MetaMoJi Share / Note for Business Ver.3 カスタムURLスキーム仕様情報

- Microsoftは、米国Microsoft Corporationの米国およびその他の国における登録商標です
- Google Driveは、Google Inc.の登録商標です。
- 本書は株式会社MetaMoJiが作成したものであり、マニュアルの著作権は、株式会社MetaMoJiに帰属します。
- 本書の内容は予告なく変更することがあります。
  2017年8月9日 ©2017 株式会社MetaMoJi

MetaMoJi

はじめに

- 本書では、iOS上のほかのアプリケーションとMetaMoJi Share / Note for Businessとの連携をスムーズに行うための仕様を説明しています。

- 連携は、iOS環境におけるカスタムURLスキームの仕組みを利用して行います。

- 本書では、製品名をそれぞれ次のように表記しています。

MetaMoJi Share for Business	Share
MetaMoJi Note for Business	Note

できること

1. ほかの（呼び出し元）アプリケーションから取り込む（インポート）

- Share / Noteに取り込めるファイル形式
  - PDFファイル（.pdf）
  - Microsoft Office製品のファイル（.xls、.xlsx、.doc、.docx、.ppt、.pptx）
  - 画像ファイル（.png、.gif、.jpg、.jpeg）
  - テキストファイル （.txt）

2. シェアノートを呼び出し元アプリケーションに受け渡す（会議開催時）

3. ほかの（呼び出し元）アプリケーションから開く

- 開くことができるファイル形式
  - MetaMoJi Share 形式（.btshare）/ MetaMoJi Note 形式（.atdoc）

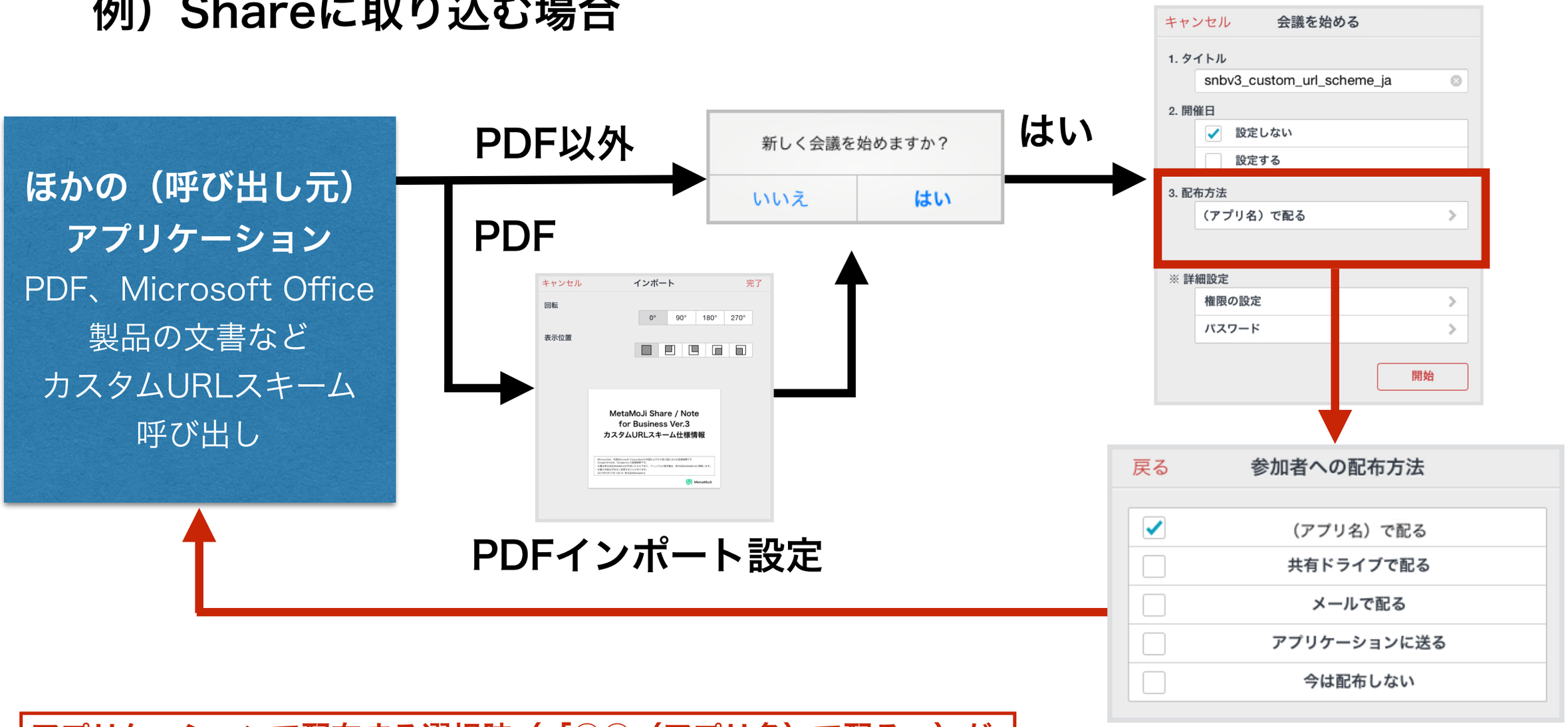
4. 呼び出し元アプリケーションに送る

- 送ることができるファイル形式
  - PDF 形式 / MetaMoJi Note 形式（.atdoc）

5. Microsoft Office製品のファイルの取り込みにはGoogle DriveのPDF変換機能を利用するため、Google Driveのアカウントが必要になります。

1. ほかの（呼び出し元）アプリケーションから取り込む(インポート)
2. シェアノートを呼び出し元アプリケーションに受け渡す（会議開催時）

例）Shareに取り込む場合



カスタムURLスキームで指定されたアプリケーションで配布する選択肢が表示されます。

アプリケーションで配布する選択肢（「○○（アプリ名）で配る」）が表示されるのは、カスタムURLスキームで開いた状態の場合です。
一度ノートを閉じた（ノート一覧画面に戻った/履歴タブで切り替えた）あと、再度開いた場合は表示されません。

※ Noteに取り込む場合は、会議の開催に関する確認・設定画面は表示されません。

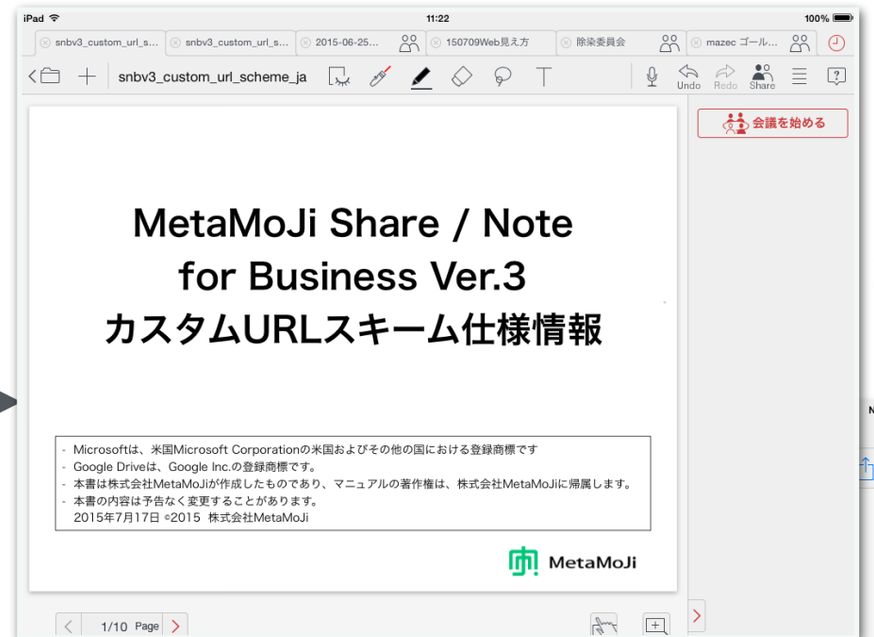
3. ほかの（呼び出し元）アプリケーションから開く

例）MetaMoJi Share / Note形式のファイルを開く場合

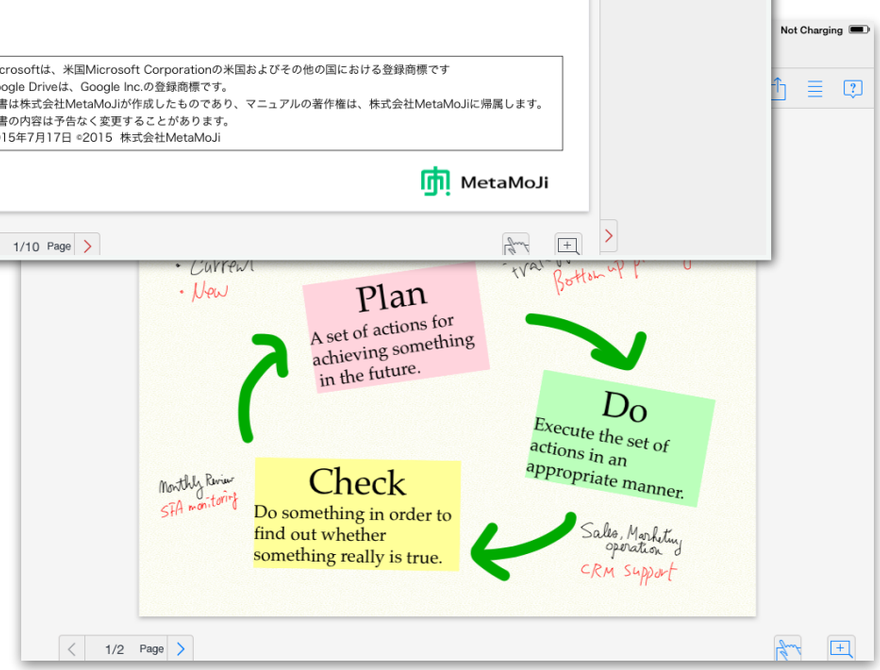
ほかの（呼び出し元）
アプリケーション

MetaMoJi Share 形式
MetaMoJi Note 形式

カスタムURLスキーム呼び出し



Share



Note

※ MetaMoJi Share形式（.btshare）のファイルは、Noteでは読み込めません。

4. 呼び出し元アプリケーションに送る

連携オプションに
"sendbackformat" が指定されている場合に表示されます

ほかの（呼び出し元）
アプリケーション

"sendbackformat" オプションで
指定した形式で送られます
（PDF / MetaMoJi Note形式）

送るメニューに「○○（アプリ名）に送る」が表示されるのは、カスタムURLスキームで "sendbackformat" オプションを付けて開いた状態の場合です。
一度ノートを閉じた（ノート一覧画面に戻った/履歴タブで切り替えた）あと、再度開いた場合は表示されません。

カスタムURLスキームによる
アプリケーション連携イメージ

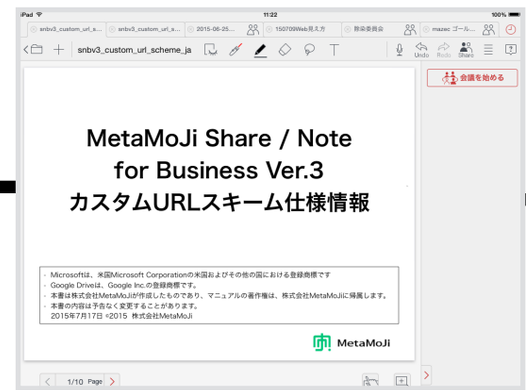
MetaMoJi Share / Note for Business

例）Shareの場合

PDF以外

ほかの
（呼び出し元）
アプリ
ケーション

PDF



送るメニュー

「送る」アイコンから

シェアノートを配る/送るメニューに「○○（アプリ名）で配る」/「○○（アプリ名）に送る」が表示されるのは、カスタムURLスキームで開いた状態の場合です。
一度ノートを閉じた（ノート一覧画面に戻った/履歴タブで切り替えた）あと、再度開いた場合は表示されません。

カスタムURLスキームの仕様

2つの連携方法から選択できます。

- 方法1　クリップボードを使った連携

  アプリ間で引き渡すデータや連携オプションをクリップボードに格納してから、カスタムURLを使ってアプリを呼び出す連携方法です。
  ※クライアントバージョン3.9.5以上で利用可能です。

- 方法2　クエリパラメータを使った連携

  アプリ間で引き渡すデータや連携オプションをカスタムURLのクエリパラメータに付加して、アプリを呼び出す連携方法です。

方法1

クリップボードを使った連携

1. 連携の流れ

ほかの(呼び出し元)アプリケーション

①データと連携オプションをクリップボードに格納する。

②MetaMoJi Share / NoteのカスタムURLをオープンする。

MetaMoJi Share / Note for Business

③起動 → クリップボードに格納されたものを取り出してインポートする。

○○(アプリ名)に送る / ○○(アプリ名)で配る

④データと連携オプション（ファイル名）をクリップボードに格納する。

⑤呼び出し元アプリのカスタムURLをオープンする。

⑥起動 → クリップボードに格納されたものを取り出す。

方法1

クリップボードを使った連携

2. クリップボードに積む項目

クリップボードの1つ目のアイテムとして、次の2つの表現形式を格納します。

表現形式（UTI）	型	説明
com.metamoji.noteanytime.importpb.data	Data	PDFやMetaMoJi Note形式(.atdoc)などインポートするデータ
com.metamoji.noteanytime.importpb.options	Dictionary	ファイル名や呼び出し元アプリ名などの連携オプションを格納した辞書 ※ 3.連携オプションをご覧ください。

方法1

クリップボードを使った連携

3. 連携オプション

キー	型	説明
name	String	ファイル名を想定 ※ ここで指定された拡張子によりデータの扱いが決定されます。
srcname	String	呼び出し元のアプリケーション名 ※ メニューUIなどの文字列として使用されます。
srcurl	String	呼び出し元アプリケーションを呼び出すために使うURL文字列
sendbackformat	String	sendbackformatパラメータを付ける場合は次のいずれかを指定 ・sendbackformat=atdoc ... MetaMoJi Note形式(※ 音声データはメディアサーバーリンクで固定) ・sendbackformat=pdf ... PDF形式 ※ 必須ではありません。指定されてない場合は、送るメニューに「○○(アプリ名)に送る」は表示されません。

方法1

クリップボードを使った連携

4. URLの書式

```
<スキーム>:///importpb
```

項目	説明
スキーム	Shareで開く:shareanytimebiz3 Noteで開く:noteanytimebiz3

方法1

クリップボードを使った連携

5. 連携サンプル（Swift 3）

◆ MetaMoJi Share / Note for Businessの呼び出し

```
func open(_ urlScheme: String, filePath: String, srcName: String, sendBackFormat: String?) {
    guard let data = try? Data(contentsOf: URL(fileURLWithPath: filePath)) else {
        print("Failed to read data!")
        return
    }

    let srcurl = "YOUR-APP:///"

    var options: [String: String] = [:]
    options["name"] = (filePath as NSString).lastPathComponent
    options["srcname"] = srcName
    options["srcurl"] = srcurl
    if let sbformat = sendBackFormat {
        options["sendbackformat"] = sbformat
    }

    let pasteboard = UIPasteboard.general
    guard let optionsPList = try? PropertyListSerialization.data(fromPropertyList: options,
                                                                 format: .binary,
                                                                 options: .allZeros) else {
        print("Failed to make plist!")
        return
    }

    pasteboard.items = [
        [
            "com.metamoji.noteanytime.importpb.data": data,
            "com.metamoji.noteanytime.importpb.options": optionsPList,
        ]
    ]

    let urlString = "\(urlScheme):///importpb"
    if let url = URL(string: urlString) {
        UIApplication.shared.openURL(url)
    }
}
```

方法1

クリップボードを使った連携

5. 連携サンプル（Swift 3）

◆ MetaMoJi Share / Note for Businessからのデータの受け取り

```
func storeFile() -> Bool {
   let pasteboard = UIPasteboard.general

    guard let data = pasteboard.data(forPasteboardType: "com.metamoji.noteanytime.importpb.data") else { return false }
    guard let optionsData = pasteboard.data(forPasteboardType: "com.metamoji.noteanytime.importpb.options") else { return false }

    guard let optionsAny = try? PropertyListSerialization.propertyList(from: optionsData, options: [], format: nil) else { return false }
    guard let options = optionsAny as? Dictionary<String, String> else { return false }

    guard let file = options["name"] else { return false }

    let documentsPath = NSSearchPathForDirectoriesInDomains(.documentDirectory, .userDomainMask, true)[0]
    let fileURL = URL(fileURLWithPath: file, relativeTo: URL(fileURLWithPath: documentsPath, isDirectory: true))

    return (try? data.write(to: fileURL, options: .atomic)) != nil
}
```

方法1

クリップボードを使った連携

5. 連携サンプル（Objective-C）

◆ MetaMoJi Share / Note for Businessの呼び出し

```
- (void)open:(NSString *)urlScheme withFile:(NSString *)filePath srcName:(NSString *)srcName sendBackFormat:(NSString *)sendBackFormat {
    NSString *srcurl = @"YOUR-APP:///";

    NSMutableDictionary *options = [NSMutableDictionary new];
    options[@"name"] = [filePath lastPathComponent];
    options[@"srcname"] = srcName;
    options[@"srcurl"] = srcurl;
    if (sendBackFormat != nil) {
options[@"sendbackformat"] = sendBackFormat;
}



    UIPasteboard *pasteboard = [UIPasteboard generalPasteboard];

    pasteboard.items =
    @[
        @{
            @"com.metamoji.noteanytime.importpb.data": [NSData dataWithContentsOfFile:filePath],
            @"com.metamoji.noteanytime.importpb.options": [NSPropertyListSerialization dataWithPropertyList:options
                                                                                                 format:NSPropertyListBinaryFormat_v1_0
                                                                                               options:0
                                                                                                 error:nil],
        }
    ];

    NSString *urlString = [NSString stringWithFormat:@"%@:///importpb", urlScheme];
    NSURL *url = [[NSURL alloc] initWithString:urlString];
    UIApplication *app = [UIApplication sharedApplication];
    [app openURL:url];
}
```

方法1

クリップボードを使った連携

5. 連携サンプル（Objective-C）

◆ MetaMoJi Share / Note for Businessからのデータの受け取り

```
- (BOOL)storeFile {
    BOOL result = NO;

    UIPasteboard *pasteboard = [UIPasteboard generalPasteboard];
    NSData *data = [pasteboard dataForPasteboardType:@"com.metamoji.noteanytime.importpb.data"];
    NSData *optionsData = [pasteboard dataForPasteboardType:@"com.metamoji.noteanytime.importpb.options"];
    if (data != nil && optionsData != nil) {
        NSDictionary *options;
        options = [NSPropertyListSerialization propertyListWithData:optionsData options:NSPropertyListImmutable format:nil error:nil];
        if (nil != options && [options isKindOfClass:[NSDictionary class]]) {
            NSString *file = options[@"name"];
            if (nil != file) {
                NSString *documentsPath = [NSSearchPathForDirectoriesInDomains(NSDocumentDirectory, NSUserDomainMask, YES) objectAtIndex:0];
                NSString *filePath = [documentsPath stringByAppendingPathComponent:file];
                [data writeToFile:filePath atomically:YES];
                result = YES;
            }
        }
    }

    return result;
}
```

方法2

クエリパラメータを使った連携

1. URLの書式

<スキーム>:///import?data=<データ>&name=<データ名>&srcname=<呼び出し元アプリ名>&srcurl=<呼び出し元アプリに返すためのURL>&sendbackformat=<呼び出し元アプリに返す際のデータ形式>

項目	説明
スキーム	Shareで開く：shareanytimebiz3 Noteで開く：noteanytimebiz3
データ	PDFやMetaMoJi Note形式（.atdoc）などインポートするデータの内容をBase64URL Encordingでエンコードして指定
データ名	ファイル名を想定 UTF-8文字列をPercent-Encodingでエンコードして指定 ※ ここで指定された拡張子によりデータの扱いが決定されます。
呼び出し元アプリ名	呼び出し元のアプリケーション名 UTF-8文字列をPercent-Encodingでエンコードして指定 ※ メニューUIなどの文字列として使用されます。
呼び出し元アプリに返すためのURL	呼び出し元アプリケーションを呼び出すために使うURL文字列 UTF-8文字列をPercent-Encodingでエンコードして指定 ※ この文字列には { } で囲んだプレースホルダを埋め込むことができ、その部分を該当するデータで置き換えた状態で呼び出します（プレースホルダについて詳しくは次ページをご覧ください）。
呼び出し元アプリに返す際のデータ形式	sendbackformatパラメータを付ける場合は次のいずれかを指定 ・sendbackformat=atdoc … MetaMoJi Note形式（※ 音声データはメディアサーバーリンクで固定） ・sendbackformat=pdf … PDF形式 ※ 必須ではありません。指定されてない場合は、送るメニューに「○○（アプリ名）に送る」は表示されません。

方法2

クエリパラメータを使った連携

2. 呼び出し元アプリケーションに返すためのURL

この文字列には { } で囲んだプレースホルダを埋め込むことができます。その部分を該当するデータで置き換えた状態で呼び出します。

プレースホルダ

- `{data}`

  呼び出し元アプリケーションに戻すデータ（PDF、MetaMoJi Note形式など）をBase64URL Encodingでエンコードしたものに置き換えます。

  ※ ほかのアプリケーションから呼び出す場合に指定されるパラメータの「データ」に相当します。

- `{name}`

  呼び出し元アプリケーションに戻すデータのファイル名を、UTF-8文字列のPercent-Encodingでエンコードしたものに置き換えます。

  ※ ほかのアプリケーションから呼び出す場合に指定されるパラメータの「データ名」に相当します。

方法2

クエリパラメータを使った連携

3. カスタムURLスキームの例

shareanytimebiz3:///import?data=5LuK5pel44Gv(略)&name=sample%2epdf&srcname=SampleApp&srcurl=sampleapp%3a%2f%2f%2f%3fxxx%3d111%26yyy%3d222%26data%3d%7bdata%7d%26name%3d%7bname%7d&sendbackformat=pdf

項目	値	説明
<スキーム>	shareanytimebiz3	Shareで開く
<データ>	(バイナリ列)	
<データ名>	sample.pdf	データのファイル名
<呼び出し元アプリ名>	SampleApp	SampleAppという名称のアプリから呼ばれた
<呼び出し元アプリに返すためのURL>	sampleapp:///?xxx=111&yyy=222&data={data}&name={name}	呼び出し元アプリにデータを戻す際に指定するカスタムURLスキーム
<呼び出し元アプリに返す際のデータ形式>	pdf	PDF形式で送る