<br>【未使用】 | 省略可<br>(0-n) | カテゴリを格納する。<br>1 つの emoji 内に複数のカテゴリを記述可能とする。<br>カテゴリを指定しない場合は category を省略可とする。<br>（Android 端末内の該当するカテゴリに分類することを可能にする。） |

◆読み付加情報フォーマット DTD(参考)

```
<!--  Decoration Emoji Character Data Extension (Version 1.0)  -->
<!DOCTYPE emoji [
  <!ELEMENT emoji (label+, category*) >
    <!ELEMENT label ( (name, part?, note?) | (name, note?, part?) ) >
      <!ELEMENT name (#PCDATA) >
      <!ELEMENT part (#PCDATA) >
      <!ELEMENT note (#PCDATA) >
    <!ELEMENT category (#PCDATA) >
] >
```



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

◆日本語向け品詞一覧

| 品詞_ID | 品詞＊ | 説明 |
|---|---|---|
| 1 | 名詞 | 普通名詞 |
| 2 | 名詞-固有名詞 | 固有名詞 |
| 3 | 名詞-代名詞 | いわゆる代名詞，不定語．<br>例：「それ」「ここ」「あいつ」「あなた」「みんな」「わたし」 |
| 4 | 名詞-副詞可能 | 曜日，月など時間を表す副詞的な用法を持つ名詞．量や割合などを表し副詞的に使うことのできる名詞．<br>例：「金曜」「一月」「午後」「少量」… |
| 5 | 名詞-サ変接続 | 後ろに「する」「できる」「なさる」「くださる」などが後接することができる名詞．<br>例：「インプット」「愛着」「悪化」「悪戦苦闘」「一安心」「下取り」… |
| 6 | 名詞-形容動詞語幹 | 形容動詞語幹で，「な」の前に現れるもの．<br>例：「健康」「安易」「駄目」「だめ」… |
| 7 | 名詞-ナイ形容詞語幹 | 助動詞の「ない」の直前に現れて形容詞的な働きをする語<br>例：「申し訳」「仕方」「とんでも」「違い」… |
| 8 | 名詞-数 | 漢数字，算用数字，および，「何（回）」「数（%）」「幾（夜）」．<br>例：「０」「１」「２」「何」「数」「幾」‥ |
| １１ | 名詞-接尾 | 一般に「接尾語」というよりも範囲が広く，複合名詞の後ろ要素として用いられることが普通なもの．<br>例：「君」「様」「町」「市」「県」「個」「つ」「本」「冊」「パーセント」「ｃｍ」「ｋｇ」「カ月」「時間」「時半」… |
| １２ | 名詞-接続詞的 | 単語と単語を接続する接続詞的な働きをするもの．<br>例：「（日本）対（アメリカ）」「対（アメリカ）」「（３）対（５）」「（女優）兼（主婦）」 |
| １４ | 接頭詞 | 名詞（形容動詞語幹を含む）に前接する接頭語，数に接続するもの．<br>動詞の命令形あるいは［動詞連用形］＋「なる／なさる／くださる」に前接する接頭語．<br>例：「お（水）」「某（氏）」「同（社）」「故（～氏）」「約」「およそ」「お（読みなさい）」「お（座り）」 |
| １５ | 動詞 | 動詞 |
| １６ | 形容詞 | 形容詞 |
| １７ | 副詞 | 必ず後ろで切れるもの，連体修飾が不可能な副詞．<br>例：「あいかわらず」「多分」など |
| １８ | 連体詞 | 名詞を修飾する形しかもたないもの．<br>例：「この」「その」「あの」「どの」「いわゆる」「なんらかの」「何らかの」「いろんな」「こういう」「微々たる」「堂々たる」「単なる」「いかなる」「我が」「同じ」「亡き」… |
| １９ | 接続詞 | 独立に現れる接続詞．<br>例：「が」「けれども」「そして」「じゃあ」「それどころか」… |
| ２０ | 感動詞 | 感動詞．あいさつなど．<br>例：「おはよう」「おはようございます」「こんにちは」「こんばんは」「ありがとう」「どうもありがとう」「ありがとうございます」「いただきます」「ごちそうさま」「さよなら」「さようなら」「はい」「いいえ」「ごめん」「ごめんなさい」… |
| ２１ | 記号 | 一般的な記号．<br>例：「○」「◎」「@」「$」「〒」「→」「＋」など |

＊オープンな品詞体系「IPADIC」を参考に抽出したもの。



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.

| 改定履歴 | | |
|---|---|---|
| V1.0.0 | 2011/10/17 | 新規作成 |
| V1.1.0 | 2012/3/27 | デコ絵文字更新通知機能追加<br>端末スリープ時のアニメーション停止の注意を追加<br>API 仕様を追加（インタフェース仕様書の参照を不要にした。） |
| V1.1.1 | 2012/4/10 | 「デコメ絵文字読み付加情報フォーマット」に<label>の出現数上限を追記 |
| | | |
| | | |



Copyright (C) OMRON SOFTWARE Co., Ltd. 2011 All Rights Reserved.