S!アプリ開発ガイド

[MIDP 2.0 対応端末編]

Version 1.0.6 / Aug.1,2008

ソフトバンク モバイル 株式会社

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

本書は情報提供を目的として作成されたものです。ソフトバンクモバイル株式会社は本書の記載内容に関して明示的にも、黙示的にも何ら保証するものではありません。

本書に記載されている事柄は、予告なしに変更する可能性があります。

本書の使用、または本書を使用した結果については、ユーザ各位がその責任を負うものとしますのでご了承ください。

1. ドキュメントの一部または全部を無断で改版、引用することを禁じます。
2. その他、著作権の範囲を超えて複製することを禁じます。
3. ドキュメントを運用した結果の影響については、いっさいの責任を負いかねますのでご了承ください。

[商標]

- Powered by JBlend™, (C)1997-2008 Aplix Corporation. All rights reserved.
- JBlend および JBlend に関する商標は、日本およびその他の国における株式会社アプリックスの登録商標または商標です。
- S!アプリ対応のソフトバンク携帯電話は、株式会社アプリックスが開発し、Java™アプリケーションの実行速度が速くなるように設定された JBlend ®を搭載しています。
- Java™および Java に関する商標は、米国およびその他の国における米国 Sun Microsystems, Inc.の登録商標または商標です。
- SMAF はヤマハ株式会社の登録商標または商標です。
- Macromedia ® Flash™、Flash Lite™、Flash は、Adobe System, Inc.の米国およびその他の国における登録商標または商標です。
- FeliCa はソニー株式会社の登録商標です。
- FeliCa はソニー株式会社が開発した非接触 IC カードの技術方式です。
- Bluetooth®は米国 Bluetooth SIG,Inc.の登録商標です。
- SOFTBANK およびソフトバンクの名称、ロゴは日本国およびその他の国におけるソフトバンク株式会社の登録商標または商標です。
- S!アプリ、S!カラオケ、MEXA はソフトバンクモバイル株式会社の登録商標または商標です。
- S!アプリは、Java™に対応したアプリケーションです。

その他、記載されている会社名、製品名は、各社の登録商標または商標です。

■修正履歴

Version	日付	内容
1.0.0	2006/10/1	新規文書
1.0.1	2007/1/26	2.1.4. 3D Sound API について 表 2.1.4-1 を新規追加 2.2.6. データサイズ 表 2.2.6-2 を新規追加 2.2.8.11. MIDxlet-ScreenSize WVGA に関する表示方法について追記 2.2.8.19. MIDxlet-Bluetooth API コール時のエラーについてい追記 2.3.4.1. 送信 表 2.3.4.1-1.を新規追加
1.0.2	2007/6/1	2.2.6 データサイズ 表 2.2.6.2 サウンドファイル対応コントロール一覧削除 2.2.7 データ利用 表 2.2.7-2 対応コントロール一覧新規追加
1.0.3	2007/10/5	2.1.1 CLDC 1.1 <JSR 139>について 表 2.1.1-1 System.getProperty()で取得可能なプロパティ値 追記 2.1.2 MIDP2.0 <JSR118>について 利用可能ファイルについて記載追加 2.2.3.1. 自動起動アプリ Bluetooth®起動 注意事項記載追加 SMS 起動 削除 2.2.5 データ格納域 表 2.2.5-1. データ格納域 追記 2.2.6 データサイズ 表 2.2.6-1. 利用可能データ体裁一覧 追記 2.2.7 データ利用 表 2.2.7-1. 各 API にて利用できるデータ 追記 2.2.8.9 MIDxlet-API 追記 2.3.2 セキュリティ 表 2.3.2-2. セキュリティダイアログ表示対象 API 一覧 Read User Data Access に追記 誤記載修正
1.0.4	2008/4/18	2.1.1. CLDC 1.1 <JSR 139>について 表 2.1.1-1. System.getProperty()で取得可能なプロパティ値に、以下の項目を追加 ・ 伝送速度種別取得対応の有無 2.2.3.2. ブラウザ起動アプリ codebase および、AMS-Filename に関する注意書きを追加 2.2.4. サイズ Record Store のサイズを 1MB から 3MB へ変更 2.2.8. MIDlet 属性 下記表内に記載の属性の MAX 値を変更 表 2.2.8-1. Jad ファイル ・ MIDlet-Name

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

		・ MIDlet-Vender ・ MIDlet-Description ・ MIDlet-Data-Size 表 2.2.8-2. Manifest ファイル ・ MIDlet-Name ・ MIDlet-Vender 2.3.5. Record Store 機能 Record Store 書き込み処理中の注意事項と、外部メモリカードへの保存に関する注意事項の追加 誤記載　修正
1.0.5	2008/4/18	2.1.1. CLDC 1.1 <JSR 139>について 表 2.1.1-1. System.getProperty()で取得可能なプロパティ値に、以下の項目を追加 ・ Obex 送信サイズ 2.3.2. セキュリティ 表 2.3.2-2.に「com.mexa.bluetooth」に関する情報を追加
1.0.6	2008/8/1	2.2.5. データ格納域 既存の記載を新規項版に伴い移行及び修正記載 2.2.6. データサイズ 既存の記載を新規項版に伴い移行及び修正記載 2.2.7. データ利用 既存の記載を新規項版に伴い移行及び修正記載 誤記載 修正

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

0. イントロダクション

0.1. 目的

本書は S!アプリを作成する際に必要な技術情報を開発者様（以下「CP 様」と表記）に提供するものである。

本書では MIDP 2.0 対応端末向けに提供を行なう S!アプリの作成について説明する。

0.2. 前提

本書は以下の技術について熟知していることを前提とする。

- Java™2
- J2ME
- CLDC 1.1：Connected Limited Device Configuration 1.1
- MIDP 2.0：Mobile Information Device Profile 2.0
- Location API：Location API for J2ME 1.0
- MMAPI 1.1：Mobile Media API 1.1
- WMA 1.1：Wireless Messaging API 1.1
- M3G：Mobile 3D Graphics API for J2ME

その他、弊社提供の各種開発ガイドを熟知していること。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

0.3. 参考文献

[CLDC 1.1]
http://www.jcp.org/en/jsr/detail?id=139

[CLDC 1.0]
http://www.jcp.org/en/jsr/detail?id=30

[MIDP 2.0]
http://www.jcp.org/en/jsr/detail?id=118

[MIDP 1.0]
http://www.jcp.org/en/jsr/detail?id=37

[Location API]
http://www.jcp.org/en/jsr/detail?id=179

[MMAPI 1.1]
http://www.jcp.org/en/jsr/detail?id=135

[WMA 1.1]
http://www.jcp.org/en/jsr/detail?id=120

[M3G]
http://www.jcp.org/en/jsr/detail?id=184

[GIF]
“Graphics Interchange Format Version 87a “, CompuServe
“Graphics Interchange Format Version 89a”, CompuServe

[MIDI]
“The Complete MIDI 1.0 Detailed Specification Version 96.1”, MMA

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

[SP-MIDI]
“Scalable Polyphony MIDI Specification Ver 1.0”, MMA, February 20, 2002
“SP-MIDI Device 5-24 Note Profile for 3GPP Ver 1.0”, MMA, February 15, 2002

[SVG]
“Scalable Vector Graphics(SVG) Specification Ver 1.1“, W3C
“Mobile SVG Profiles: SVG Tiny and SVG Basic“, W3C

[PNG]
“PNG (Portable Network Graphics) Specification, Version 1.2”, PNG Development Group, G.Randers-Pehrson, et. al., July 1999

[JNG]
“JNG (JPEG) Network Graphics Format Version 1.0”,

[WBMP]
“Wireless Application Environment Defined Media Type Specification Version 15-May-2001“, WAP Forum

[SMAF]
http://smaf-yamaha.com/jp/

[3D]
http://www.mascotcapsule.com/toolkit/softbank/

[Flash®]
http://www.macromedia.com/jp/

[MPEG4]
“ISO/IEC 14496-2 Information technology – Generic Coding of audio-visual objects – Part 2: Visual”, ISO/IEC
“ISO/IEC 14496-2 Information technology – Coding of audio-visual objects – Part 2: Visual AMENDMENT 1: Studio profile”, ISO/IEC

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

[MP4]
“ISO/IEC 14496-1 Information technology – Generic Coding of audio-visual objects – Part 1: System”, ISO/IEC
“ISO/IEC 14496-1 AMENDMENT1MPEG-4 : System Version2”, ISO/IEC

[HTTP]
“Hypertext Transfer Protocol – HTTP/1.1”, IETF, RFC2616, June 1999

[TS26234]
“3GPP TS26.234 Version 5.3.0”, 3GPP

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

0.4. 本書の構成

本書は以下の構成である。

1章 はじめに：各サービスの呼称について説明する。
2章 MIDP 2.0 対応端末：MIDP 2.0 対応端末向け S!アプリについて説明する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

1. はじめに

本書では、表のように各サービスを呼称する。

表記	内容
端末	S!アプリに対応した携帯電話
S!アプリサーバ	S!アプリを格納する弊社が指定する WWW サーバ
S!アプリ	ソフトバンク携帯電話で提供される Java™ サービスの名称およびアプリケーションの総称
ネイティブ	S!アプリを使用しない端末の機能
SMAF	端末用マルチメディアコンテンツ用データフォーマット Synthetic music Mobile Application Format の略
SMAF/MA2	音源 LSI MA2 での再生に対応した SMAF
SMAF/MA3	音源 LSI MA3 での再生に対応した SMAF
SMAF/MA5	音源 LSI MA5 での再生に対応した SMAF
SMAF/MA7	音源 LSI MA7 での再生に対応した SMAF
SMAF/Phrase	S!アプリ用に複数メロディの重畳再生に対応した SMAF
MIDlet	CLDC と MIDP のみを使用して動作するアプリケーション 以降では、各種 JSR および MEXA/JSCL を使用しているものも MIDlet と表記する。
JSCL	J-PHONE Specific Class Library、弊社が定義した拡張ライブラリ
MEXA（メキサ）	Mobile Entertainment eXtention API、JSCL をベースとして弊社が定義した拡張ライブラリ
JSCL-1.0.X 対応端末	System.getProperty(”microedition.profiles”)の取得する値が ”MIDP-1.0” である端末
JSCL-1.1.X 対応端末	System.getProperty(”microedition.profiles”)の取得する値が ”MIDP-1.0 JSCL-1.1.0” である端末
JSCL-1.2.1 対応端末	System.getProperty(”microedition.profiles”)の取得する値が ”MIDP-1.0 JSCL-1.2.0” である端末
JSCL-1.2.2 対応端末	System.getProperty(”microedition.profiles”)の取得する値が ”MIDP-1.0 JSCL-1.2.2” である端末
JSCL-1.3.2 対応端末	System.getProperty(”microedition.profiles”)の取得する値が ”MIDP-1.0 JSCL-1.3.2” である端末

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

MIDP 2.0 対応端末（X）	User-Agent にて取得する値が “Profile/MIDP-2.0 Configuration/CLDC-1.1” および MEXA/JSCL を搭載した端末
MIDP 2.0 対応端末（S）	User-Agent にて取得する値が “Profile/MIDP-2.0 Configuration/CLDC-1.1” である端末

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2. MIDP 2.0 対応端末

MIDP 2.0 対応端末は、各種 JSR および弊社が独自に拡張した仕様「MEXA」、「JSCL」を搭載している。さらに、一部端末においては、3D サウンド、モーションコントロールなどに対応する。以下に MIDP 2.0 対応端末の種類と定義を示す。

表 2-1. MIDP 2.0 対応端末の種類と定義

種類	定義
MIDP 2.0 対応端末（X）	「MIDP 2.0 を搭載」かつ 「MEXA/JSCL を搭載した端末」
MIDP 2.0 対応端末（S）	「MIDP 2.0 を搭載」かつ 「MEXA/JSCL を搭載しない端末」

本章では MIDP 2.0 対応端末にて搭載する各種 JSR、及び MIDP 2.0 対応端末向けの S!アプリについて説明をする。端末上での S!アプリの振る舞い、MEXA/JSCL の詳細については S!アプリ開発ガイドを参照のこと。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1. 機能説明

弊社 MIDP 2.0 対応端末では以下の仕様を搭載している。

（1） Connected Limited Device Configuration 1.1 (CLDC 1.1) <JSR 139>
（2） Mobile Information Device Profile 2.0 (MIDP 2.0) <JSR 118>

また、上記に加え、以下仕様を搭載する。各端末に搭載する仕様の詳細は S!アプリ開発ガイド「端末情報（各種）」を参照のこと。

（3）MEXA Core(JSCL)
※ 詳細については、S!アプリ開発ガイド「MEXA/JSCL 利用編」を参照のこと。
尚、本仕様は全ての MIDP 2.0 対応端末（X）にて搭載する。
（4）EnhancedGraphics API
（5）MEXA Unit1 (JSCL)
※ 詳細については、S!アプリ開発ガイド「MEXA/JSCL 利用編」を参照のこと。
（6）MEXA Unit2 (JSCL)
※ 詳細については、S!アプリ開発ガイド「MEXA/JSCL 利用編」を参照のこと。
（7）その他 MEXA Option (JSCL)
※ 詳細については、S!アプリ開発ガイド「MEXA/JSCL 利用編」を参照のこと。
（8）FeliCa Secure Client for Mobile API (FeliCa API 1.0)
或いは Mobile FeliCa Client for J2ME API(FeliCa API 2.0)
（9）3D Sound API ※VSCL の一部
（１０）Location API for J2ME (Location API) <JSR 179>
（１１）Mobile Media API 1.1 (MMAPI 1.1) <JSR 135>
（１２）Wireless Messaging API 1.1 (WMA 1.1) <JSR 120>
（１３）Mobile 3D Graphics API for J2ME (M3G) <JSR 184>
（１４）VSCL Core ※VSCL (Vodafone Specific Class Library)の一部
・Additional APIs
・Scalable Vector GraphicsAPI
（１５）Remote Control API ※VSCL の一部
（１６）Bar Code Recognition API ※VSCL の一部

以下に MIDP 2.0 対応端末に搭載している仕様のイメージを示す。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

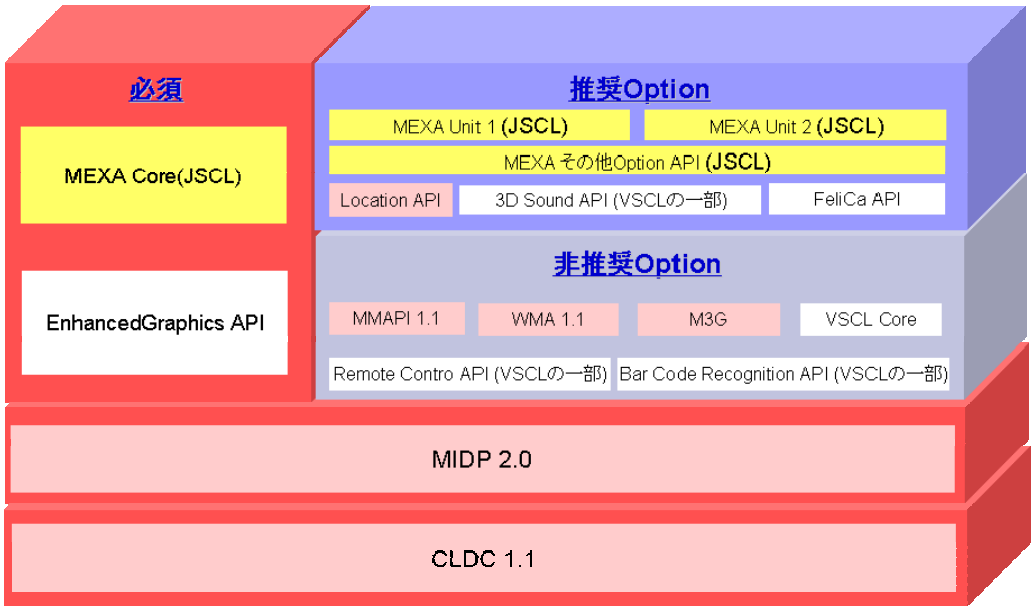


図 2.1-1. MIDP 2.0 対応端末(X)搭載仕様について

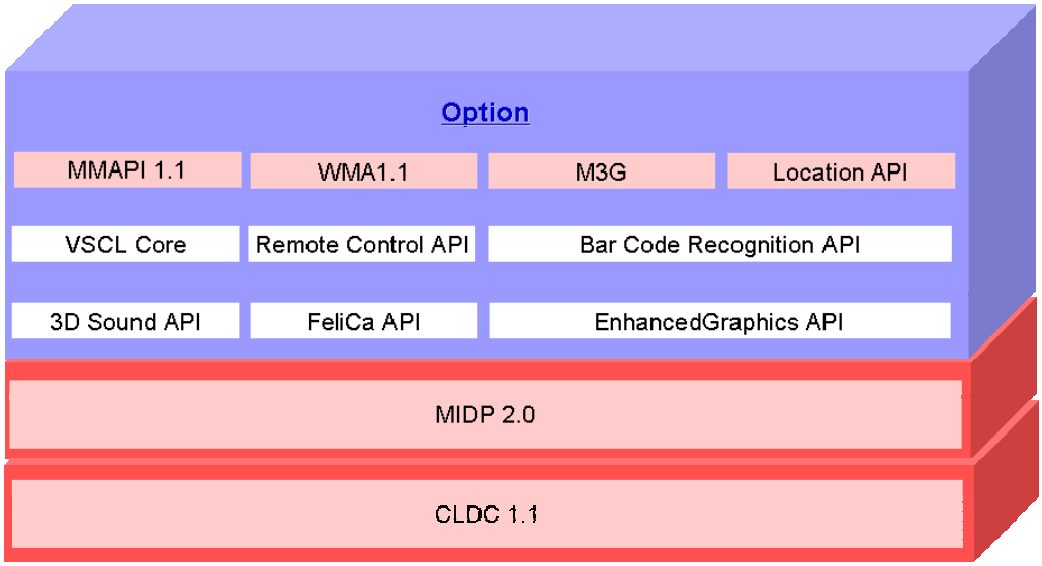


図 2.1-2. MIDP 2.0 対応端末(S)搭載仕様について

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1.1. CLDC 1.1 <JSR 139>について

CLDC 1.1 は CLDC 1.0 に改良を加えた仕様である。CLDC 1.1 では CLDC 1.0 で規定している機能の他、以下機能の拡張が追加されている。

- 浮動小数点サポート
- J2SE との互換性を高めた Calendar/Data/TimeZone クラス
- その他、各種ライブラリの改良

CLDC 1.1 の詳細については下記 URL を確認のこと。
http://www.jcp.org/en/jsr/detail?id=139

以下に弊社端末における特記事項について記載する。

＜クラス　System について＞
System.getProperty()メソッドによって端末のプロパティ値が取得可能である。
取得可能なプロパティ値を以下に示す。

表 2.1.1-1. System.getProperty()で取得可能なプロパティ値

機能	説明	getProperty()に指定するプロパティ名	値
マナーモード	マナーモードの設定状態を返す。	jscl.system.mannemode	“true” or “false”
		vscl.system.silentmode	ON：”1” OFF：”0”
オフラインモード	オフラインモードの設定状態を返す。	jscl.system.offlinemode	“true” or “false”
S!アプリ音量設定	「S!アプリ設定」中の音量を数値で返す。	jscl.system.javasetting.volume	“5”
		vscl.system.javasettingvolume	“5”
S!アプリバイブ設定	「S!アプリ設定」中のバイブ設定の設定値を数値で返す。	jscl.system.javasetting.vibration	ON：”1” OFF：”0”
		vscl.system.javasettingvibration	ON：”1” OFF：”0”
起動状態	S!アプリがどのように起動されたか返す。	jscl.system.wakeupmode	通常起動：”1” 待受アプリ：”2” ブラウザ起動:”4” S!アプリからのS!アプリ起動:”7” Bluetooth® からの自動起動：”10”

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

機能	説明	getProperty()に指定するプロパティ名	値
			時刻設定からの自動起動(RegisterAlam)：”12”
		vscl.system.wakeupmode	通常起動：”1” 待受アプリ：”2” ブラウザ起動:”3” S!アプリからのS!アプリ起動：”21” Bluetooth®からの自動起動：”24”
オフスクリーンの色深度	オフスクリーンのRGB別色深度を3桁の文字列で返す。	jscl.system.display.colordepth	例）565 R:5bit,G:6bit, B:5bit
背面液晶の有無	実装の有無を返す。	jscl.supports.subdisplay	“true” or “false”
背面液晶同時表示の有無	前面表示と背面液晶を同時に表示可能か返す。	jscl.supports.subdisplay.dualdraw	“true” or “false”
メモリカード対応有無	実装の有無を返す。	jscl.suppoets.external_storage	“true” or “false”
バーコード機能の有無	実装の有無を返す。およびサポートする読み取りフォーマットを返す。	jscl.supports.barcode	サポート無：“0” JAN 読み取りのみサポート：”1” QR 読み取りのみサポート：”2” JAN,QR 両方の読み取りサポート：”3”
IrDA 機能対応の有無	実装の有無を返す。	jscl.supports.irda	“true” or “false”
リモコン機能の有無	実装の有無を返す。	jscl.supports.remote_control	“true” or “false”
音声認識機能の有無	実装の有無を返す。	jscl.supports.voice_recognition	“true” or “false”
シリアル機能の有無	実装の有無を返す。	jscl.supports.serial	“true” or “false”
TV 機能のサポート有無	実装の有無を返す。	jscl.supports.tv	“true” or “false”
カラオケ機能の有無	実装の有無を返す。	jscl.supports.karaoke	“true” or “false”
モーションコントロール機能の有無	実装の有無を返す。	jscl.supports.msensor	“true” or “false”
S!アプリからのS!アプリ起動における起動元S!アプリ一時停止機能のサポート有無	実装の有無を返す。	jscl.supports.suspend_javaexecution	“true” or “false”
Bluetooth® S!アプリ設定	Bluetooth® S!アプリ設定の On/Off を返す。	jscl.system.btjavasetting	“true” or “false”
Bluetooth®公開設定の On/Off	公開設定の On/Off を返す。	jscl.system.btvisibilitysetting	“true” or “false”

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

機能	説明	getProperty()に指定するプロパティ名	値
端末側 Bluetooth® 設定	端末側 Bluetooth® 設定の On/Off を返す。	jscl.system.btswitchsetting	“true” or “false”
TV 予約機能の有無	実装の有無を返す。	jscl.supports.tv_reserve	非サポート：”0” 録画予約のみサポート：”1” 視聴予約のみサポート：”2” 録画・視聴予約サポート（予約データは録画・視聴は一括管理）：”3” 録画・視聴予約サポート（予約データは録画・視聴は個別管理）：”4”
EnhancedFEP Control の開始位置	開始位置のＹ値を返す	jscl.system.e-fep_startposition ただし、904SH のみ以下となる。 vscl.system.e-fep_startposition	例）190 Java 実行画面の(X,Y：0,190)の位置から下が FEP 画面の場合、190 を返す
履歴情報からの起動有無	履歴情報からの起動の有無を返す ※一時停止時は状態をリセットし、再開時に該当する状態を返す	mexa.system.push_history	“true” or “false” ※1
再開種別	S!アプリがどのように再開されたかの状態を返す。	mexa.system.resumemode	初期状態：”0” 通常再開：”1” ※2 Bluetooth® による再開要求：”10” 時刻設定による再開要求：”12”
伝送速度種別取得対応の有無	実装の有無を返す	mexa.supports.transmissionrate	“true” or “false”
赤外線送信サイズ	赤外線通信において送信可能な最大サイズを返す	mexa.supports.maxobexsize	送信最大サイズを byte 単位で返す

※　1　履歴情報からの S!アプリ起動における起動状態は通常起動となる。
※　2　ユーザ操作により再開、もしくは待受設定により自動再開された時に返す値

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1.2. MIDP 2.0 <JSR 118>について

MIDP 2.0 ではユーザインタフェース、ネットワーク、及びセキュリティについてサポートする。さらに以下機能についてもサポートする。

- ◆ GIF ファイルの利用
  GIF87a、GIF89a（Animated GIF は 1 フレーム目の静止画像）の表示が可能である。
- ◆ 画像の半透明効果（α合成）
  α値の範囲は最低 2 段階（透過/非透過）、最大 256 段階である。
  ※　α値の対応については端末実装依存となる。
- ◆ S!アプリからのブラウザ起動、音声発信機能
  ブラウザ起動(一部拡張あり)、音声発信はネイティブの機能を利用する。
- ◆ レコードストアの共有利用

MIDP 2.0 の詳細については下記 URL を確認のこと。
http://www.jcp.org/en/jsr/detail?id=118

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

以下に弊社端末において追加として実装する機能を説明する。

◆　S!アプリからのネイティブブラウザ起動

S!アプリからネイティブブラウザを起動することが可能である。本機能はplatformRequest()のスキームにて以下を指定することにより利用する。

表 2.1.2-1.　platformRequest()に指定するスキーム

	起動スキーム	動作
ブラウザ	http://, https://	端末のブラウザを起動させる
PC ブラウザ	pcb-http://, pcb-https://	端末の PC ブラウザを起動させる

本機能について、以下の制限事項がある。

- 拡張ヘッダは付加されない。
- 弊社固有のクエリは利用禁止とする。
- Https 通信は、End-to-End のみ可能である。

◆　S!アプリからのネイティブメーラー起動

S!アプリからネイティブのメーラー機能を起動することが可能である。本機能はplatformRequest() のスキームにて「mailto:<メールアドレス>?<header1>=<value1>&<header2>=<value2>」を設定することにより利用する。指定可能なヘッダー（header1、header2）は、subject、to、cc、bcc、bodyである。ただし、宛先を示すヘッダがネイティブのメーラーで指定可能な宛先より多く設定されている場合、指定可能な宛先分のみを有効とする。

（例）

mailto:test1@softbank.ne.jp?cc=test2@softbank.ne.jp&subject=abc&body=xyz

・送信表示内容

To:test1@softbank.ne.jp
Cc:test2@softbank.ne.jp
Subject: abc
Body: xyz

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

◆　S!アプリからの S!アプリ起動

S!アプリから他の S!アプリを起動することが可能である。S!アプリからの S!アプリ起動において、起動元 S!アプリを「主アプリ」、起動した S!アプリを「従アプリ」とする。一部端末では従アプリを起動した際、主アプリを一時停止することが可能である。主アプリが一時停止した場合、従アプリ終了時、主アプリを自動的に再開する。

本機能は主アプリにおいて、以下設定を行うことにより利用する。

1.　特殊スキームの設定

platformRequest()のスキームにて「v-app:<MIDlet-Name>|<MIDlet-Vendor>|<MIDlet-Version>|<主アプリ制御パラメータ>&<name1>=<value1>&<name2>=<value2>」を設定する必要がある。

2.　MIDlet 属性の設定（主アプリを一時停止する場合）

主アプリにおいて利用するヒープサイズを MIDxlet-Execution-Heap に指定する必要がある。MIDxlet-Execution-Heap の詳細については、2.2.8. MIDlet 属性を参照のこと。

ただし、従アプリが以下に該当する S!アプリである場合、主アプリからの起動を行なうことができない。

- 外部メモリカードに保存している S!アプリ
  - ※　端末内に保存されている S!アプリのみ起動を行い、外部メモリカードに保存されている S!アプリの起動は行わない。
- MIDlet 属性に「MIDxlet-Java-Execution: N」と記載した S!アプリ
- 主アプリの署名/証明書より上位の署名/証明書を付与した S!アプリ
  - ※　起動が可能な S!アプリは以下のとおりである。
  - ◆　主アプリ：Trusted Domain (Third Party Domain)
    従アプリ：Trusted Domain (Third Party Domain)、Untrusted Domain
  - ◆　主アプリ：Untrusted Domain
    従アプリ：Untrusted Domain

本章では、1.　特殊スキームの設定について説明を行う。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

＜特殊スキームの設定＞
S!アプリからの S!アプリ起動を利用するには、platformRequest()のスキームにて「v-app:<MIDlet-Name>|<MIDlet-Vendor>|<MIDlet-Version>※1|<主アプリ制御パラメータ>※2&<name1>=<value1>&<name2>=<value2>」を設定する必要がある。
また、主アプリで指定したパラメータ（name1、name2）を従アプリに受け渡すことが可能である。ただし、受け渡し可能なパラメータは以下を満たす必要がある。
・受け渡し可能なパラメータは 16 個まで指定可能とする。
　ただし、スキームの最大サイズを超過する場合はこの限りではない。
・受け渡しを行うパラメータ（name、value）のサイズはそれぞれ 128bytes までとする。

※1　MIDlet-Version を省略し、バージョンを問わない起動を行なうことが可能
※2　主アプリ制御パラメータには、以下値を指定

表 2.1.2-2.　主アプリ制御パラメータ

主アプリ制御 パラメータの指定値	説明
1	従アプリが起動した際、主アプリを終了する。
2	従アプリが起動した際、主アプリを一時停止する。

（例）
v-app:testApp|SoftBank|1.0|1&point=100&stage=3

・従アプリ
　MIDlet-Name: testApp
　MIDlet-Vendor: SoftBank
　MIDlet-Version: 1.0

・主アプリ動作
　従アプリが起動した際、終了する。

・受け渡しパラメータ
　name1,value1：point,100
　name2,value2：stage,3

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

また、MIDP2.0 において利用できるファイルは以下のとおりである。

表 2.1.2-3. 対応ファイルコントロール(音曲ファイル／動画ファイル)

ファイル種別	ファイルフォーマット	対応コントロール
音曲ファイル	AMR (NB)	VolumeControl
	Interactive MIDI	VolumeControl
	SMF(MIDI)	VolumeControl
	SP-MIDI	VolumeControl
	PCM	VolumeControl
	Tone Sequence	ToneControl VolumeControl
	Smaf	VolumeControl
	Phrase	VolumeControl
動画ファイル	3GPP	VolumeControl
	MP4	VolumeControl

表 2.1.2-4. 対応ファイルコントロール(画像ファイル)

ファイルフォーマット	拡張子	対応コントロール
GIF 87a	.gif	VideoControl GUIControl
GIF 89a	.gif	VideoControl GUIControl
Animated GIF	.gif	RateControl VideoControl GUIControl

※ Animated GIF に関しては、1 フレーム目の静止画像を表示する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1.3. EnhancedGraphics API について

EnhancedGraphics API とは弊社が独自に規定した API である。
EnhancedGraphics API では S!アプリでの描画をサポートする機能が含まれる。

- fillPolygon
  配列で指定された座標を頂点とする多角形を塗りつぶす機能
- drawPolyline
  配列で指定された座標を頂点とする点を結ぶ線分（多折線）を描画する機能
- getAntiAliasMode
  多角形と多折線のアンチエイリアス処理の描画モードを取得する機能
- setAntiAliasMode
  多角形と多折線のアンチエイリアス処理の描画モードを設定する機能

以下に EnhancedGraphics API の各バージョンに含まれる API を記載する。

表 2.1.3-1. EnhancedGraphics API

API	Ver 1.0	Ver 2.0
drawPolyline(int[] xPoints, int[] yPoints, int count)	○	○
drawPolyline(int[] xPoints, int[] yPoints, int offset, int count)	○	○
fillPolygon(int[] xPoints, int[] yPoints, int count)	○	○
fillPolygon(int[] xPoints, int[] yPoints, int offset, int count)	○	○
getAntiAliasMode()	－	○
getAntiAliasMode(Boolean antialias)	－	○

※　fillPolygon()の描画方式は「even-odd-rule」となる。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1.4. 3D Sound API について

3D Sound API では、ステレオヘッドフォンまたはステレオスピーカーを利用し3Dポジショナルオーディオを実現する。また、ドップラー効果およびリバーブ効果により、音楽再生時の表現を向上することが可能である。本機能は一部端末にて搭載している。以下に主な機能について説明する。

◆ **3D ポジショナルオーディオ**

プレイヤーとリスナーの位置/リスナーの向きの指定/プレイヤーとリスナーの位置関係(絶対/相対)指定により、複数の音源をリスナーの頭の周辺に配置し、リアルタイムで移動可能。

◆ **ドップラー効果**

プレイヤーとリスナーの速度の指定により音程をコントロール可能。

◆ **リバーブ効果**

空間の響きをコントロール可能。(洞窟/コンサートホール/水中 等)

＜特記事項＞

- SMAF、SMF、WAVE、Tone Sequenceの音曲に対し適用可能。
- 最大４つの処理ソース(チャンネル)を利用可能。
- SMAFに関しては3D Soundの情報を埋め込むことが可能。（この場合、S!アプリからの3D Sound制御は不可）
- MEXA/JSCLにより規定されたPlayerに対しての制御はできない。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

また、本機能にて利用できる楽曲ファイルは以下のとおりである。

表 2.1.4-1. 3D Sound Control 対応ファイル一覧

ファイルフォーマット	利用可否
MIDI(SMF)	○
SP-MIDI	○
SMAF	○
SMAF(SMAF/Phrase)	△
3GPP	△
WAVE(PCM)	○
Tone Sequence	○
AMR NB	△
MP4	△

［凡例］

○　：利用可能

△　：対応端末のみ利用可能

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1.5. MMAPI <JSR 135>について

MMAPI 1.1 ではより豊富なメディア（画像・音曲）を S!アプリで利用できる仕様を規定する。

MMAPI 1.1 の詳細については下記 URL を確認のこと。
http://www.jcp.org/en/jsr/detail?id=135

以下に弊社端末における特記事項について記載する。

＜インタフェース　Control について＞
S!アプリで扱うファイルフォーマットによって、利用できる機能が異なる。
MIDP 2.0 対応端末向け S!アプリで扱うファイルフォーマット、及び各ファイルフォーマットで利用できる機能については S!アプリ開発ガイド「端末情報（各種）」を参照のこと。尚、MIDP 2.0 対応端末向け S!アプリではストリーミングの再生はできない。

＜カメラ機能の利用について＞
MIDP 2.0 対応端末での MMAPI 1.1 を用いたカメラ機能の利用について説明する。尚、本機能は一部端末での実装となる。

◆ ファインダー
S!アプリ上のファインダーサイズがネイティブのファインダーサイズより小さい場合、表示サイズ分を切り取り表示する。
S!アプリ上のファインダーサイズがネイティブのファインダーサイズより大きい場合、ネイティブのファインダー以外の部分は「白」で表示する。

◆ キャプチャ
キャプチャ処理実行時のシャッター音の鳴音については、ネイティブの設定となる。また、キャプチャ範囲は、S!アプリ上に表示しているファインダー部分のみとなる。尚、キャプチャ画像のエンコード形式は PNG、JPEG となる。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

＜プレイヤーについて＞
MMAPI 1.1 で利用するプレイヤーには以下の 5 つの状態があり、S!アプリにて適宜管理を行う。
UNREALIZED、REALIZED、PREFETCHED、STARTED、CLOSED

尚、カメラ用プレイヤーを利用する場合の注意点を以下に記載する。

- カメラ用プレイヤーとその他のプレイヤーは同時に生成できない。
- カメラ用プレイヤー生成時のパラメータは、width と height のみ対応となり、video_enc と fps には対応しない。
- width もしくは height にネイティブの最大サイズ以上の値を指定した場合、Exception を返す。
- パラメータを指定しない場合には、ネイティブで対応している最大サイズとなる。

2.1.6. WMA 1.1 <JSR 120>について

WMA 1.1 は S!アプリにおける SMS 送信、受信の機能をサポートする。

WMA 1.1 の詳細については下記 URL を確認のこと。
http://www.jcp.org/en/jsr/detail?id=120

以下に弊社端末における特記事項について記載する。

＜通信プロトコルについて＞
MIDP 2.0 対応端末向け S!アプリにて利用できる通信プロトコルは、SMS である。

＜メッセージの送信＞
S!アプリから送信するメッセージには以下制限がある。
- メッセージをネイティブのメールボックスに保存しない。
- MIDP 2.0 対応端末の設定に関わらず配信確認をしない。

また、以下の状態で送信を行うと MIDP 2.0 対応端末はエラーを返す。
- 送信可能な長さを超えるメッセージの送信

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

＜メッセージの受信＞
MIDP 2.0 対応端末ではメッセージの受信時、メッセージ内のポート番号の有無によって以下の動作を行う。また、受信したメッセージが連結メッセージの場合、メッセージ連結後に以下の動作を行う。

- ◆ メッセージにポート番号を含む場合
  メッセージ受信時、ユーザに受信通知を行わない。受信したメッセージはネイティブのメールボックス以外の場所に一時的に保存し、ネイティブから確認することはできない。
- ◆ メッセージにポート番号を含まない場合
  メッセージ受信時、ユーザに受信通知を行う。受信したメッセージはネイティブのメールボックスに通常のメッセージとして保存する。

＜メッセージの取得＞
ポート番号を含むメッセージの取得は S!アプリで指定したポート番号と同じポート番号のメッセージのうち、最も古いメッセージを取得する。また、S!アプリで取得したメッセージは削除となる。

2.1.7. M3G <JSR 184>について

M3G はモバイル環境における 3D 描画アプリを可能とする、3D グラフィックス描画、アニメーションの実行などの機能をサポートする。

M3G の詳細については下記 URL を確認のこと。
http://www.jcp.org/en/jsr/detail?id=184

また、M3G ファイルの作成に関しては株式会社エイチアイ様のホームページ掲載のツールを参照すること。
http://www.mascotcapsule.com/toolkit/softbank/

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1.8. VSCL Core について

VSCL を一部端末において採用する。VSCL では既存の標準仕様では規定されていない独自の API について規定している。本章では VSCL について説明をする。

2.1.8.1. Additional APIs

VSCL では以下の機能を MEXA/JSCL より移植する。ただし、VSCL と MEXA/JSCL には機能の違いがある。違いについては Appendix B. VSCL と MEXA/JSCL の違いを参照のこと。

- リスナーインタフェース
  （例　インタフェース　BodyOpenListener）
- 画像エンコードクラス
  （例　クラス　ImageEncoder）
- 待受アプリ用基底クラス
  （例　クラス　ResidentMIDlet）
- デバイス制御用クラス
  （例　クラス　DeviceControl）

2.1.8.2. SVG API

SVG Tiny1.1 に準拠した 2D ベクトルグラフィックス描画機能を実現する。
MIDP 2.0 対応端末向け S!アプリから可能な操作を以下に示す。

- ビューポート（幅)、(高さ）の取得と変更
- パン（スクロール）
- パンの初期化
- 回転（度指定）
- 座標系の相互変換（ビューポイント座標⇔ユーザスペース座標）
- 拡大/縮小
- Graphics オブジェクトへの描画

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.1.9. Remote Control API

S!アプリにて、赤外線リモートコントロールを実現するための機能である。このAPIにより、リモートコントローラ対応機器を制御する赤外線コマンドの転送が可能となる。本機能はMEXA/JSCLより移植したものである。赤外線リモコンに対応した端末にて利用が可能である。

2.1.10. Bar Code Recognition API

紙面上などに印刷されたバーコードを読み取り、解析結果をS!アプリに渡すことができる機能である。サポートするバーコード形式を以下に示す。

表 2.1.10-1. 対応バーコード形式

種類	Type	対応形式	対応
1次元	EAN	EAN-13、EAN-8	△
1次元	JAN	JAN-13、JAN-8	△
1次元	UPC	UPC-A、UPC-E	△
2次元	DataMatrix	ECC-200、72 x 72 以上	△
2次元	QR	Model 2 Version1～10	○

［凡例］
○：バーコード機能に対応した全ての端末にて取得可能
△：バーコード機能に対応した一部端末にて取得可能

※ バーコード機能、及び上記バーコードに対応した端末にて利用が可能である。
※ MEXA/JSCLでは連続読み取りを利用できるが、VSCLでは連続読み取りを利用できない。
※ MEXA/JSCLでは対応バーコード形式はJANとQRのみである。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2. S!アプリの構成

2.2.1. 利用 API による S!アプリの種類

MIDP 2.0 対応端末向け S!アプリには利用する API により、以下のような S!アプリがある。

＜xx2/xx3、904T＞

◆ MEXA/JSCL 利用 S!アプリ（MIDP 2.0 対応端末向け）
利用可能プロファイル　：CLDC 1.1、MIDP 2.0、3D Sound API、MEXA/JSCL

◆ JSR 利用 S!アプリ
利用可能プロファイル　：CLDC 1.1、MIDP 2.0、WMA 1.1、MMAPI 1.1、M3G、VSCL、3D Sound API

※ MEXA/JSCL と WMA 1.1、MMAPI 1.1、M3G、VSCL の併用は不可能である。

＜xx4 以降＞

◆ MEXA/JSCL 利用 S!アプリ（MIDP 2.0 対応端末向け）
・・・MEXA/JSCL を利用した S!アプリ

◆ JSR 利用 S!アプリ
・・・MEXA/JSCL を利用していない S!アプリ

※ MEXA/JSCL と各 JSR および VSCL の併用が可能。ただし、同じ機能（Graphics3D（MEXA/JSCL)と M3G など）を利用する場合、MIDlet 属性「MIDxlet-API」の記載によって動作が異なる。動作差異については 2.2.8.9. MIDxlet-API を参照のこと。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.2. S!アプリの動作

MIDP 2.0 対応端末では以下の 5 パターンの S!アプリをダウンロードすることができる。

① MEXA/JSCL 利用 S!アプリ（MIDP 2.0 対応端末向け）
MIDP 2.0 対応端末（X）のみで動作する。

② JSR 利用 S!アプリ
MIDP 2.0 対応端末（S）、MIDP 2.0 対応端末（X）で動作する。

③ MIDP 2.0 利用 S!アプリ
－CLDC 1.1 、MIDP 2.0 のみを利用した S!アプリ－
MIDP 2.0 対応端末（S）、MIDP 2.0 対応端末（X）で動作する。

④ MEXA/JSCL 利用 S!アプリ（MIDP 1.0 対応端末向け）
－CLDC 1.0、MIDP 1.0、JSCL を利用した S!アプリ－
MIDP1.0 対応端末、MIDP 2.0 対応端末（X）で動作する。

⑤ MIDP 1.0 利用 S!アプリ
－CLDC 1.0、MIDP 1.0 のみを利用した S!アプリ－
MIDP1.0 対応端末、MIDP 2.0 対応端末（S）、MIDP 2.0 対応端末（X）で動作する。
ただし、MIDP 2.0 対応端末において動作をさせるためには、
セキュリティダイアログ表示対象 API を利用していない場合にも MIDlet-Permissions を記載する必要がある。
詳細は 2.2.8.5. MIDlet-Permissions を参照のこと。

以下に弊社端末における上記 S!アプリの動作について示す。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

表 2.2.2-1. S!アプリの動作

	端末		
	MIDP 2.0 対応端末（X）	MIDP 2.0 対応端末（S）	既存端末（MIDP1.0 対応端末）
①MEXA/JSCL 利用（MIDP 2.0 対応端末向け）	◎	×	×
②JSR 利用	◎	◎	×
③MIDP2.0 利用	◎	◎	×
④MEXA/JSCL 利用（MIDP 1.0 対応端末向け）	○	×	◎
⑤MIDP1.0 利用	○※1	○※1	◎

［凡例］

◎：そのまま動作可能

○：属性を修正すれば動作可能（詳細は 2.4.8.MIDlet 属性の変更を参照のこと）

×：動作不可能

※1　属性の修正に加え、MIDlet-Permissions を追加記載する必要がある。
詳細は 2.2.8.5. MIDlet-Permissions を参照のこと。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.3. S!アプリの種類

S!アプリには通常のS!アプリ(インストールアプリ)と待受アプリの2種類ある。MIDP 2.0対応端末では上記2種類に加えて、自動起動アプリ、ブラウザ起動アプリの2種類を追加した。以下に自動起動アプリ、ブラウザ起動アプリについて説明する。

2.2.3.1. 自動起動アプリ

自動起動アプリとは S!アプリライブラリから選択/起動するのではなく、イベントにより自動的に起動する S!アプリである。自動起動アプリには時刻設定起動、Bluetooth®起動の3種類がある。本機能に対応した端末にて自動起動アプリを利用することができる。ただし、各1つの登録および動作を保障する。

以下に時刻設定起動、Bluetooth®起動について説明する。

◆ 時刻設定起動

S!アプリ（MIDlet Suite）から指定した時間に指定したMIDletを起動する機能である。本機能はMIDP 2.0にて規定しているPushRegistry.registerAlarm()を利用し実現する。ただし、設定できる時間は365日（31536000000ms）とする。また、1分（60000ms）単位で設定した動作についてのみ保証する。

◆ Bluetooth®起動

Bluetooth®起動とは通信先端末（B）<サーバ>がS!アプリを起動していない状態において、通信元端末（A）<クライアント>からBluetooth®起動要求を送出し、通信先端末（B）<サーバ>のBluetooth® S!アプリを起動する機能である。

※ 本機能はオフィシャルコンテンツでのみ利用可能となる。

以下にBluetooth®起動を行うためのサーバ側の設定、およびクライアント側の設定を説明する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

【サーバ側】

MIDlet 属性（MIDlet-Push-<n>）でプロトコルとして、Bluetooth®を指定した S!アプリは、ダウンロード時に Bluetooth®起動の S!アプリとして登録し、保存する。MIDlet 属性にて Bluetooth®を指定する方法は以下のとおりである。

MIDlet-Push-<n>:
bluetooth://:<Application ID>,<MIDletClassName>,<AllowedSender>

＜各パラメータの説明＞

・Application ID
Bluetooth®起動を行なう S!アプリの ID。12 桁の ASCII にて指定する。ID の体系は以下のとおりである。

000000	XXXX	YY
固定値	<CP 様 ID>	任意

・MIDletClassName
Bluetooth®起動を行なう S!アプリのクラス名を指定する。

・AllowedSender
Bluetooth®起動を許可する相手の Bluetooth アドレス。
（＊により任意アドレスとすることが可能）

「受信した ID と S!アプリの ID（Application ID）が一致」、「送信先<クライアント>の Bluetooth アドレスがフィルタ文字列（AllowedSender）による判定にパス」した場合、S!アプリが Bluetooth®起動する。

＜注意事項＞

- S!アプリ（A）のダウンロード時に Bluetooth®起動の S!アプリとして登録する際、すでに別の S!アプリ（B）が同一<Application ID>を登録している場合、端末は S!アプリ（A）のダウンロードはしない。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

【クライアント側】
クライアント側の S!アプリより以下に示す Bluetooth®起動要求送信フォーマットをサーバ側に送出することにより、Bluetooth®起動を行なう。

Bletooth®起動要求送信フォーマット
Java|<ApplicationID>|<Sender>|<Comment>

＜各パラメータの説明＞
・Application ID
Bluetooth®起動を行なう S!アプリの ID。12 桁の ASCII にて指定する。ID の体系は以下のとおりである。
000000　XXXX　YY
固定値　<CP 様 ID>　任意
・Sender
UTF-8 にて最大 10 文字までの送信者を指定する。送信者は Bluetooth®起動時の確認画面において表示を行なう。
・Comment
UTF-8 にて最大 30 文字までのコメントを指定する。コメントは Bluetooth®起動時の確認画面において表示を行なう。

クライアント側 S!アプリよりサービス登録時に送出した Application ID とサーバ側 S!アプリの Jad ファイルに指定した ID が一致する S!アプリを起動する。

＜注意事項＞
- S!アプリを外部メモリに保存している場合、Bluetooth®起動要求を受けても起動することはできない。
- サーバ側端末より受信した Bluetooth®起動要求をユーザーが任意で拒否した場合、Bluetooth®起動要求を送信した端末からの Bluetooth 起動要求を一定時間無視する。Bluetooth®起動要求を送信した端末以外からの要求に対しては応答可能である。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.3.2. ブラウザ起動アプリ

ブラウザ起動アプリとは Web ページの object 要素に指定し、Web ページから起動する S!アプリである。本機能に対応した一部端末のみブラウザ起動アプリを利用することができる。Web ページの指定方法を以下に説明する。

- object 要素
  id　　：a 要素と関連付けを行なう ID を指定する。
  type　　：「text/vnd.sun.j2me.app-descriptor」を指定する。
  declare　：S!アプリはインライン表示不可能であるため、必ず「declare」を指定する。
  classid　：「x-oma-application:java-ams」を指定する。
  codebase：Jad ファイルの基準 URL（スキーム名、ホスト名）を指定する。
  ※尚、本値については、後述の param 要素「AMS-Filename」と合わせて、256bytes 以内に収めることを推奨する。

- param 要素
  name　：詳細は表 2.2.3.2-1.を参照のこと
  value　：詳細は表 2.2.3.2-1.を参照のこと

- a 要素
  href　　：object 要素と関連付けを行なう ID を指定する。

表 2.2.3.2-1. param 要素

name	value	説明
AMS-Filename	（Jad ファイル名）	起動を行なう S!アプリの Jad ファイル名および Jad ファイルの URL におけるクエリ部分を記載する。
MIDlet-Name	（アプリ名）	S!アプリの MIDlet-Name に指定した値を記載する。
MIDlet-Vendor	（ベンダー名）	S!アプリの MIDlet-Vendor に指定した値を記載する。
MIDlet-Version	（バージョン）	S!アプリの MIDlet-Version に指定した値を記載する。
AMS-Startup	auto	該当 S!アプリを端末に保存している場合、自動的に S!アプリを起動する。保存していない場合、ダウンロードを行う。ダウンロード後、自動的に起動する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

	launch-only	該当 S!アプリを端末に保存している場合のみ、S!アプリの起動を行い、ダウンロードは行わない。
	download-confirm	該当 S!アプリを端末に保存している場合、起動するか否かはユーザの選択に従う。保存していない場合、ダウンロードを行う。

また、Web ページで指定したパラメータを VSCL 及び MEXA/JSCL 内 DeviceControl#getWakeupParam()にて取得することが可能である。
ただし、受け渡し可能なパラメータは以下を満たす必要がある。

・受け渡し可能なパラメータは 10 個まで指定可能とする。
・受け渡しを行うパラメータ（name、value）のサイズはそれぞれ UTF-8 換算で 256bytes までとする。

以下に Web ページの指定方法について、（例）を示す。

(例)

```
<html>
  :
 <body>
  <div>
   <object id = “test” type = “text/vnd.sun.j2me.app-descriptor” declare = “declare”
    classid = “x-oma-application:java-ams”
    codebase = “http://www.test.com/” >
    <param name = “AMS-Filename”
                        value = “xxx.jad?id=xxx&id2=xxx&id3=xxx” />
    <param name = “MIDlet-Name” value = “sample” />
    <param name = “MIDlet-Vendor” value = “SoftBank” />
    <param name = “MIDlet-Version” value = “1.0” />
    <param name = “AMS-Startup” value = “auto” />
   </object>

   <a href = “#test”>ブラウザ起動アプリ</a>
  </div>
 </body>
</html>
```

※ 「xxx」には CP 様任意の値を記載

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.4. サイズ

MIDP 2.0 対応端末向け S!アプリのサイズを以下に説明する。

S!アプリは JAD ファイル、JAR ファイル、RecordStore の 3 つの要素で構成される。JAD ファイル、JAR ファイル、RecordStore それぞれにおいてサイズ制限がある。JAD ファイルは、**6KBytes** 以下、JAR ファイルは **1MBytes** 以下、RecordStore は **3Mbytes**(一部端末を除く)以下とする。JAD ファイル、JAR ファイル、RecordStore の合計サイズを **4MBytes** 以下とする。

尚、本書において 1Kbytes = 1024Bytes、1MBytes = 1024KBytes とする。

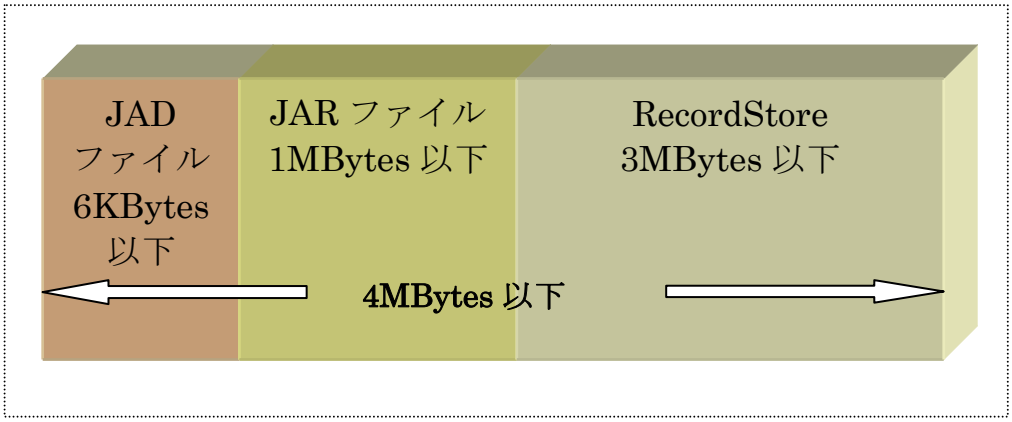


図 2.2.4-1. MIDP 2.0 対応端末における S!アプリのサイズ

※　ただし、ダウンロード時に付加するヘッダー等を含み 4MBytes とする。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5. MIDlet 属性

MIDP 2.0 搭載に伴う追加及び変更した MIDlet 属性について説明する。
下表でマスクされた属性は、新規に追加或いは変更となった MIDlet 属性である。
尚、Jad ファイル・Manifest ファイルは Unicode(UTF-8)で示したテキストファイルである。

表 2.2.8-1. Jad ファイル

必須属性	説明
MIDlet-Name	S!アプリ名 （文字列　MAX:64Bytes）
MIDlet-Version	S!アプリのバージョン番号 （数値）
MIDlet-Vendor	S!アプリのベンダー名 （文字列　MAX:64Bytes）
MIDlet-Jar-URL	JAR ファイルを指す URL （URL　MAX:1024Bytes）
MIDlet-Jar-Size	JAR ファイルのサイズ （数値）
条件必須属性 (Jad or Manifest)	**説明**
MIDlet-<n>	名前、アイコン、クラスをカンマで区切ったもの （文字列　MAX:1024Bytes）
Option 属性（標準）	**説明**
MIDlet-Description	S!アプリの詳細説明文 （文字列　MAX:128Bytes）
MIDlet-Icon	端末にアイコンとして表示する画像ファイルへのパス （文字列　MAX:64Bytes）
MIDlet-Info-URL	アプリケーションのプロパティから CP 様が用意した HTML ドキュメントへ遷移する。遷移先の URL を記載する。 （URL　MAX:1024Bytes）
MIDlet-Data-Size	RecordStore(永続記憶域)のサイズ （数値　MAX:3MBytes）
MIDlet-Install-Notify	ダウンロード結果通知先 URL （URL　MAX:256 文字）
MIDlet-Delete-Notify	削除通知先 URL （URL　MAX:256 文字）
MIDlet-Delete-Confirm	S!アプリ削除確認時、表示するテキストメッセージ （文字列　MAX:64 文字）

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

MIDlet-Push-<n>	S!アプリ起動のトリガーとなるメッセージ送信元のURL、クラス名、IP アドレス （MAX:2048Bytes）
MIDlet-Permissions	セキュリティダイアログ表示対象 API の Permission name を列記 （文字列　MAX:2048Bytes） ※　セキュリティダイアログ表示対象 API を利用した場合には必須
MIDlet-Permissions-Opt	セキュリティダイアログ表示対象 API の Permission name のうち、許可/非許可で S!アプリの動作を変えるものを列記 （文字列　MAX:2048Bytes）
MIDlet-Certificate-<n>-<m>	Jar ファイルの証明書
MIDlet-Jar-RSA-SHA1	Jar ファイルの署名
Option 属性（拡張）	**説明**
MIDxlet-API	MEXA/JSCL ライブラリを指定 （ JOCL-1.0.0 or JSCL-1.0.0 or JSCL-1.0.1 or JSCL-1.1.0 or JSCL-1.1.1 or JSCL-1.2.0 or JSCL-1.2.1 or JSCL-1.2.2 or JSCL-1.3.2 or JSCL-1.4.2 or MEXA）
MIDxlet-Resident	常駐フラグを指定するか否か
MIDxlet-ScreenSize	S!アプリにて利用する画面サイズを指定 （MAX:32Bytes） ※　本属性の記載がない場合、端末画面サイズの違いにより意図しない表示となる場合があるため、注意すること。
MIDxlet-Aux	S!アプリ上のディスプレイ及び音声を外部出力するか否か
MIDxlet-MSensor	モーションコントロールを利用するか否か
MIDxlet-Karaoke	S!カラオケとして起動するか否か また、S!アプリ制御型/端末制御型として起動するか否か
MIDxlet-Sound-Priority	MP4(なし/AAC)再生を利用するか否か
MIDxlet-WideScreen	S!アプリの描画方向を指定
MIDxlet-Java-Execution	他 S!アプリからの起動を許可するか否か
MIDxlet-Execution-Heap	S!アプリにて利用するヒープサイズを記載 ※　S!アプリからの S!アプリ起動利用時のみ
MIDxlet-Bluetooth	S!アプリにて Bluetooth®を利用するか否か
MIDxlet-Slave-Application	当該 S!アプリが他の S!アプリに従属しているか否か

表 2.2.8-2. Manifest ファイル

必須属性	説明
MIDlet-Name	S!アプリ名 （文字列　MAX:64Bytes）

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

MIDlet-Version	S!アプリのバージョン番号 （数値）
MIDlet-Vendor	S!アプリのベンダー名 （文字列　MAX:64Bytes）
MicroEdition-Profile	J2ME™のプロファイル （MIDP-1.0 or MIDP-2.0）
MicroEdition-Configuration	J2ME™のコンフィグレーション （CLDC-1.0 or CLDC-1.1）
条件必須属性 （Jad or Manifest）	**説明**
MIDlet-<n>	名前、アイコン、クラスをカンマで区切ったもの （文字列　MAX:1024Bytes）
Option 属性（標準）	**説明**
MIDlet-Permissions	セキュリティダイアログ表示対象 API の Permission name を列記 （文字列　MAX:2048Bytes） ※　セキュリティダイアログ表示対象 API を利用した場合には必須
MIDlet-Permissions-Opt	セキュリティダイアログ表示対象 API の Permission name のうち、許可/非許可で S!アプリの動作を変えるものを列記する。 （文字列　MAX:2048Bytes）

※　Jad 及び Manifest には上記で記載した MIDlet 属性を記載する。CP 様にて任意に文字列を追加することは可能だが、その際には「MIDlet-XXX」や「MIDxlet-YYY」といった文字列を追加記載してはならない。

※　MIDlet 属性名には「.」を利用することはできない。利用した場合の動作については保証できない。

※　MIDlet 属性内に URL を記載する場合、弊社固有のクエリを利用してはならない。

※　下記 MIDlet 属性を利用する場合には必ず Jad ファイルに記載する必要がある。

- MIDlet-Jar-URL
- MIDlet-Info-URL
- MIDlet-Install-Notify
- MIDlet-Delete-Notify
- MIDxlet-Slave-Application

※　上記に記載のない MIDlet 属性については MIDP 2.0 対応端末では無視をする。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

※ 下記 MIDlet 属性の指定値最大サイズについては、S!アプリ開発ガイド「端末情報 MIDP 2.0 対応端末編(各種)」を参照のこと。

- MIDlet-Name
- MIDlet-Vender
- MIDlet-Description
- MIDlet-Data-Size

以下に MIDP 1.0 から追加及び変更した MIDlet 属性について詳細を説明する。

2.2.5.1. MIDlet-Install-Notify

追加

① 意味：S!アプリをダウンロードした際、ダウンロード結果通知を端末より送出する。ダウンロード結果通知先 URL を記載する。
② 書式：URL
③ 制限：最大 256 文字
④ その他：MIDlet-Install-Notify に URL を記載すると S!アプリをダウンロードした際、記載した URL へダウンロード結果通知を送出する。

2.2.5.2. MIDlet-Delete-Notify

追加

① 意味：S!アプリを削除した際、削除通知を端末より送出する。削除通知先 URL を記載する。
② 書式：URL
③ 制限：最大 256 文字
④ その他：MIDlet-Delete-Notify に URL を記載すると S!アプリを削除した後の最初の S!アプリダウンロードの際、記載した URL へ削除通知を送出する。

2.2.5.3. MIDlet-Delete-Confirm

追加

① 意味：S!アプリの削除時確認画面に表示するテキストメッセージを記載する。
② 書式：符号化文字集合 JISX0201 および JISX0208 で規定される文字を用いた文字列で記載すること。
③ 制限：最大 64 文字

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5.4. MIDlet-Push-<n>

追加

① 意味：S!アプリ起動のトリガーとなるメッセージ送信元の URL、クラス名、IP アドレスを記載する。

② 制限：最大 2048Bytes

2.2.5.5. MIDlet-Permissions

追加

① 意味：S!アプリが使用するセキュリティダイアログ表示対象 API の Permission name を列記する。ただし、MIDlet-Permission-opt に該当するものは含まない。セキュリティダイアログ表示対象 API を利用している場合には必ず記載すること。

② 制限：最大 2048Bytes

③ その他：カンマ区切りで該当するクラスの Permission name を列記する。

※ MIDP-1.0、CLDC-1.0 のみを利用した S!アプリを MIDP 2.0 対応端末向けに提供する場合、セキュリティダイアログ表示対象 API を利用していない S!アプリにおいても MIDlet-Permissions を MIDlet 属性に追加する必要がある。尚、この場合には Permission name に「javax.microedition.io.Connector.http」を記述すること。

2.2.5.6. MIDlet-Permissions-Opt

追加

① 意味：S!アプリが使用するセキュリティダイアログ表示対象 API の Permission name のうち、端末の対応/非対応によって S!アプリが振る舞いを変えるものを列記する。本属性に記載した場合、端末が該当機能に非対応であることを考慮した S!アプリを作成する必要がある。本属性に記載した Permissions name は MIDlet-Permissions には記載を行なわないこととする。

② 制限：最大 2048Bytes

③ その他：カンマ区切りで該当するクラスの Permission name を列記する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5.7. MIDlet-Certificate-<n>-<m>

追加

① 意味：Jar ファイルの証明書について記載する。

2.2.5.8. MIDlet-Jar-RSA-SHA1

追加

① 意味：Jar ファイルの署名について記載する。

2.2.5.9. MIDxlet-API

変更（旧 MIDlet-OCL）

① 意味：S!アプリ固有の拡張ライブラリ名を記載する。

② 書式：S!アプリにて利用する最新のライブラリ名を記載する。ライブラリ名は「＜ライブラリ名＞‘-’＜バージョン番号＞」で表記し、ライブラリ名の間はカンマを区切り文字として記載する。（カンマの前後には空白文字を置かないこと）尚、MEXA 以降に規定されたライブラリ（Bluetooth®など）については、「JSCL-1.4.2」或いは「MEXA」と記載すること。

（例１）

MEXA Core：クラス SpriteCanvas <JSCL-1.1.X 以降>
クラス FixedPoint <JSCL-1.0.X 以降>
を利用した S!アプリの場合

◆ MIDxlet-API: JSCL-1.1.0

（例２）

MEXA Core：クラス SpriteCanvas <JSCL-1.1.X 以降>
MEXA MotionControl：クラス MostionDetectiveSensor <JSCL-1.3.2 以降>
を利用した S!アプリの場合

◆ MIDxlet-API: JSCL-1.3.2

（例３）

MEXA Core：クラス SpriteCanvas <JSCL-1.1.X 以降>
MEXA Bluetooth：クラス Device <MEXA 以降>
を利用した S!アプリの場合

◆ MIDxlet-API: JSCL-1.4.2 or MEXA

③ その他：「JSCL」、「MEXA」の記載がある場合、ダウンロードした S!アプリを MEXA/JSCL 使用と判断する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

＜xx2/xx3、904T＞

・ 記載あり（MEXA/JSCL 利用 S!アプリ）
使用可能プロファイル：CLDC 1.1、MIDP 2.0、３D Sound API、MEXA/JSCL
MEXA/JSCL を使用している S!アプリと判断し WMA 1.1、MMAPI 1.1、M3G、VSCL の機能を使用した場合にはエラーとする。

・ 記載なし（JSR 利用 S!アプリ）
使用可能プロファイル：CLDC 1.1、MIDP 2.0、WMA 1.1、MMAPI 1.1、M3G、VSCL 、３D Sound API
MEXA/JSCL を使用していない S!アプリと判断し、MEXA/JSCL の機能を使用した場合にはエラーとする。

＜xx4 以降＞

同機能（Graphics3D（MEXA/JSCL)と M3G など）を利用する場合、本属性の記載あり/記載なしによって下記動作となる。

表 2. 2. 8. 9-1. MIDxlet-API の記載による動作差異

機能名	記載あり MEXA/JSCL	記載なし その他(JSR or VSCL)
DeviceControl	DeviceControl <MEXA Core(JSCL)>	Additional APIs <VSCL Core>
3D Graphics	Graphics3D <MEXA Core(JSCL)>	M3G <JSR 184>
待受アプリ制御	Resident <MEXA Unit2(JSCL)>	Additional APIs <VSCL Core>
ImageEncorder	ImageEncoder <MEXA Unit2(JSCL)>	Additional APIs <VSCL Core>
赤外線リモコン	RemoteControl <MEXA/JSCL>	Remote Control API <VSCL Core>
バーコード認識	BarCode <MEXA/JSCL>	Bar Code Recognition API <VSCL>

記載ありの場合・・・MEXA/JSCL のみ利用可能
記載なしの場合・・・JSR、VSCL のみ利用可能

※排他対象 Profile 使用時の動作

MEXA/JSCL と JSR で同機能の profile がある場合には、MIDxlet-API で Profile の排他を行う。起動時に排他対象 profile 利用クラスをロードする際にエラーとして起動を中止するか、排他対象の Profile の API を使用した時点でエラーとなる。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5.10. MIDxlet-Resident

変更（旧　MIDlet-Resident）

① 意味：MIDlet-Resident と同様である。

② その他：記載方法については S!アプリ開発ガイド「MIDP 1.0 対応端末編」を参照のこと。

2.2.5.11. MIDxlet-ScreenSize

変更（旧　MIDlet-Application-Range）

① 意味：S!アプリにて利用する画面サイズを指定する。

② 書式：MIDxlet-ScreenSize: 120,130 or 240,260 or 240,320 or 480,520 or 480,640

※　802SE に対応する場合には以下を記載すること。

MIDxlet-ScreenSize: 176,182

③ 制限：120,130(240,260/240,320/480,520/480,640)間はカンマ（半角）を使用する。

数値は全て半角を使用する。属性値に空白を入れない。ただし、”：”と属性値に空白は入れてもよい。

尚、オフィシャルコンテンツにおいて、本属性の記載なし或いは値を 0,0 と記載した S!アプリを S!アプリサーバへアップロードした際には、値を 240,260 として扱うため注意すること。

④ その他：S!アプリで利用する画面サイズの指定方法は以下のとおりである。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

表 2. 2. 8. 11-1. MIDxlet-ScreenSize の記載による表示方法

MIDxlet-ScreenSize の記載 端末画面サイズ	120,130	176,182	240,260	240,320	480,520	480,640
120 x 130 (QQVGA)	・画面サイズ 120 x 130 ・表示エリア 120 x 130 ＜等倍表示＞	DL 不可	DL 不可	DL 不可	DL 不可	DL 不可
176 x 182 (802SE)	・画面サイズ 120 x 130 ・表示エリア 120 x 130 ＜等倍表示＞	・画面サイズ 176 x 182 ・表示エリア 176 x 182 ＜等倍表示＞	DL 不可	DL 不可	DL 不可	DL 不可
240 x 260 (QVGA)	・画面サイズ 120 x 130 ・表示エリア 240 x 260 ＜2 倍表示＞	・画面サイズ 240 x 260 ・表示エリア 240 x 260 ＜等倍表示＞	・画面サイズ 240 x 260 ・表示エリア 240 x 260 ＜等倍表示＞	DL 不可	DL 不可	DL 不可
240 x 320 (WQVGA)	・画面サイズ 120 x 130 ・表示エリア 240 x 260 ＜2 倍表示＞	・画面サイズ 240 x 260 ・表示エリア 240 x 260 ＜等倍表示＞	・画面サイズ 240 x 260 ・表示エリア 240 x 260 ＜等倍表示＞	・画面サイズ 240 x 320 ・表示エリア 240 x 320 ＜等倍表示＞	DL 不可	DL 不可
480 x 520 (VGA)	・画面サイズ 120 x 130 ・表示エリア 480 x 520 ＜4 倍表示＞	・画面サイズ 240 x 260 ・表示エリア 480 x 520 ＜2 倍表示＞	・画面サイズ 240 x 260 ・表示エリア 480 x 520 ＜2 倍表示＞	・画面サイズ 480 x 520 ・表示エリア 480 x 520 ＜等倍表示＞	・画面サイズ 480 x 520 ・表示エリア 480 x 520 ＜等倍表示＞	DL 不可
480 x 640 (WVGA)	・画面サイズ 120 x 130 ・表示エリア 480 x 520 ＜4 倍表示＞	・画面サイズ 240 x 260 ・表示エリア 480 x 520 ＜2 倍表示＞	・画面サイズ 240 x 260 ・表示エリア 480 x 520 ＜2 倍表示＞	・画面サイズ 480 x 640 ・表示エリア 480 x 640 ＜2 倍表示＞	・画面サイズ 480 x 520 ・表示エリア 480 x 520 ＜等倍表示＞	・画面サイズ 480 x 640 ・表示エリア 480 x 640 ＜等倍表示＞

2.2.5.12. MIDxlet-Aux

変更（旧　MIDlet-Aux）

① 意味：MIDlet-Aux と同様である。

② その他：記載方法については S!アプリ開発ガイド「MIDP 1.0 対応端末編」を参照のこと。本機能に対応していない端末では本属性を無視する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5.13. MIDxlet-MSensor

① 意味：モーションコントロールを利用するか否か。
② 書式：ISO8850-1 の‘Y’もしくは‘N’の 1 文字で指定する。（大文字である点に注意）
③ 制限：モーションコントロールを利用する場合には‘Y’を指定する。利用しない場合は‘N’を指定する。
④ デフォルト値：N
⑤ その他：モーションコントロール非対応端末では、本属性を無視する。
モーションコントロールを利用した S!アプリは、継続起動待受アプリとして起動することは出来ない。つまり、MIDxlet-MSensor: Y の場合、MIDxlet-Resident: S を無視する。（MIDxlet-Resident: N として動作する）

2.2.5.14. MIDxlet-Karaoke

① 意味：S!カラオケとして起動するか否か。又、その時、S!アプリ制御型/端末制御型として起動するか否か。
② 書式：ISO8850-1 の‘Y’、‘N’、‘X’の 1 文字で指定する。（大文字である点に注意）
③ 制限：端末制御型 S!カラオケとして起動する場合は‘Y’を指定する。通常の S!アプリとして起動する場合は‘N’を指定する。S!アプリ制御型 S!カラオケとして起動する場合は‘X’を指定する。
④ デフォルト値：N
⑤ その他：カラオケ機能非対応端末では、本属性を無視する。
S!カラオケは、待受アプリ/継続起動待受アプリとして起動することは出来ない。
◆ MIDxlet-Karaoke が‘Y’または‘X’である場合、以下の MIDlet 属性をデフォルト値として S!アプリを起動する。
MIDxlet-Resident: N
MIDxlet-Aux: Y
MIDlet-Karaoke が‘Y’の場合、３Ｄグラフィックス機能を利用することはできない。利用した場合には例外(NoClassDefFoundError)が発生する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5.15. MIDxlet-Sound-Priority

追加

① 意味：MP4（なし／AAC）再生を利用するか否か。

② 書式：ISO8850-1 の‘Y’もしくは‘N’の 1 文字で指定する。（大文字である点に注意）

③ 制限：S!アプリにて MP4（なし／AAC）再生を行う場合には‘Y’を指定する。再生を行わない場合には‘N’を指定する。

④ デフォルト値：N

⑤ その他：MP4（なし／AAC）再生非対応端末では、本属性を無視する。

＜804SH＞

MIDxlet-Sound-Priority が‘Y’の場合、以下点に注意すること

・３Ｄグラフィックス機能を利用することはできない。利用した場合には例外(NoClassDefFoundError)が発生する。

・外部出力機能の利用ができない。（MIDxlet-Aux を無視する）

・描画速度が遅くなる場合がある。

2.2.5.16. MIDxlet-WideScreen

追加

① 意味：S!アプリの描画方向を指定。

② 書式：ISO8850-1 の‘Y’もしくは‘N’の 1 文字で指定する。（大文字である点に注意）

③ 制限：横画面モードにて描画を行う場合には‘Y’を指定する。
縦画面モードにて描画を行う場合には‘N’を指定する。

④ デフォルト値：N

⑤ その他：以下に横画面モード/縦画面モードのイメージを記載する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

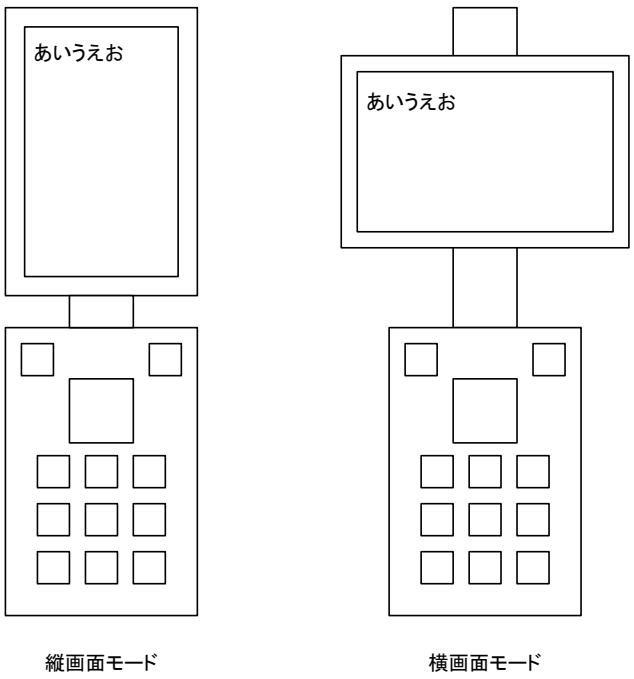


図 2.2.8.16-1. 縦横画面例

※ MIDxlet-WideScreen: Y の場合、座標（x,y）が変更となるため、以下の値には注意すること。
・S!アプリにて取得する液晶サイズ
・MIDxlet-ScreenSize にて設定する値

2.2.5.17. MIDxlet-Java-Execution

追加

① 意味：他 S!アプリからの起動許可/非許可を指定。
② 書式：ISO8850-1 の‘Y’もしくは‘N’の 1 文字で指定する。（大文字である点に注意）
③ 制限：他 S!アプリからの起動を許可とする場合には‘Y’を指定する。
他 S!アプリからの起動を非許可とする場合には‘N’を指定する。
④ デフォルト値：Y
⑤ その他：MIDxlet-Java-Execution: N と記載した S!アプリに対して、他 S! アプリより起動要求を行なうことはできない。起動要求を行なう際、起動元 S!アプリにエラーを返す。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5.18. MIDxlet-Execution-Heap

追加

① 意味：S!アプリにて利用するヒープサイズをバイト数で記載する。記載がない場合、S!アプリは端末のヒープサイズを最大限使用することとみなす。

② 書式：自然数で表記する。

③ 制限：各端末のヒープサイズを最大値とする。

④ デフォルト値：各端末ヒープサイズの最大値

⑤ その他：本属性はS!アプリからのS!アプリ起動において、主アプリを一時停止する端末にのみ利用可能とし、主アプリを終了する端末では本属性を無視する。S!アプリからのS!アプリ起動において、主アプリを一時停止する場合、端末のヒープサイズから主アプリのヒープサイズ（MIDxlet-Execution-Heapの値）を引いたヒープ領域を従アプリで利用することとなる。

（例）端末のヒープサイズが4Mかつ主アプリを一時停止可能な端末の場合

表 2.2.8.18-1. MIDxlet-Execution-Heapの記載および動作

1つ目のS!アプリ（主アプリ）	2つ目のS!アプリ（従アプリ）	端末動作
MIDxlet-Execution-Heap 記載なし	－	主アプリを起動時、4Mのヒープを割り当てる。 従アプリを起動時、エラーとなる。
MIDxlet-Execution-Heap: 2097152（2M）	MIDxlet-Execution-Heap 記載なし	主アプリを起動時、2Mのヒープを割り当てる。 従アプリを起動時、残りの2Mを割り当てる。
MIDxlet-Execution-Heap: 2097152（2M）	MIDxlet-Execution-Heap: 1048576（1M）	主アプリを起動時、2Mのヒープを割り当てる。 従アプリを起動時、1Mのヒープを割り当てる。
MIDxlet-Execution-Heap: 2097152（2M）	MIDxlet-Execution-Heap: 3145728（3M）	主アプリを起動時、2Mのヒープを割り当てる。 従アプリを起動時、エラーとなる。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.5.19. MIDxlet-Bluetooth

追加

① 意味：S!アプリにて Bluetooth®を利用するか否か。

② 書式：ISO8850-1 の‘Y’もしくは‘N’の 1 文字で指定する。（大文字である点に注意）

③ 制限：S!アプリにて Bluetooth®を利用する場合には‘Y’を指定する。
S!アプリにて Bluetooth®を利用しない場合には‘N’を指定する。

④ ④ デフォルト値：N

⑤ その他：Bluetooth®非対応端末では、本属性を無視する。
Bluetooth®を利用した S!アプリは、待受アプリ/継続起動待受アプリとして起動することは出来ない。

- MIDxlet-Bluetooth が‘Y’である場合、以下の MIDlet 属性をデフォルト値として S!アプリを起動する。
  MIDlet-Resident: N

また、Bluetooth を利用した S!アプリは、常に着信優先動作として動作する。

尚、以下の場合には API コール時エラーをかえす。

１．本属性を‘Y’に指定し端末の Bluetooth®設定が OFF の場合

２．本属性がない場合

３．本属性が‘N’の場合

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.2.8.20. MIDxlet-Slave-Application

追加

① 意味：S!アプリが他の S!アプリに従属していることを宣言する。
本属性を記載した S!アプリは必ず別 S!アプリからの PlatformRequest()によってのみ起動可能となる。

② 書式：従属元の MIDlet-Name, 従属元の MIDlet-Vendor, コンテンツ URL
（当該 S!アプリのダウンロードページなど）

③ 制限：最大 1024byte
コンテンツ URL に弊社固有クエリを利用してはならない。

④ その他：本属性を記載した S!アプリを PlatformRequest()以外の契機で起動した場合（S!アプリライブラリ、ブラウザ上等)、当該 S!アプリの起動は行わず本属性に記載した S!アプリを起動する。但し、待受け設定された S!アプリが起動する際には本属性を無視して S!アプリを起動する。
MIDlet-Name, MIDlet-Vendor が一致する S!アプリが端末内に存在する場合には、一致する S!アプリを起動する。また、存在しない場合は、起動できない旨を通知した上でブラウザを起動してコンテンツ URL へアクセスするかどうかの確認画面を表示する。
また、本属性は Untrusted MIDlet では利用できない。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.3. 基本動作

MIDP 2.0 対応端末向け S!アプリの動作は、MIDP 1.0 対応端末向け S!アプリとほぼ同様である。本章では、MIDP 2.0 対応端末向け S!アプリ固有の動作について説明する。

2.3.1. ダウンロード

MIDP 2.0 対応端末向けに S!アプリを提供する場合、S!アプリに「Trusted Domain（Third Party Domain)」の署名/証明書を付与する必要がある。尚、S!アプリに署名/証明書を付与しない場合には Untrusted MIDlet となり、一部機能を S!アプリにて利用することができない。

S!アプリをダウンロードする際、端末にて Jad/Jar の解析を行う。解析の結果、取得した Jad/Jar が不正であると判断した場合には、不正なファイルとしてダウンロードを行わない。MIDP 2.0 対応端末では MIDP 1.0 対応端末での Jad/Jar 解析に加え、①署名検証、②証明書検証についても解析を行う。解析した結果によりダウンロードした S!アプリを「Trusted MIDlet（Third Party Domain)」あるいは「Untrusted MIDlet」とする。

① 署名検証 ・・・署名不一致の場合にはダウンロード不可となり、署名がない場合には Untrusted MIDlet とする。

② 証明書検証・・・証明書不一致の場合にはダウンロード不可となり、証明書（Third Party Domain）が一致する場合には Trusted MIDlet（Third Party Domain）とする。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.3.2. セキュリティ

MIDP 2.0 対応端末向け S!アプリにおけるセキュリティは MIDP 2.0 で規定している「Trusted MIDlet」を採用する。
Trusted MIDlet とは、MIDP 2.0 にて規定している PKI による認証メカニズムで信頼した S!アプリである。PKI による認証メカニズムによって信頼されない S!アプリは Untrusted MIDlet とする。

MIDP 2.0 対応端末向け S!アプリは、それぞれの Domain に規定されたセキュリティポリシーに基づき、起動/実行する。つまり、S!アプリは Trusted Domain、Untrusted Domain に基づき、起動/実行中、ユーザへセキュリティダイアログを表示する。セキュリティダイアログの表示は機能ごとに規定している。以下に機能別のセキュリティダイアログ表示方法について示す。

表 2.3.2-1. 機能別表示方法

Fuction Group	機能名称	Trusted Domain (Third Party Domain)		Untrusted Domain	
		デフォルト	その他	デフォルト	その他
Phone Call	通話発信	④	①	④	①
Net Access	ﾈｯﾄﾜｰｸ接続	③	①、②、④	×	
Messaging	ﾒｰﾙ送受信	④	①	④	①
Application Auto Invocation	ｱﾌﾟﾘ自動起動	③	①、②、④	③	①、④
Local Connectivity	外部接続	③	①、②、④	③	①、②、④
Multimedia Recording	録画/録音	③	①、②	④	①、③
Read User Data Access	個人情報取得	④	①、②、③	×	
Write User Data Access	個人情報書込	④	①、②、③	×	
Location Access	位置情報取得	④	①、③	×	
MEXA Bluetooth	Bluetooth®	③	①、②	×	

［凡例］

① No：

◆ セキュリティダイアログは表示しない。

API コール時に SecurityException を返す。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

② Blanket：

◆ セキュリティダイアログは機能（Function Group）ごとに表示する。

尚、過去にユーザがセキュリティダイアログによる確認を行った場合には、以下の場合を除きセキュリティダイアログを再表示しない。

※ ユーザが S!アプリを削除した場合

※ ユーザが当該 S!アプリの Function Group に対するセキュリティ設定を変更した場合

③ Session：

◆ セキュリティダイアログは機能（Function Group）ごとに表示する。

◆ S!アプリを一度終了した上で、再度起動した際には非許可と判断せずにセキュリティダイアログを表示する。

④ Oneshot：

◆ セキュリティダイアログ表示対象となる API のコールごとにセキュリティダイアログを表示する。

× 利用不可：

◆ 機能の利用ができない。

◆ セキュリティダイアログは表示せず API コール時に SecurityException を返す。

［補足］

※ セキュリティダイアログに対して、ユーザが実行拒否を選択した場合には対象となる API は実行せず、Exception を返す。よって、セキュリティダイアログ表示対象 API を利用する場合には Exception により S!アプリがエラー終了しないよう配慮する必要がある。

※ ②Blanket、③Session では S!アプリの起動後、セキュリティダイアログを表示する。

または

セキュリティダイアログ表示対象となる API の初回コール時にセキュリキュリティダイアログを表示する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

表 2.3.2-2. セキュリティダイアログ表示対象 API 一覧

Function Groups	Permission name	Package/Class/Method
Phone Call	com.j_phone.io.Connector.PhoneConnection	com.j_phone.io.PhoneConnection#connect()
Net Access	javax.microedition.io.Connector.http javax.microedition.io.Connector.https	javax.microedition.io.Connector#open(name)
		javax.microedition.io.Connector#open(name , mode)
		javax.microedition.io.Connector#open(name , mode , timeouts)
		com.j_phone.amuse.Phrase(url)
		com.j_phone.media.MediaPlayer(url)
		com.j_phone.media.MediaPlayer#setMediaData(url)
		com.jblend.media.jpeg.JpegData(url)
		com.jblend.media.png.PngData(url)
		com.jblend.media.smaf.SmafData(url)
		com.jblend.media.smaf.phrase.AudioPhrase(url)
		com.jblend.media.smaf.phrase.Phrase(url)
		com.j_phone.io.(BrowserConnection)Connector.open() スキームが”url://”、”urls://”の場合
Messaging	javax.microedition.io.Connector.sms	javax.microedition.io.Connector#open(name)
		javax.microedition.io.Connector#open(name , mode)
		javax.microedition.io.Connector#open(name , mode , timeouts)
	javax.wireless.messaging.sms.send	javax.wireless.messaging.MessageConnection#send()
		javax.microedition.io.Connector#open(name , mode)
		javax.microedition.io.Connector#open(name , mode , timeouts)
	javax.wireless.messaging.sms.receive	javax.wireless.messaging.MessageConnection#receive()
		javax.microedition.io.Connector#open(name , mode)
		javax.microedition.io.Connector#open(name , mode , timeouts)
	com.j_phone.system.MailAgent.send	com.j_phone.system.MailAgent#send(data)
	com.j_phone.system.MailAgent.receive	com.j_phone.system.MailAgent#receiveRemainder(data)
Application Auto Invocation	javax.microedition.io.PushRegistry	javax.microedition.io.PushRegistry#registerConnection()

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

		javax.microedition.io.PushRegistry#registerAlarm()
Local Connectivity	javax.microedition.io.Connector.comm	javax.microedition.io.Connector#open(name)
		javax.microedition.io.Connector#open(name , mode)
		javax.microedition.io.Connector#open(name , mode , timeouts)
	com.vodafone.io.RemoteControl	com.vodafone.io.RemoteControl#send()
	com.j_phone.io.RemoteControl	com.j_phone.io.RemoteControl#send(numOfData , data)
	com.j_phone.io.Connector.ServerObexConnection	com.j_phone.io.ServerObexConnection#accept()
	com.j_phone.io.Connector.ClientObexConnection	com.j_phone.io.ClientObexConnection#connect()
Multimedia Recording	javax.microedition.media.control.RecordControl	javax.microedition.media.control.RecordControl#setRecordLocation(locator)
	javax.microedition.media.control.VideoControl.getSnapshot	javax.microedition.media.control.VideoControl#getSnapshot()
	com.j_phone.io.Connector.BarcodeConnection.capture	com.j_phone.io.BarCodeConnection#capture()
	com.j_phone.io.Connector.CameraConnection.capture	com.j_phone.io.CameraConnection#capture()
	com.j_phone.io.Connector.VideoConnection.capture	com.j_phone.io.VideoConnection#capture()
Read User Data Access	com.vodafone.midlet.ResidentMIDlet	com.vodafone.midlet.ResidentMIDlet()
	com.j_phone.io.FileUtility.read	com.j_phone.io.FileUtility#getMediaPlayer(path)
		com.j_phone.io.FileUtility#play(path)
	com.j_phone.io.Connector.StorageConnection.read	com.j_phone.io.StorageConnection#exists()
		com.j_phone.io.StorageConnection#getLength ()
		com.j_phone.io.StorageConnection#getType ()
		com.j_phone.io.StorageConnection#getTypeString ()
		com.j_phone.io.StorageConnection#isCopyrighted ()
		com.j_phone.io.StorageConnection#isFile ()
		com.j_phone.io.StorageConnection#isFolder ()
		com.j_phone.io.StorageConnection#list ()
		com.j_phone.io.StorageConnection#openInputStream()
	com.j_phone.media.ResourceOperatorManager.read	com.j_phone.media.ResourceOperatorManager#getResourceOperator(type)

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

	com.j_phone.midlet.ResidentMIDlet	com.j_phone.midlet.ResidentMIDlet()
	com.j_phone.system.DeviceControl.read	com.j_phone.system.DeviceControl#setMailListener(listener)
		com.j_phone.system.DeviceControl#setScheduledAlarmListener(listener)
		com.j_phone.system.DeviceControl#setTelephonyListener(listener)
	com.j_phone.phonedata.CallInfo.read	com.j_phone.phonedata.CallInfo#getInstance()
	com.j_phone.phonedata.PhoneDataConnector.read	com.j_phone.phonedata.PhoneDataConnector#getElementCount(name , index)
		com.j_phone.phonedata.PhoneDataConnector#getRestCount(name , index)
	com.j_phone.phonedata.AddressBook.read	com.j_phone.phonedata.AddressBook#elements(type,searchString,form,max)
		com.j_phone.phonedata.(AddressBook)PhoneData#elements(position , max , sortType)
	com.j_phone.phonedata.ReceivedMailBox.read	com.j_phone.phonedata.(ReceivedMailBox)PhoneData#elements(position , max , sortType)
		com.j_phone.phonedata.ReceivedMailBox#getUnReadMailCount()
	com.j_phone.phonedata.SentMailBox.read	com.j_phone.phonedata.(SentMailBox)PhoneData#elements(position , max , sortType)
	com.j_phone.system.DeviceControl.Location	com.j_phone.system.DeviceControl#getLatitude()
		com.j_phone.system.DeviceControl#getLongitude()
		com.j_phone.system.DeviceControl#getPlaceName()
	com.j_phone.system.DeviceControl.owner	com.j_phone.system.DeviceControl#getmyTelNumber()
		com.j_phone.system.DeviceControl#getImei()
Write User Data Access	com.j_phone.io.Connector.StorageConnection.write	com.j_phone.io.StorageConnection#createFolder()
		com.j_phone.io.StorageConnection#delete()
		com.j_phone.io.StorageConnection#openOutputStream()
		com.j_phone.io.StorageConnection#renameTo(newName)
	com.j_phone.phonedata.AddressBook.write	com.j_phone.phonedata.(AddressBook)PhoneData#createElement(element)
		com.j_phone.phonedata.(AddressBook)PhoneData#delete(element)

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

		com.j_phone.phonedata.(AddressBook)PhoneData.importElementRawData(data)
	com.j_phone.phonedata.ReceivedMailBox.write	com.j_phone.phonedata.(ReceivedMailBox)PhoneData#createElement(element)
		com.j_phone.phonedata.(ReceivedMailBox)PhoneData#delete(element)
		com.j_phone.phonedata.(ReceivedMailBox)PhoneData#importElementRawData(data)
		com.j_phone.phonedata.MailData#setState(state)
	com.j_phone.phonedata.SentMailBox.write	com.j_phone.phonedata.(SentMailBox)PhoneData#createElement(element)
		com.j_phone.phonedata.(SentMailBox)PhoneData#delete(element)
		com.j_phone.phonedata.(SentMailBox)PhoneData#importElementRawData(data)
	com.j_phone.io.TelevisionConnection.write	com.j_phone.io.TelevisionConnection#removeReserveData(TelevisionReserveData)
Location Access	com.j_phone.system.DeviceControl.updateLocation	com.j_phone.system.DeviceControl#updateLocationInfo()
MEXA Bluetooth	com.vodafone.bluetooth.open	com.vodafone.bluetooth.SessionManager#open(service)
		com.vodafone.bluetooth.SessionMember#open(service)
	com.vodafone.bluetooth.openSecured	com.vodafone.bluetooth.SessionManager#openSecured(service, encrypt)
		com.vodafone.bluetooth.SessionMember#openSecured(service, encrypt, authorized)
	com.mexa.bluetooth.open	com.mexa.bluetooth.SessionManager#open(service)
		com.mexa.bluetooth.SessionMember#open(service)
	com.mexa.bluetooth.openSecured	com.mexa.bluetooth.SessionManager#openSecured(service, encrypt)
		com.mexa.bluetooth.SessionMember#openSecured(service, encrypt, authorized)

※ 上記記載の他、SecurityException を返す API がセキュリティダイアログ表示対象 API となる。SecurityException を返す API については各 JSR に関するドキュメントを参照のこと。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.3.3. 通知

MIDP 2.0 対応端末では、2.2.8. MIDlet 属性で説明した MIDlet-Install-Notify 属性、MIDlet-Delete-Notify 属性を Jad ファイルに記載することができる。本属性を記載することにより、S!アプリのダウンロード・削除の結果について、CP 様の任意の URL へ通知を送出することができる。ただし、端末が圏外などの理由で正常に通信が行えない場合には通知の再送はしない。

Jad ファイル内に MIDlet-Install-Notify 属性値を記述している場合、MIDP 2.0 対応端末は以下のようにダウンロード結果通知を送出する。

- ダウンロードが終了した時点で通知を送出する。
- ダウンロード以外（外部メモリからの移動、外部メモリへの移動）のインストール時には通知しない。
- 属性値に記載した URL へ通知を送出する。

Jad ファイル内に MIDlet-Delete-Notify 属性値を記述している場合、MIDP 2.0 対応端末は以下のように削除通知を送出する。

- S!アプリの削除を行った後の最初のダウンロード処理の最後の時点で通知を送出する。
- 複数削除を行った場合には最後に削除した S!アプリのみ通知を行う。
- 端末内の S!アプリを削除した場合にのみ通知を送出する。
- 属性値に記載した URL へ通知を送出する。

各通知には POST リクエストの Body の最初の行に Status Code と Status Message を記載している。
Status Code 及び Status Message の一覧を以下に示す。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

表 2.3.3-1. Status Code/Status Message 一覧

Status Code	Status Message	内容
900	Successe	インストール成功時に送出
901	Insufficient Memory	S!アプリ保存領域不足時に送出
902	User Cancelled	ユーザキャンセル時に送出
904	JAR size mismatch	JAD,JAR のサイズ整合が取れない場合に送出
905	Attribute Mismatch	必須属性不足、JAD/JAR の整合が取れない場合に送出
906	Invalid Descriptor	JAD の文字コード不正時に送出
907	Invalid JAR	JAR が不正な場合に送出
908	Incompatible Configuration or Profile	Configuration、Profile が間違っている場合に送出
909	Application authentication failure	証明の失敗時に送出
910	Application authorization failure	同一 S!アプリが存在している場合に送出
911	Push registration failure	Push Registry が失敗した場合に送出
912	Deletion Notification	S!アプリ削除後の最初のダウンロード処理の最後に送出

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.3.4. 通信機能

MIDP 2.0 対応端末向け S!アプリにてサポートしている通信機能を以下に示す。

表 2.3.4-1. 通信機能

	Scheme	対応
HttpConnection	http://	○
HttpsConnection	https://	○
CommConnection	comm.:com# (#はポート番号)	△

［凡例］
○：全ての端末にて対応
△：一部端末にて対応

2.3.4.1. 送信

＜利用可能なメソッドについて＞
MIDP 2.0 対応端末向け S!アプリの HTTP 通信で利用可能なメソッドは、GET と POST である。さらに、端末で実装をしている場合に限り HEAD が利用できる。その他メソッドについてはサポートしない。

＜URL について＞
MIDP 2.0 対応端末では S!アプリにて通信機能を利用する際、セキュリティダイアログを表示する。この際、一部の端末ではセキュリティダイアログ内に URL を表示する。よって、通信機能を利用する場合には URL に弊社固有のクエリを利用してはならない。

＜Header 情報について＞
S!アプリから HTTP もしくは HTTP s 通信を行う際に S!アプリから下記に示す Header を付与することはできない。

- User-Agent
- x-wap-profile
- x -jphone-uid

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

※　端末から標準で送出する HTTP Header を S!アプリから指定した場合の動作については端末実装依存となり、動作保証できない。

尚、S!アプリからの通信時に以下の Header 情報が自動付与される。
ただし、値については端末実装依存となる。

表 2.3.4.1-1. Header 情報一覧

Header	GET 通信時	POST 通信時	HEAD 通信時
Accept	*/*	*/*	*/*
Accept-Encoding	端末が対応している全ての Charset	端末が対応している全ての Charset	端末が対応している全ての Charset
Accept-Language	identity	identity	identity
Host	要求するリソースのホスト名およびポート番号	要求するリソースのホスト名およびポート番号	要求するリソースのホスト名およびポート番号
User-Agent	端末ごとの User-agent	端末ごとの User-agent	端末ごとの User-agent
x-wap-profile	端末ごとの x-wap-profile	端末ごとの x-wap-profile	移端末ごとの x-wap-profile
Content-Type	－	Application/octet-stream	－
Content-Length 若しくは Transfer-Encoding	－	Content-Length の場合は送信サイズ。Transfer-Encoding の場合は「chunked」	－

＜User-Agent について＞
MIDP 2.0 対応端末ではブラウザからの通信時、S!アプリからの通信時、その他アプリケーション（SVG、Flash）からの通信時において、送出する User-Agent が異なる。また、Trusted MIDlet からの通信時と Untrusted MIDlet からの通信時では、送出する User-Agent が異なる。

以下に S!アプリからの通信時に送出する User-Agent、Untrusted MIDlet からの通信時に送出する User-Agent の特徴を示す。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

S!アプリからの通信時

- UE-Application-Type は「Java」となる。
- UE-Application-Name、UE-Application-Version は端末実装依存となる。

（例）

・User-Agent: SoftBank/1.0/111SB/SB0001/SN123456789012345
Java/Java/1.0 Profile/MIDP-2.0 Configuration/CLDC-1.1

Untrusted MIDlet からの通信時

- User-Agent に”UNTRUSTED/1.0”を追加あるいは”UNTRUSTED/1.0”となる。

（例１）

・User-Agent: SoftBank/1.0/111SB/SB0001/SN123456789012345
Java/Java/1.0 Profile/MIDP-2.0 Configuration/CLDC-1.1
UNTRUSTED/1.0

（例２）

User-Agent: **UNTRUSTED**/1.0

2.3.4.2. 受信

＜利用可能データについて＞

MIDP 2.0 対応端末向け S!アプリでは HTTP 通信を利用し、2.2.6. データサイズで記載したデータを受信することができる。ただし、著作物保護設定をしたファイルを利用することはできない。利用可能なデータの詳細については 2.2.6. データサイズを参照のこと。

＜レスポンス受信時動作＞

サーバからレスポンスを受信した際の動作は以下のいずれかとなる。尚、どちらの動作を行うかについては端末実装依存となる。

- HTTP Status Code の種別に関わらず S!アプリへ返却する。
- 接続先より 30X レスポンスを受信した際は、ネイティブ層にてリダイレクトを自動で実施しリダイレクト先からのレスポンス結果を S!アプリへ返却する

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.3.4.3. HTTPs 通信

MIDP 2.0 対応端末向け S!アプリでは HTTPs 通信を利用することができる。MIDP 2.0 対応端末向け S!アプリでの HTTPs 通信には①「MIDP 2.0 対応端末 ⇔ CP 様 Web サーバ」間で SSL を利用する方法 ＜End-to-End＞と②「MIDP 2.0 対応端末 ⇔ Pull-GW」間と「Pull-GW ⇔ CP 様 Web サーバ」間で SSL を利用する方法 ＜中継＞がある。以下に各方法について説明をする。

① 「MIDP 2.0 対応端末 ⇔ CP 様 Web サーバ」間で SSL を利用する方法
＜End-to-End＞

MIDP 2.0 で規定しているインタフェース HttpsConnection を利用することにより実現する。

② 「MIDP 2.0 対応端末 ⇔ Pull-GW」間と「Pull-GW ⇔ CP 様 Web サーバ」間で SSL を利用する方法
＜中継＞

MIDP 2.0 で規定しているインタフェース HttpsConnection を利用し、特殊な URL を指定することにより実現する。以下に特殊な URL および SSL セッションの中継の一例を示す。

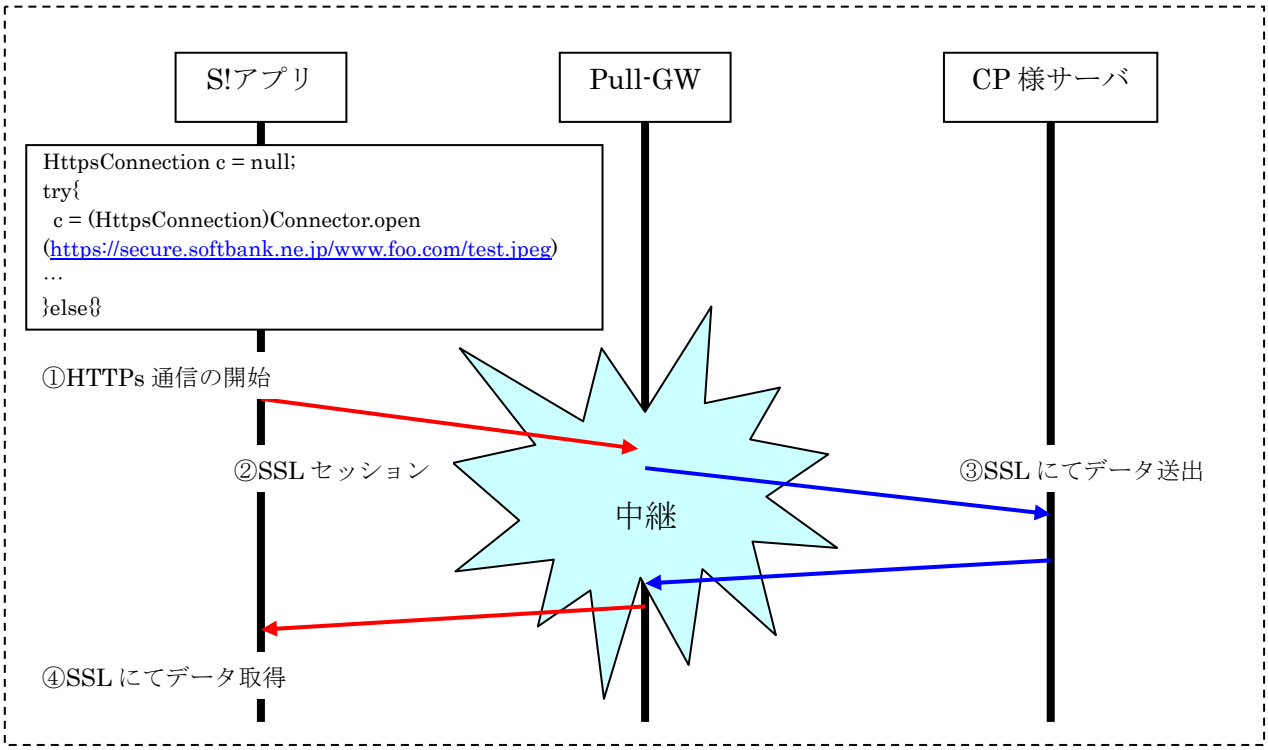
＜特殊な URL＞
HttpsConnection に下記 URL 指定する。

https://secure.softbank.ne.jp/<URL>
※ <URL>には CP 様 Web サーバを指定する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

＜SSL セッションの例＞

SSL セッションの中継の一例を以下に示す。



Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.3.5. Record Store 機能

MIDP 2.0 対応端末向け S!アプリでは MIDP 2.0 にて規定している Record Store 機能を利用できる。以下に MIDP 2.0 対応端末向け S!アプリで利用できる Record Store 機能について説明する。

- MIDP 1.0 対応端末向け S!アプリと同様、MIDlet-Data-Size 属性により Record Store のデータサイズを指定する。
- Record Store に対して、以下に記載する参照モードを指定できる。
  ① 共有可能（AUTHMODE_ANY）
  対応する S!アプリ（A）とは異なる S!アプリ（B）からも Record Store を利用することができる。
  ② 共有不可能（AUTHMODE_PRIVATE）
  対応する S!アプリ（A）のみで利用が可能であり、異なる S!アプリ（B）から Record Store を利用することはできない。
  Record Store を作成する時に参照モードを指定しない場合には、②共有不可として Record Store を作成する。
- 2 つの Record Store を同時に OPEN することが可能である。
- 1 つの S!アプリ（MIDlet Suite）内にある複数の MIDlet が、それぞれの Record Store へ互いにアクセスすることが可能である。
- １つの Record Store ファイルにつき、最低５つの Record Store が作成可能である。尚、Record 数については Record Store ファイルのサイズ内であれば制限はない。
- Record Store の書き込み処理中に一時停止、終了などの割り込み処理が入った場合、書き込み処理中のデータ内容は保証されない。
- 512KB 以上の Record Store を S!アプリで利用している場合、S!アプリを外部メモリカードから起動することができない場合がある。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.4.　S!アプリの継承

2.4.1. 外部メモリカードへの転送

MIDP 2.0 対応端末内部に保存した S!アプリは全て、外部メモリカードへの転送が可能となる。MIDP 1.0 対応端末では MIDlet 属性（MIDlet-Save）を記載することにより、外部メモリカードへの転送可否が設定できたが、MIDP 2.0 対応端末ではこの MIDlet 属性を無視する。
外部メモリカードへ転送する際には従来と同様､S!アプリを暗号化する｡ただし、以下点に注意すること。

- ◆ 暗号化の方式は各端末により異なる場合があるため、同一ユーザ間での起動の保証はできない。
- ◆ 外部メモリカードに保存の際、一部端末にて Jad ファイルを暗号化しない。Jad ファイルの記載には注意すること。

2.4.2. com.jblend.micro.lcdui 利用 S!アプリ

MIDP 2.0 対応端末では MEXA/JSCL にて規定している com.jblend.micro.lcdui の仕様が一部変更となった。このため、LocalizedTextField などを利用している MIDP 1.0 対応端末向け S!アプリを MIDP 2.0 対応端末向けに対応するためには、再度コンパイルをする必要がある。尚、MIDP 2.0 対応端末向けに対応した S!アプリは MIDP 1.0 対応端末にて正常に動作しない場合がある。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.4.3. 位置情報について

MIDP 2.0 対応端末では、位置情報取得が行えない端末がある。位置情報を取得する際に Exception が発生する可能性があるため、例外処理を行う必要がある。

2.4.4. S!アプリのアイコンについて

MIDP 2.0 対応端末では表示できるアイコンのサイズなどは端末実装依存となる。詳細は S!アプリ開発ガイド「端末情報（各種）」を参照のこと。

2.4.5. 独自スキームの対応について

MIDP 1.0 対応端末において openInputStream(“resource://～”)、open(“jar://～”)などにおいて「resource://」や「jar://」といった独自スキームを利用することができる。しかし、MIDP 2.0 対応端末ではこれらのスキームは端末実装依存となる。

2.4.6. S!アプリでの文字エンコードについて

MIDP 2.0 対応端末向け S!アプリでのデフォルト文字エンコードは利用した API により異なる。以下に各 API を利用した場合の文字エンコードについて記載する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

◆ MEXA/JSCL 利用 S!アプリ
・・・MIDlet 属性に「MIDxlet-API: JSCL-1.X.X」あるいは「MIDxlet-API: MEXA」と記載した S!アプリ

デフォルト文字エンコード　：SJIS

◆ JSR 利用 S!アプリ
・・・MIDlet 属性に「MIDxlet-API: JSCL-1.X.X」あるいは「MIDxlet-API: MEXA」と記載のない S!アプリ

デフォルト文字エンコード　：UTF-8 あるいは ISO8859-1

よって、String#getByte()を利用する場合には、String#getByte("UTF-8")のように明示的に文字エンコードを指定する必要がある。

2.4.7. S!アプリにおける URL の記載について

MIDP 2.0 対応端末向けに S!アプリを提供する場合、下記ケースにて記載する URL には弊社固有のクエリを記載してはならない。

◆ MIDlet 属性（MIDlet-Info-URL、MIDxlet-Slave-Application など）
◆ S!アプリからの HTTP 通信（HTTPs 通信を含む）
◆ MEXA/JSCL におけるネイティブのブラウザ起動
◆ MIDP 2.0 におけるネイティブのブラウザ起動

2.4.8. MIDlet 属性の変更

MIDP 2.0 では、規定していない MIDlet 属性（Option 属性）を利用する場合、「MIDlet-XXX」と記載してはならず、「MIDxlet-XXX」と記載する必要がある。よって、MIDP 1.0 対応端末向けに提供している S!アプリを MIDP 2.0 対応端末（X）向けに提供するためには、一部 MIDlet 属性を変更する必要がある。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

表 2.4.8-1. 変更する MIDlet 属性一覧

変更前	変更後
MIDlet-OCL	MIDxlet-API
MIDlet-Resident	MIDxlet-Resident
MIDlet-Application-Range	MIDxlet-ScreenSize
MIDlet-Aux	MIDxlet-Aux
MIDlet-Karaoke	MIDxlet-Karaoke
MIDlet-Network	MIDxlet-Network
MIDlet-Serial	MIDxlet-Serial
MIDlet-Save	MIDxlet-Save
MIDlet-Application-Security	MIDxlet-Application-Security
MIDlet-RemoteControl	MIDxlet-RemoteControl
MIDlet-Resource	MIDxlet-Resource
MIDlet-Copyright	MIDxlet-Copyright

※ 灰色の属性については MIDP 2.0 対応端末では対応をしていないため無視をする。

また、S!アプリの利用画面サイズを指定する属性（MIDlet-Application-Range、MIDxlet-ScreenSize）のデフォルト値は MIDP 1.0 対応端末と MIDP 2.0 対応端末、あるいは端末の液晶サイズによって異なる。このため、下記のとおり変更を行なうこと。(本変更は MIDP1.0、MIDP2.0 ともに注意して記載すること)

表 2.4.8-2. 変更する MIDlet 属性値一覧

S!アプリ 利用画面サイズ	変更前	変更後
120 x 130	MIDlet-Application-Range 記載なし	MIDxlet-ScreenSize: 120,130
240 x 260	MIDlet-Application-Range: 0,0	MIDxlet-ScreenSize: 240,260
	MIDxlet-ScreenSize 記載なし	
	MIDxlet-ScreenSize: 0,0	

※ Vodafone 802SE 向けに S!アプリを提供する場合、以下のように変更すること。

MIDxlet-ScreenSize: 176,182

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

2.5. S!アプリの開発

MIDP 2.0 対応端末向け S!アプリの開発には、弊社より提供するツールを利用することができる。

◆ MEXA/JSCL 利用 S!アプリ、JSR 利用 S!アプリ
本アプリを開発するためには S!アプリ開発支援ツール（MEXA SDK）を利用すること。

※ 上記以外のツールを利用した際の動作については動作保障対象外となる。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

Appendix A. VSCL API

本章では独自仕様である VSCL について説明する。
VSCL では以下パッケージについて規定している。

表 A-1. API 一覧

パッケージ	概要
com.vodafone.io	リモコンにより操作可能な機器に対する赤外線コマンド送信用パッケージ。
com.vodafone.lcdui	LCDUI 上での立体描画用パッケージ。SVGImage クラスは、W3C SVG 仕様 1.1 のモバイル向けサブセットであり、画像のズーム/パン/回転をサポートする。
com.vodafone.media.barcode	バーコード認識およびバーコードデータ取得するためのパッケージ。
com.vodafone.midlet	待受アプリを作成するための基底クラスが入っている。ResidentMIDlet クラスを継承して待受アプリを作成することが可能である。
com.vodafone.system	イベントおよびデバイス機能の制御を受け取るためのパッケージ。
com.vodafone.util	lcudi.Image オブジェクトを PNG または JPEG 形式に変換するためのパッケージ。
com.vodafone.media.audio3d	ヘッドフォンまたはステレオスピーカー付きハンドセットを使った 3D ポジショナルオーディオを実装するパッケージ

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

A.1. com.vodafone.io パッケージ

このパッケージには、リモコンにより操作可能な機器に対する赤外線コマンド送信用のクラスが入っている。

表 A.1-1. クラス

名前	概要
RemoteControl	赤外線データを送信してリモコン機能を実現するためのクラスである。
RemoteControlData	赤外線送信するコマンドデータおよびパラメータを設定するクラスである。

A.2. com.vodafone.lcdui パッケージ

このパッケージには、W3C SVG 仕様 1.1 のモバイル向けサブセットであり、画像のズーム/パン/回転をサポートする SVGImage クラスが入っている。

表 A.2-1. クラス

名前	概要
SVGImage	S!アプリにて SVG Tiny 1.1 画像の操作および描画を可能とする。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

A.3. com.vodafone.media.barcode パッケージ

このパッケージには、バーコードリーダ機能を実現するためのインタフェースが含まれている。

表 A.3-1. インタフェース

名前	概要
BarcodeControl	バーコードデータの認識、取得を制御するためのインタフェースである。

A.4. com.vodafone.midlet パッケージ

このパッケージには、待受アプリを作成するための基底クラスが入っている。

表 A.4-1. クラス

名前	概要
ResidentMIDlet	待受アプリの為の基底クラスである。このクラスを継承して待受アプリを作成することが可能である。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

A.5. com.vodafone.system パッケージ

このパッケージには、端末に固有なデバイス機能に関するクラス及びインタフェースが含まれている。

表 A.5-1. インタフェース

名前	概要
BodyOpenListener	折畳式端末の開閉状態を通知するリスナーを実装するためのインタフェースである。

表 A.5-2. クラス

名前	概要
DeviceControl	端末に固有なデバイスを制御する機能を提供するためのクラスである。

A.6. com.vodafone.util パッケージ

このパッケージには、Image オブジェクトを PNG または JPEG 形式に変換するためのクラスが含まれている。

表 A.6-1. クラス

名前	概要
ImageEncoder	Image オブジェクト上の画像データを PNG または JPEG 形式にエンコードするためのクラスである。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

A.7. com.vodafone.media.audio3d パッケージ

このパッケージには、3D Sound API でステレオヘッドフォンまたはステレオスピーカーを利用し 3D ポジショナルオーディオを実現するインタフェース/クラスが含まれている。

表 A.7-1. インタフェース

名前	概要
Audio3DControl	このインタフェースは、オーディオソースの 3 次元の配置を制御する。
ExtendedAudioControl	オーディオ再生に空間効果を加える。
ReverbControl	オーディオプレイヤに対するリバーブ効果を有効にする。

表 A.7-2. クラス

名前	概要
Environment3D	このクラスは、3D オーディオ処理のリスナー及びオーディオ環境を表す。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

Appendix B. VSCL と MEXA/JSCL の違い

VSCL は MEXA/JSCL より移植した機能を含む。しかし、一部機能において変更があり、全く同じ機能ではない。本章では VSCL と MEXA/JSCL 違いについて説明をする。

B.1. イベントリスナー

端末でイベントが発生した場合、S!アプリでは発生したイベントを取得可能である。本章では VSCL と MEXA/JSCL で取得可能なイベントの違いについて示す。

表 B.1-1. イベントリスナーの違い

分類	イベント	概要	対応	
			VSCL	MEXA/JSCL
電話	着信開始	着信時、以下の情報を取得 ・名前 ・電話番号	○	○
	通話終了	電話が切れたことを取得	○	○
メール	メール受信/続きメール受信	メール送信/続きメール受信完了時、結果（成功/失敗/中断/一部失敗/不明）を取得	×	○
	メール受信	メール受信完了時、以下の情報を取得 ・名前 ・メールアドレス ・種類（SMS,MMS など）	○	○
端末開閉	端末開	閉じた端末が開いたことを取得	△	△
	端末閉	開いた端末が閉じたことを取得	△	△
位置情報	位置情報更新	位置情報の更新完了時、結果（成功/失敗）を取得	×	○
端末内データ	送信メールボックス変更	送信メールボックス内のデータが変更したことを取得	×	○

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

	受信メールボックス変更	受信メールボックス内のデータが変更したことを取得	×	○
	アドレス帳変更	アドレス帳のデータが変更したことを取得	×	○
	データフォルダ変更	データフォルダ内のデータが変更したことを取得	×	○
外部メモリ	外部メモリ装着	外部メモリカードを装着したことを取得	×	△
	外部メモリ取り外し	外部メモリカードを取り外したことを取得	×	△
アラーム	アラーム通知	アラーム時、設定したコメントを取得	○	○

［凡例］

○：全ての端末にて取得可

△：対応端末にて取得可

×：取得不可

B. 2. 画像エンコード

S!アプリにて JPEG、PNG ファイルフォーマットへのエンコードを行うことができる。本章では VSCL と MEXA/JSCL での画像エンコード機能の違いについて示す。

表 B.2-1. 画像エンコード機能の違い

フォーマット	圧縮レベル	概要	対応	
			VSCL	MEXA/JSCL
JPEG	6Kbytes	6Kbytes 以内のデータサイズに JPEG 圧縮を行う	○	○
	12Kbytes	12Kbytes 以内のデータサイズに JPEG 圧縮を行う	○	○
	30Kbytes	30Kbytes 以内のデータサイズに JPEG 圧縮を行う	○	○
	100Kbytes	100Kbytes 以内のデータサイズに JPEG 圧縮を行う	○	×
	200Kbytes	200Kbytes 以内のデータサイズに JPEG 圧縮を行う	○	×
	300Kbytes	300Kbytes 以内のデータサイズに JPEG 圧縮を行う	○	×

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

	ノーマル	QVGA写メールモードのノーマル相当の圧縮率でJPEG圧縮を行う	×	○
	ファイン	画質優先でJPEG圧縮を行う	×	○
	画質優先	画質優先でJPEG圧縮を行う	○	×
PNG	－	PNG圧縮を行う	△	△

［凡例］

○：全ての端末にて対応

△：一部端末にて対応

×：非対応

B. 3. デバイス制御

S!アプリより以下の端末制御を行うことができる。本章ではVSCLとMEXA/JSCLにて可能な端末制御機能の違いについて示す。

表 B.3-1. デバイス制御機能の違い

項目	内容	対応	
		VSCL	MEXA/JSCL
バックライト制御	バックライトを点滅させる	×	○
バッテリー状態	バッテリー残量を取得する	○	○
電界状態	電界強度を取得する	○	○
端末開閉状態	端末の開閉状態を取得する	△	△
外部メモリカード状態	装着状態を取得する	×	○
Key State	キーの押下状態を取得する	×	○
オートリピート	キーのオートリピート状態を取得する	×	○
緯度取得	緯度を取得する	×	○
経度取得	経度を取得する	×	○
新着状態	不在着信/新着メールの有無を取得する	×	○
バイブレーション動作状態	バイブレーションが動作しているかを取得する	×	○
バックライト動作状態	バックライトが動作しているかを取得する	×	○
８方向キー状態	８方向キーモードであるかを取得する	×	○

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

［凡例］
○：全ての端末にて取得可
△：対応端末にて取得可
×：取得不可

B. 4. バーコード認識

MIDP 2.0 対応端末（X）、JSCL-1.2.X/JSCL-1.3.2 対応端末では、バーコード認識を行うことができる。また、MIDP 2.0 対応端末では VSCL を利用し、MMAPI を利用したバーコード認識を行うことができる。（ただし、バーコード認識に対応した端末のみ）本章では VSCL と MEXA/JSCL にて可能なバーコード機能の違いについて示す。

＜対応バーコード形式＞
VSCL と MEXA/JSCL にて対応するバーコード形式が異なる。以下、その違いについて示す。

表 B.4-1. バーコード形式の違い

種類	Type	対応	
		VSCL	MEXA/JSCL
1 次元	EAN	△	×
1 次元	JAN	△	△
1 次元	UPC	△	×
2 次元	Datamatrix	△	×
2 次元	QR	○	○

［凡例］
○：全ての端末にて取得可
△：対応端末にて取得可
×：取得不可

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

＜読み取りモード＞

S!アプリから以下のようなバーコード読み取りモードが指定できる。以下にVSCL と MEXA/JSCL にて指定可能な読み取りモードの違いについて示す。

表 B.4-2. 読み取りモードの違い

種類	対応	
	VSCL	MEXA/JSCL
1 次元コード 1 回のみ読み取り	○	○
1 次元コード連続読み取り	×	○
2 次元コード 1 回のみ読み取り	○	○
2 次元コード連続読み取り	×	○
1 次元/2 次元指定なし（自動認識）1 回のみ読み取り	○	○

［凡例］

○：全ての端末にて指定可
　　ただし、対応するバーコード形式のみ読み取りが可能

×：指定不可

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

Appendix C. MIDP 2.0 対応端末と MIDP 1.0 対応端末における MEXA/JSCL の違い

MIDP 2.0 対応端末（X）では弊社が独自に拡張した仕様（MEXA/JSCL）を搭載する。MEXA/JSCL は MIDP 1.0 対応端末にて搭載した仕様である。しかし、一部機能において変更があり、MIDP 2.0 対応端末（X）と MIDP 1.0 対応端末にて全く同じ機能を搭載しているわけではない。本章では、MIDP 2.0 対応端末（X）と MIDP 1.0 対応端末における MEXA/JSCL 機能の違いを説明する。
また、MEXA/JSCL 利用 S!アプリ（MIDP 2.0 対応端末向け）作成における注意点を説明する。

C.1. 音声認識

MEXA/JSCL では音声認識機能を実現するための API を規定している。しかし、音声認識機能は MIDP 2.0 対応端末（X）では端末実装依存となっている。本機能は MIDP 2.0 対応端末（X）の一部端末にて利用できる機能となる。

C.2. 外部出力

JSCL-1.1.X 以降対応端末では S!アプリを TV などに出力する外部出力の機能を利用することができる。本機能は MIDP 2.0 対応端末（X）では端末実装依存となっている。本機能は MIDP 2.0 対応端末（X）の一部端末にて利用できる機能となる。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

C.3. S!カラオケ

JSCL-1.2.2/JSCL-1.3.2 対応端末では S!カラオケの機能を利用することができる。本機能は MIDP 2.0 対応端末(X)では端末実装依存となっている。本機能は MIDP 2.0 対応端末（X）の一部端末にて利用できる機能となる。

C.4. シリアル制御

シリアル制御対応の MIDP 1.0 対応端末ではシリアル制御機能を利用することができる。本機能は MIDP 2.0 対応端末（X）では一切利用できない。

C.5. ブラウザ起動

MEXA/JSCL ではネイティブのブラウザを起動して指定の URL にアクセスできる機能を利用することができる。本機能は MIDP 2.0 対応端末（X）でも利用できる。
ただし、ブラウザ起動を利用する場合には URL に弊社固有のクエリを利用してはならない。

C.6. 自動拡大表示機能

MIDP 2.0 対応端末では S!アプリを自動的に拡大する機能を実装しない。このため、高精細画面に対応していない S!アプリを拡大して表示するためには MIDxlet-ScreenSize の属性を利用する必要がある。MIDxlet-ScreenSize については 2.2.8.11. MIDxlet-ScreenSize を参照のこと。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

C.7. ファイルシステム機能

MEXA/JSCL では StorageConnection インタフェースを利用し、端末内部のデータフォルダ及び外部メモリカード上のファイルにアクセスできる。MIDP 2.0 対応端末（X）における本機能の注意点について以下に記載する。

＜フォルダ作成＞

JSCL-1.2.X 以降対応端末ではデータフォルダ内に自由にフォルダ/ファイルを作成することが可能であるが、MIDP 2.0 対応端末（X）では自由にフォルダ/ファイルを作成することができない。ファイルシステム機能を利用するためには S!アプリ開発ガイド「端末情報（各種)」に記載したパスに作成した独自フォルダ/ファイルを利用することを推奨する。

＜フォルダ削除/フォルダ名変更＞

JSCL-1.2.X 以降対応端末ではデータフォルダ内に作成したフォルダの削除/フォルダ名の変更を自由に行うことが可能であるが、MIDP 2.0 対応端末（X）ではフォルダ内にサブフォルダやファイルが存在する場合にはフォルダの削除/フォルダ名の変更を行なうことができない。よって、MIDP 2.0 対応端末（X）向け S!アプリにてファイルシステム機能を利用し、S!アプリにて利用するフォルダの削除/フォルダ名の変更を行う場合、フォルダ内のサブフォルダやファイルを全て削除してから行なうこと。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

C.8. SpriteCanvas#paint()での Graphics オブジェクトの初期状態

Canvas クラスの paint()メソッドに渡す Graphics オブジェクトについて、paint()を呼び出した時点でのクリップ領域に何らかの仮定を設けて明示的にクリップ領域を設定していない場合、意図したとおりの画面描画を行なわない場合がある。paint()メソッドの先頭など、必要なタイミングで Graphics#setClip()を実行する必要がある。

例）

```
public void paint(Graphics g) {
  g.setClip(0, 0, getVirtualWidth(), getVirtualHeighit() );
   :
```

特に、SpriteCanvas を使用し、実画面サイズを越える（仮想画面）領域への描画を行なう場合、描画メソッドを実行する前に適切にクリップ領域を設定する必要がある。

C.9. SpriteCanvas での Canvas 継承メソッドの使用

SpriteCanvas を継承するクラスでは、Canvas クラスで MIDP 2.0 から使用可能となったメソッドのうち、以下のものは継承できない。

- public Ticker getTicker()
- public String getTitle()
  super メソッドを呼び出した場合、null を返す。
- public void setTicker(Ticker ticker)
- public void setTitle(String title)
  super メソッドを呼び出した場合、何も行なわない。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

C.10. SpriteCanvas でのフルスクリーンモードの使用

SpriteCanvas においても MIDP 2.0 にて規定している Canvas クラスの setFullScreenMode(mode)メソッドにより、フルスクリーンモードを使用することが可能である。ただし、サイズの変化を把握した描画処理を適宜行なう必要がある。

C.11. SpriteCanvas と MIDP2.0 Game パッケージとの併用

MIDP 2.0 にて規定している Layer(Sprite, TiledLayer)オブジェクトの上に SpriteCanvas の描画を行なうなど、GameCanvas および Layer と SpriteCanvas の同時使用については動作保証できない。

C.12. 2D スプライト描画と 3D ポリゴン描画の重ね合わせ

SpriteCanvas 上に MEXA/JSCL の 3D 描画を行なうことはできない。MIDP 2.0 にて規定している GameCanvas 上には以下のような方法で MEXA/JSCL の 3D 描画を行なうことが可能である。

例)

```
public Class MyGameCanvas extends GameCanvas {
  :
  public void main_loop() {

    // Obtains Graphics object of Drawing Buffer of GameCanvas
    Graphics g = getGraphics();

    // Draws any Layers
    // Draws 3D Object on Top
    (( Graphics3D )g ).drawFigure(...);

    // Flush Drawing Buffer to Real Screen
    flushGraphics();
  }             }
```

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

C.13. 端末機能・設定によるデバイス操作メソッドへの影響

端末の S!アプリ設定により、S!アプリによるバイブおよびバックライトの操作を禁止している場合、Display オブジェクトの vibrate()および flashBacklight()の戻り値は false となる。また、DeviceControl オブジェクトの setDeviceActive(VIBRATION or BACK_LIGHT, true or false)の戻り値は true（ただし、実際の端末制御は行なわない）となる。

C.14. DeviceControl によるバックライト常時 ON/常時 OFF

DeviceControl の blink()を、バックライトの常時 ON/常時 OFF のために使用することはできない。バックライトの常時 ON/常時 OFF を行なうためには、DeviceControl の setDeviceActive()を使用すること。

C.15. DeviceControl と MIDP 2.0 Game パッケージの併用

DeviceControl の getDeviceState(KEY_STATE) と GameCanvas の getKeyState()は同一の S!アプリで使用が可能である。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

C.16. MediaPlayer、PhrasePlayer、MMAPI Player の再生における競合

(com.j_phone.media、com.jblend.media)MediaPlayer、(com.j_phone.amuse 、com.jblend.media.smaf.phrase)PhrasePlayer、MMAPI の Player では再生時に利用するリソースの取得が競合する。すでにリソースを確保している場合、後発のメソッドにてリソースを確保すると例外が発生する。

◆ com.j_phone.amuse.PhrasePlayer

⇔ com.jblend.media.smaf.phrase.PhrasePlayer 呼び出し

com.j_phone.amuse.PhrasePlayer,com.jblend.media.smaf.phrase.PhrasePlayer 間では、どちらかの PhrasePlayer を取得した段階で、もう一方の PhrasePlayer が取得できない。

表 C.16-1. PhrasePlayer における競合

			com.j_phone.amuse				
			Phrase Player getPlayer	Phrase Player getTrack	Phrase Track setPhrase	Phrase Track start	Phrase Track pause
com. jblend. media. smaf. phrase	Player 状態	Plyer 取 得 状態	Runtime Exception	×	×	×	×
	ﾄﾗｯｸ 状態	停止	×	×	×	×	×
		再生	×	×	×	×	×
		一 時 停止	×	×	×	×	×

[凡例]

×： 実行不可能

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

◆ MMAPI Player ⇔ PhrasePlayer 呼び出し

MMAPI Player、PhrasePlayer 間では以下のような例外が発生する。

（1） MMAPI Player → PhrasePlayer 呼び出し

表 C.16-2. MMAPI Player → PhrasePlayer における競合

		PhrasePlayer				
		Phrase Player getPlayer	Phrase Player getTrack	Phrase Track setPhrase	Phrase Track start	Phrase Track pause
MMAPI Player 状態	NO_DATA	○	○	○	○	○
	REALIZED	○	○	○	○	○
	PREFECTCHED	○	○	Runtime Exception	Runtime Exception	Runtime Exception
	STARTED	○	○	Runtime Exception	Runtime Exception	Runtime Exception

[凡例]

○： 実行可能

（2） PhrasePlayer → MMAPI Player 呼び出し

表 C.16-3. PhrasePlayer → MMAPI Player における競合

			MMAPI Player				
			create Player	realize	prefetch	start	stop
Phrase Player 状態	Player 状態	Plyer 取 得 状態	○	○	○	○	○
	ﾄﾗｯｸ 状態	停止	○	○	Media Exception	×	×
		再生	○	○	Media Exception	×	×
		一 時 停止	○	○	Media Exception	×	×

[凡例]

○： 実行可能

×： 実行不可能

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

◆ com.j_phone.media.MediaPlayer ⇔ com.jblend.media.MediaPlayer
呼び出し

com.j_phone.media.MediaPlayer、com.jblend.media.MediaPlayer 間では、play を呼び出し、再生状態になるとリソースを確保する。よってどちらかの MediaPlayer が再生状態にある場合、もう一方の MediaPlayer で play を呼ぶと下記例外が発生する。

表 C.16-4. MediaPlayer における競合

		com.jblend.media.MediaPlayer				
		Phrase Player getPlayer	Phrase Player getTrack	Phrase Track setPhrase	Phrase Track start	Phrase Track pause
com. j_phone. media. MediaPlayer	停止中	○	○	○	○	○
	再生中	○	○	Runtime Exception	○	○
	一時停止	○	○	○	○	○

[凡例]
○： 実行可能

C.17. 一時停止/再開における MediaPlayer、PhrasePlayer、MMAPI Player の再生再開位置

MediaPlayer、PhrasePlayer、MMAPI Player での再生中に一時停止/再開した時、再開の際には一時停止時の位置から再生を再開する。ただし、一時停止/再開後の動作においては端末実装依存となるため、音声データによっては再開の際には先頭から再生する場合がある。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

C.18. 着信時に使用できる MediaPlayer,PhrasePlayer 機能

MIDP 2.0 対応端末（X）と JSCL-1.1.X 以降対応端末では着信時に使用可能な MediaPlayer、PhrasePlayer の機能が異なる。以下に着信時に使用できる MediaPlayer、PhrasePlayer 機能の違いを示す。

表 C.18-1. 着信時に使用できる MediaPlayer、PhrasePlayer 機能の違い

クラス	メソッド	MIDP 2.0 対応端末（X）	JSCL-1.1.X 以降対応端末
com.j_phone.amuse.PhrasePlayer	kill()	Runtime Exception	正常終了
com.j_phone.amuse.Phrase	Phrase (byte[] data)	正常終了	Runtime Exception
	Phrase (java.lang.String url)	正常終了	Runtime Exception
com.jblend.media.smaf.phrase.PhrasePlayer	kill()	Runtime Exception	正常終了
com.jblend.media.smaf.phrase.Phrase	Phrase (byte[] data)	正常終了	Runtime Exception
	Phrase (java.lang.String url)	正常終了	Runtime Exception
com.jblend.media.smaf.phrase.AudioPhrase	AudioPhrase (byte[] data)	正常終了	Runtime Exception
	AudioPhrase (java.lang.String url)	正常終了	Runtime Exception

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

Appendix D. MIDP 2.0 対応端末と MIDP 1.0 対応端末の動作差異

MIDP 2.0 対応端末では MIDP 2.0 及び各端末実装依存のプラットフォームを実装している。MIDP 1.0 対応端末に実装していた MIDP 1.0 及び各端末共通のプラットフォームと動作が異なる場合があり、S!アプリ作成において注意が必要となる。本章では、MIDP 2.0 対応端末と MIDP 1.0 対応端末における動作差異を説明する。

D.1. repaint()から paint()への流れについて

```
repaint();
// (A)
```

(A)の部分のコードが実行される時点で repaint()を契機とした画面更新が終わっていることは保証できない。 (A)の部分に画面更新が終わったことを前提とするコードが必要な場合は、repaint() と(A)のコードの間に serviceRepaints()を呼び出して repaint()による画面更新の実行を待ち合わせる必要がある。

D.2. Display#setCurrent()の同期/非同期について

MIDP 1.0 対応端末では setCurrent() は同期メソッドとして実装されていたが、MIDP 2.0 対応端末では非同期メソッドとして実装している。

例１）

```
  // (A)
  setCurrent(d);
  // (B)
```

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

◆ MIDP 1.0 対応端末（同期実装）の場合：
(A) カレント切り替え前を前提としたコード
(B) カレント切り替え後を前提としたコード

◆ MIDP 2.0 対応端末（非同期実装）の場合：
(A) カレント切り替え前を前提としたコード
(B) カレント切り替え前を前提としたコード

MIDP 2.0 対応端末の実装(非同期実装)では、(B)の部分のコードが実行される時点で setCurrent()を契機としたカレント切り替えが終わっていることは保証しない。 つまり、(B)の部分のコードは、カレントがまだ切り替わってない事を前提としたコードでなければならない。カレント切り替え後に実行する必要のあるコードは、showNotify()が呼び出された以降、または isShown()が true を返す状態になってから実行されるように記述する必要がある。

例２）

```
setCurrent(d);
serviceRepaints();          // (C)
```

◆ MIDP 1.0 対応端末（同期実装）の場合：
(C)カレント切り替えに伴って VM が repaint()を呼び出すことを前提とした serviceRepaints()によりカレント切り替えの待ち合わせが可能。

◆ MIDP 2.0 対応端末（非同期実装）の場合：
(C)serviceRepaints()でのカレント切り替えの待ち合わせはできない。

(C)が実行される時点で カレント切り替え後，repaint()呼び出し済みであることは保証されない。よって、(C)の serviceRepaints()で画面更新まで処理がブロックすることは期待できない。 このように setCurrent()によるカレント切り替えを待ち合わせるには、以下のような待ち合わせコードが必要です。

```
setCurrent( d );
while( !d.isShown() ){
  Thread.sleep(100);
}
```

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

D.3. Displayable.isShown()が true になるタイミング

Display.setCurrent(Displayable) 後に Displayable.isShown() が true になるタイミングは以下のようになる。

◆ MIDP 1.0 対応端末：
setCurrent() を抜けた直後

◆ MIDP 2.0 対応端末：
setCurrent() 後の初回の paint() 実行直後

D.4. keyPressed()/keyReleased()/keyRepeated()での判定

MIDP 1.0 対応端末では以下のようなコードで GameAction に該当するキーの判定ができたが、MIDP 2.0 対応端末では期待どおりに判定できない。

```
protected void keyPressed(int keyCode) {
  if(keyCode == UP) {
    System.out.println ("keyPressed("+ keyCode +"): UP");
  }
  else if(keyCode == FIRE) {
    System.out.println ("keyPressed("+ keyCode +"): FIRE");
  }
}
```

正しいキー判定をするためには、S!アプリで Canvas#getGameAction(keycode) および getKeyCode(gameAction) により GameAction とキーコードの対応を取得する必要がある。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

例１）

```
protected void keyPressed(int keyCode) {
  int gameAction = getGameAction (keyCode);
  if(gameAction == UP) {
    System.out.println ("keyPressed("+ keyCode +"): UP");
  }
  else if(gameAction == FIRE) {
    System.out.println ("keyPressed("+ keyCode +"): FIRE");
  }
}
```

下記方法は間違っているわけではないが、複数のキーに同じ GameAction が割り当てられているプラットフォームの場合、意図しない振る舞いになる可能性がある。

例２）

```
protected void keyPressed(int keyCode) {
  if(keyCode == getKeyCode(UP)) {
    System.out.println ("keyPressed("+ keyCode +"): UP");
  }
  else if(keyCode == getKeyCode(FIRE)) {
    System.out.println ("keyPressed("+ keyCode +"): FIRE");
  }
}
```

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

D.5. Graphics クラスのプリミティブ描画メソッドでの Anchor Point 指定

Anchor Point 指定する際、下記に注意すること。

◆ MIDP 1.0 対応端末：
縦方向だけ、または横方向だけの指定でも描画が可能。

```
g.drawChar ('A', 0, 0, Graphics.LEFT);
g.drawChars (chars, 0, 5, getWidth() / 2, 60, Graphics.HCENTER);
```

また、Graphics.VCENTER は Graphics#drawImage()以外の文字列描画系メソッド（drawString()、drawSubstring()、drawChar()、drawChars()）にも指定可能である。

◆ MIDP 2.0 対応端末：
縦方向だけ、または横方向だけの指定では IllegalArgumentException が発生する。描画するには、縦方向・横方向の両方を指定する必要がある。

```
g.drawChar ('A', 0, 0, Graphics.LEFT | Graphics.TOP);
```

また、Graphics.VCENTER を指定可能なメソッドはイメージ描画系のメソッド（drawImage()、copyArea()、drawRegion()）のみであり、文字列描画系メソッド（drawString()、drawSubstring()、drawChar()、drawChars()）に指定した場合には、IllegalArgumentException が発生する。

D.6. openRecordStore()の引数が不正な場合の例外

MIDP 1.0 対応端末では openRecordStore()の recordStoreName が不正な場合には RecordStoreException が発生するが、MIDP 2.0 対応端末では IllegalArgumentException が発生する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

D.7. 表示系のコールバックメソッドについて

MIDP 1.0 対応端末では表示系のコールバックメソッドは、以下のスレッドから呼んでいた。しかし、MIDP 2.0 対応端末では全て同一のスレッド（イベントスレッド）から呼ぶ。 したがって、下記のリスト中の表示系のコールバックを処理している途中、別の表示系のコールバックメソッドを呼ぶことはない。

◆ paint スレッド
Canvas#paint

◆ callSerially スレッド
Display#callSerially に登録した、Runnable#run

◆ イベントスレッド
Canvas#keyPressed
Canvas#keyRepeated
Canvas#keyReleased
Canvas#pointerPressed
Canvas#pointerDragged
Canvas#pointerReleased
CommandListener#commandAction

◆ Display#setCurrent が呼ばれたスレッド
Canvas#showNotify
Canvas#hideNotify

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

D.8. Canvas#paint()中のセキュリティダイアログ

Canvas クラスを継承した S!アプリで paint()メソッドを実行中、セキュリティダイアログ表示対象 API（HTTP/HTTPs 通信など）を実行した場合、セキュリティダイアログの表示前に実行した描画メソッドの結果を破棄する。

例）

```
protected void paint( Graphics g ) {
  // ここから
  g.setclip( 0, 0, 100, 100 );
  g.setColor( 0x0ffffff );
  g.fillRect( 0, 0, getWidth(), getHeight() );
  // ここまでの描画結果は無効になる
  try{
    HttpConnection conn =
      ( HttpConnection )Connector.open( url, Connector.READ, true );
    // セキュリティダイアログ表示
    try {
      conn.setRequestMethod(HttpConnection.GET);
      InputStream in = conn.openInputStream();
                        :
    } finally {
      conn.close();
    }
  }
  catch( SecurityException e ){
    // ユーザが HTTP 通信を許可しなかった場合の処理を記述
  }
    // 以降の描画結果は有効
  g.drawLine( 0, 0, 20, 30 );
  g.drawString( "Hello world!", 20, 20, Graphics.TOP|Graphics.LEFT );
                        :
}
```

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

セキュリティダイアログ表示対象 API（SecurityException を送出する可能性のあるメソッド）を Canvas 継承クラスの paint()メソッド内で使用する場合、S!アプリでの描画は、セキュリティダイアログ表示対象 API の後に実行するように記述すること。また、その際には try～catch により、SecurityException を適切に処理すること。

D.9. RecordStore の書き込みタイミングについて

MIDP 1.0 対応端末では RecordStore への書き込み内容を一旦ヒープ領域にて保持し、S!アプリの終了時に端末内部メモリに書き込みを行なっていた。MIDP 2.0 対応端末では RecordStore への書き込み内容をヒープ領域にて保持せず、随時端末内部メモリへの書き込み処理を開始する。ただし、端末内部メモリへの書き込み完了タイミングについては端末実装依存となるため注意すること。

D.10. Image#createImage()の利用について

MIDP 1.0 対応端末では Image#createImage()を利用する際、相対パスにて指定することが可能となっていた。しかし、MIDP 2.0 対応端末では絶対パスにて指定する標準の方法となる。

＜MIDP 1.0 対応端末にて動作可能＞
Image#createImage(image);

＜MIDP 2.0 対応端末にて動作可能＞
Image#createImage(/image);

D.11. TextField、TextBox の NUMERIC 時の“-”入力

TextField、TextBox にて NUMERIC を利用している場合、MIDP 1.0 対応端末では“-”の入力が可能であるが、MIDP 2.0 対応端末では Exception が発生する。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved

D.12. MIDlet セレクタの振る舞いについて

S!アプリを MIDlet スイートにて作成した場合、下記に注意すること。

◆ MIDP 1.0 対応端末：

- MIDlet#notifyPaused()を呼び出した場合、実行は MIDlet セレクタに遷移する。
- MIDlet が Background 状態で Display#setCurrent()を呼び出した場合、その MIDlet は Foreground 状態になる。
- 表示を破棄した MIDlet をユーザが MIDlet セレクタで選択した場合、その MIDlet の MIDlet#startApp()を呼び出す。

◆ MIDP 2.0 対応端末：

- MIDlet#notifyPaused()を呼び出した場合、VM は一時停止する。
- MIDlet が Background 状態で Display#setCurrent()を呼び出しても、その MIDlet は Foreground 状態にならない。
  表示を破棄した MIDlet をユーザが MIDlet セレクタで選択しても、その MIDlet の MIDlet#startApp()は呼び出さない。

Copyright© 2008 SOFTBANK MOBILE Corp. All rights reserved