
MapFan API サーバーAPI 技術仕様書 第2.1.0 版

2018 年 1 月 10 日
インクリメント P 株式会社

目次

1. はじめに	7
本書の目的	7
2. MapFan API サーバーAPI リファレンス	8
API 一覧と概要	8
2.1. API の使用に関するフロー	10
2.2. 認証	11
2.2.1. 認証 : auth	11
リクエスト	11
サンプルリクエスト	11
レスポンス	12
サンプルレスポンス (正常時)	12
サンプルレスポンス (エラー時)	12
2.3. 地図画像取得	13
2.3.1. 地図画像取得 API : map	13
リクエスト	13
サンプルリクエスト	15
レスポンス	15
サンプルレスポンス (正常時)	16
サンプルレスポンス (エラー時)	18
2.3.2. サイズ指定地図画像取得 API : mapimage	19
リクエスト	19
サンプルリクエスト	21
レスポンス	22
サンプルレスポンス (正常時)	22
サンプルレスポンス (エラー時)	23
2.4. 検索	24
2.4.1. 住所検索 : addr	24
リクエスト	24
サンプルリクエスト	26
レスポンス	26
サンプルレスポンス (正常時:JSONP 返却)	27
サンプルレスポンス (正常時:XML 返却)	28
サンプルレスポンス (エラー時)	29
2.4.2. 住所階層検索 : addrlist	30

リクエスト	30
サンプルリクエスト	31
レスポンス	32
サンプルレスポンス（正常時:JSONP 返却）	33
サンプルレスポンス（正常時:XML 返却）	34
サンプルレスポンス（エラー時）	35
2.4.3. 住所逆引き検索：addrname	36
リクエスト	36
サンプルリクエスト	37
レスポンス	37
サンプルレスポンス（正常時:JSONP 返却）	37
サンプルレスポンス（正常時:XML 返却）	37
サンプルレスポンス（エラー時）	38
2.4.4. スポット検索：spot	39
リクエスト	39
サンプルリクエスト	41
レスポンス	42
サンプルレスポンス（正常時:JSONP 返却）	43
サンプルレスポンス（正常時:XML 返却）	45
サンプルレスポンス（エラー時）	46
2.4.5. スポット検索（タウンページデータ検索オプション）：spotopt ※ご利用には別途オプション契約が必要です	47
リクエスト	47
サンプルリクエスト	49
レスポンス	50
サンプルレスポンス（正常時:JSONP 返却）	51
サンプルレスポンス（正常時:XML 返却）	53
サンプルレスポンス（エラー時）	54
2.4.6. スポット周辺検索：spotarea	55
リクエスト	55
サンプルリクエスト	57
レスポンス	58
サンプルレスポンス（正常時:JSONP 返却）	59
サンプルレスポンス（正常時:XML 返却）	61
サンプルレスポンス（エラー時）	62

2.4.7. スポット周辺検索（タウンページデータ検索オプション）：spotareaopt ※ご利用には別途オプション契約が必要です	63
リクエスト	63
サンプルリクエスト	65
レスポンス	66
サンプルレスポンス（正常時:JSONP 返却）	67
サンプルレスポンス（正常時:XML 返却）	69
サンプルレスポンス（エラー時）	70
2.4.8. 駅検索：sta	71
リクエスト	71
サンプルリクエスト	73
レスポンス	73
サンプルレスポンス（正常時:JSONP 返却）	76
サンプルレスポンス（正常時:XML 返却）	79
サンプルレスポンス（エラー時）	82
2.4.9. 駅階層検索：stalist	83
リクエスト	83
サンプルリクエスト	84
レスポンス	85
サンプルレスポンス（正常時:JSONP 返却）	86
サンプルレスポンス（正常時:XML 返却）	87
サンプルレスポンス（エラー時）	88
2.4.10. 最寄駅検索：staarea	89
リクエスト	89
サンプルリクエスト	91
レスポンス	91
サンプルレスポンス（正常時:JSONP 返却）	94
サンプルレスポンス（正常時:XML 返却）	98
サンプルレスポンス（エラー時）	101
2.4.11. 郵便番号検索：zip	102
リクエスト	102
サンプルリクエスト	104
レスポンス	104
サンプルレスポンス（正常時:JSONP 返却）	105
サンプルレスポンス（正常時:XML 返却）	106
サンプルレスポンス（エラー時）	106

2.4.12. 多言語スポット周辺検索 : spotareaml※ご利用には別途オプション契約が 必要です	107
リクエスト	107
サンプルリクエスト	108
レスポンス	109
サンプルレスポンス (正常時:JSONP 返却)	110
サンプルレスポンス (正常時:XML 返却)	111
サンプルレスポンス (エラー時)	112
2.5. ルート機能	113
2.5.1. ルート検索 : calcroute	113
リクエスト	113
サンプルリクエスト	120
レスポンス	121
サンプルレスポンス (正常時:JSONP 返却)	128
サンプルレスポンス (正常時:XML 返却)	131
サンプルレスポンス (エラー時)	134
補足	134
2.6. ユーティリティ	135
2.6.1. 緯度経度測地系変換 : transform	135
リクエスト	135
サンプルリクエスト	137
レスポンス	137
サンプルレスポンス (正常時:JSONP 返却)	138
サンプルレスポンス (正常時:XML 返却)	138
サンプルレスポンス (エラー時)	138
付則	139
サーバーホスト一覧	139
エラーコード一覧	140
認証 API	140
地図 API	141
ルート API	142
検索 API	144
ユーティリティ API	147
地図画像取得	148
アイコン種別一覧	148

ルート機能	150
誘導コード一覧	150
経路種別一覧	151
道路種別一覧	152
規制車種一覧	153
施設情報	154
歩行者リンク付加情報	156
システムの推奨構成	158
PV の定義	159
改版履歴	160

1. はじめに

本書の目的

本書では、インCREMENT P が提供するインターネット地図配信サービス「MapFan API」のサーバーAPI の技術仕様について説明します。

記載されている内容・仕様・デザインなどは、予告無く変更される可能性がありますのでご注意ください。

2. MapFan API サーバーAPI リファレンス

MapFan API サービスでは HTTP(S) GET パラメータによる API を提供しています。

ルート検索結果は、あくまでも参考情報です。

自動車専用道路などといった、自転車/徒歩で通行できない道路をルート案内する可能性がございますので、ご利用の際は必ず実際の道路状況や交通規制に従ってご利用ください。

道路交通情報データは、(公財)日本道路交通情報センター(JARTIC)及び(一財)道路交通情報通信システムセンター(VICS センター)から提供しているものです。

また、情報提供には、VICS センターの技術が用いられています。

API 一覧と概要

分類	API 名	概要
認証	認証	各 API を利用するための認証を行います。
地図	地図画像取得	WMTS 形式の地図画像を提供します。
	サイズ指定地図画像取得	指定された緯度経度、サイズ、縮尺の地図画像を提供します。
検索	住所検索	住所検索機能を提供します。
	住所階層検索	住所階層検索機能を提供します。
	住所逆引き検索	緯度経度から住所情報を取得する住所逆引き検索機能を提供します。
	スポット検索	スポット検索機能を提供します。
	スポット検索(タウンページデータ検索オプション)	タウンページデータ(NTT タウンページ株式会社提供)を検索対象に加えたスポット検索機能を提供します。 本機能のご利用には、別途オプション契約が必要となります。
	スポット周辺検索	指定された緯度経度を中心とした周辺スポットの検索機能を提供します。
	スポット周辺検索(タウンページデータ検索オプション)	タウンページデータ(NTT タウンページ株式会社提供)を検索対象に加えた周辺スポットの検索機能を提供します。 本機能のご利用には、別途オプション契約が必要となります。
	駅検索	駅検索機能を提供します。
	駅階層検索	駅階層検索機能を提供します。

	最寄駅検索	指定された緯度経度を中心とした周辺駅の検索機能を提供します。
	郵便番号検索	郵便番号による住所検索機能を提供します。
	多言語スポット周辺検索	検索結果に外国語を加えた周辺スポットの検索機能を提供します。 本機能のご利用には、別途オプション契約が必要となります。
ルート	ルート検索	ルート検索機能を提供します。
	渋滞考慮ルート検索	VICS 渋滞情報を考慮したルート検索機能を提供します。 本機能のご利用には、別途オプション契約が必要となります。
VICS	VICS 渋滞画像取得	VICS 渋滞情報の渋滞線、規制線を描画したイメージ画像を提供します。 本機能のご利用には、別途オプション契約が必要となります。
	VICS 規制情報取得	VICS 渋滞情報の規制情報、駐車場情報を提供します。 本機能のご利用には、別途オプション契約が必要となります。
ユーティリティ	緯度経度測地系変換	緯度経度情報の測地系および表記形式を変換する機能を提供します。

2.1. API の使用に関するフロー

MapFan API サービスを使用する場合、最初に認証を行う必要があります。

認証を行い、正常に完了した場合は戻り値として各 API を利用するためのアクセスキーを得ることができます。

2.2. 認証

2.2.1. 認証 : auth

各種 API を利用するためのアクセスキーを提供します。

認証 API を利用するには弊社より提供します「認証 ID」が必要になります。

アクセスキーには一定の有効期限が存在するため、長時間の操作を行う場合は定期的に認証を行い、アクセスキーを更新し続ける必要があります。

リクエスト

テスト利用 URL	https://api-auth-pre.mapfan.com/v1/auth?appid={appid}&date={date}&callback={callback}		
本番利用 URL	https://api-auth.mapfan.com/v1/auth?appid={appid}&date={date}&callback={callback}		
パラメータ	型	必須/任意	説明
appid	string	必須	認証 ID 弊社より提供された認証 ID を記述
date	number	必須	利用時の現在時刻 yyyyMMddhhmmss で表記します。 例: 2014 年 11 月 1 日 10 時 20 分 30 秒の場合、「20141101102030」と入力します。
callback	string	任意	JSONP で利用する際のコールバック用関数名 ※callback のパラメータが無い場合は「JSON」で返却

サンプルリクエスト

```
https://api-auth.mapfan.com/v1/auth?appid=xxxxx&date=20141101102030&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	認証 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
key	アクセスキー	各 API サービスに使用するためのアクセスキー

サンプルレスポンス（正常時）

```
jsonp_callback_function (
  {“status”:“success”,
   “key”:“xxxxx”}
)
```

サンプルレスポンス（エラー時）

- ・認証 ID が未設定の場合

```
jsonp_callback_function (
  {“status”:“[E03001] auth parameter error (appid is required)”}
)
```

エラーコード一覧は付則を参照

2.3. 地図画像取得

2.3.1. 地図画像取得 API : map

WMTS 形式の地図画像を提供します。

リクエスト

テスト利用 URL	http(s)://api-map-pre.mapfan.com/v1/map?key={key}&tilematrix={tilematrix}&tilerow={tilerow}&tilecol={tilecol}&format={format}&target={target}&mapstyle={mapstyle}&logo={logo}&lang={lang}&landmarkicon={landmarkicon}&resolution={resolution}&rotationangle={rotationangle}		
本番利用 URL	http(s)://api-map.mapfan.com/v1/map?key={key}&tilematrix={tilematrix}&tilerow={tilerow}&tilecol={tilecol}&format={format}&target={target}&mapstyle={mapstyle}&logo={logo}&lang={lang}&landmarkicon={landmarkicon}&resolution={resolution}&rotationangle={rotationangle}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
tilematrix	string	必須	タイルマトリックス “EPSG コード:スケール値”を記入します。 EPSG コードは EPSG:3857、または EPSG:900913 のみ有効 スケール値は 6-21
tilerow	int	必須	WMTS タイル番号(緯度方向) 地図取得位置の緯度を WMTS タイル番号に変換した値
tilecol	int	必須	WMTS タイル番号(経度方向) 地図取得位置の経度を WMTS タイル番号に変換した値
format	string	任意	出力する画像形式 png の場合は image/png(デフォルト), jpeg の場合は image/jpeg

パラメータ	型	必須/任意	説明
target	int	任意	地図の表示内容 背景のみを表示する場合は、1 注記のみを表示する場合は、2 背景 + 注記を表示する場合は、3(デフォルト)
mapstyle	string	任意	地図デザイン指定用パラメータ ※1
logo	string	任意	ブランドロゴの表示/非表示指定パラメータ ※1
lang	string	任意	注記言語指定パラメータ 本機能のご利用には、別途オプション契約が必要となります。 ※1
landmarkicon	string	任意	ランドマークアイコンの表示/非表示指定パラメータ ※1
resolution	int	任意	解像度パラメータ 標準の解像度の場合は 1 (デフォルト) 標準の 2 倍の解像度の場合は 2 標準の 4 倍の解像度の場合は 4 解像度に 2 を指定した場合、標準の 2 倍のサイズの画像が返され、4 を指定した場合、標準の 4 倍のサイズの画像が返されます。
rotationangle	int	任意	角度パラメータ 注記文字列を指定した角度に傾けた画像が返されます。(デフォルト:0(北上)) 範囲:0-359(整数のみ)

※1 設定方法の詳細は別紙”地図デザイン設定方法と表示イメージ”を参照

サンプルリクエスト

```
http://api-map.mapfan.com/v1/map?key=xxxxx
&tilematrix=EPSG%3A900913%3A16&tilerow=25804&tilecol=58199&format=image%2Fpng&maps
tyle=std_pc&logo=off&lang=ja&landmarkicon=off&resolution=1&rotationangle=0
```

レスポンス

tilematrix, tilerow, tilecol で指定した位置の地図画像を返します。

地図画像は、それぞれ以下の形式で提供します。

サイズ	形式
標準の解像度: 256x256 ピクセル	png または jpeg
標準の 2 倍の解像度: 512x512 ピクセル	
標準の 4 倍の解像度: 1024x1024 ピクセル	

サンプルレスポンス（正常時）

地図画像の出力例(新宿駅_西口周辺)

・mapstyle=std_pc を指定した場合



・mapstyle=std_sp を指定した場合



•mapstyle=gray_pc を指定した場合



•resolution=2 を指定した場合



サンプルレスポンス（エラー時）

- ・WMTS タイルマトリックスが未設定の場合

[E03001] map parameter error (tilematrix is required)

エラーコード一覧は付則を参照

2.3.2. サイズ指定地図画像取得 API : mapimage

指定された緯度経度、サイズ、縮尺の地図画像を提供します。

リクエスト

テスト利用 URL	http(s)://api-map-pre.mapfan.com/v1/mapimage?key={key}&size={size}&lonlat={lonlat}&scale={scale}&format={format}&target={target}&mapstyle={mapstyle}&logo={logo}&lang={lang}&landmarkicon={landmarkicon}&resolution={resolution}&rotationangle={rotationangle}&icon={icon}&scaler={scaler}¢ericon={centericon}		
本番利用 URL	http(s)://api-map.mapfan.com/v1/mapimage? key={key}&size={size}&lonlat={lonlat}&scale={scale}&format={format}&target={target}&mapstyle={mapstyle}&logo={logo}&lang={lang}&landmarkicon={landmarkicon}&resolution={resolution}&rotationangle={rotationangle}&icon={icon}&scaler={scaler}¢ericon={centericon}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
size	string	必須	画像サイズ 記述形式 [画像幅(ピクセル値)]+"",[画像高さ(ピクセル値)] 記述例: 320,240 (幅 320px, 高さ 240px の場合) 範囲: 最小値: 1px, 1px 最大値: 2048px, 2048px
lonlat	string	必須	中心緯度経度 記述形式: [経度の値]+",[緯度の値] 記述例: 139.767231,35.681196" 範囲: 最小: 経度 120, 緯度 20 最大: 経度 155, 緯度 46
scale	int	必須	地図縮尺 範囲: 6-21

パラメータ	型	必須/任意	説明
format	string	任意	出力する画像形式 png の場合は image/png(デフォルト), jpeg の場合は image/jpeg
target	int	任意	地図の表示内容 背景のみを表示する場合は、1 注記のみを表示する場合は、2 背景 + 注記を表示する場合は、3(デフォルト)
mapstyle	string	任意	地図デザイン指定用パラメータ ※1
logo	string	任意	ブランドロゴの表示/非表示指定パラメータ。 ※1
lang	string	任意	注記言語指定パラメータ 本機能のご利用には、別途オプション契約が必要となります。 ※1
landmarkicon	string	任意	ランドマークアイコンの表示/非表示指定パラメータ ※1
resolution	int	任意	解像度パラメータ 標準の解像度の場合は 1 (デフォルト) 標準の 2 倍の解像度の場合は 2 解像度に 2 を指定した場合、size で指定した画像サイズの 2 倍の画像サイズで返されます。
rotationangle	int	任意	角度パラメータ 地図を指定した角度に傾けた画像が返されます。(デフォルト:0(北上)) 範囲:0-359(整数のみ)

パラメータ	型	必須/任意	説明
icon	string	任意	<p>アイコン設定</p> <p>記述形式:</p> <p>[アイコン定義#1] + " " + [アイコン定義#2] + " " ... [アイコン定義#n]</p> <p>アイコン定義記述形式:</p> <p>[アイコン種別コード] + "," + [経度の値] + "," + [緯度の値]</p> <p>記述例:</p> <p>1,139.7672, 35.6811 102,139.7671,35.6799</p> <p>範囲:</p> <p>最小:経度 120, 緯度 20</p> <p>最大:経度 155, 緯度 46</p> <p>n はアイコン定義の件数で、最大 50 件まで指定可能です。但し scaler・centericon が on の場合、表示されるスケーラ・中心点アイコンもそれぞれアイコン定義として件数にカウントされます。</p> <p>アイコン種別コードの設定値については、付則. アイコン種別一覧を参照。</p>
scaler	string	任意	<p>スケーラ表示</p> <p>非表示の場合は off (デフォルト)</p> <p>表示の場合は on</p>
centericon	string	任意	<p>中心点アイコン表示</p> <p>非表示の場合は off (デフォルト)</p> <p>表示の場合は on</p>

※1 設定方法の詳細は別紙”地図デザイン設定方法と表示イメージ”を参照

サンプルリクエスト

```
http://api-map.mapfan.com/v1/mapimage?key=xxxxx
&size=320,240&lonlat=139.767231,35.681196&scale=15&format=image%2Fpng&mapstyle=std_pc&logo=on&lang=ja&landmarkicon=on&resolution=1&rotationangle=45&icon=1,139.76505757,35.68313212|105,139.7715374,35.67966047|3,139.76247749,35.68430013|4,139.76264448,35.67964567&scaler=on&centericon=on
```

レスポンス

size, lonlat, scale で指定した画像サイズ、位置、スケールの地図画像を返します。
resolution で解像度に 2 を指定した場合、size で指定した画像サイズの 2 倍の画像サイズで返されます。

サンプルレスポンス（正常時）

サイズ指定地図画像の出力例（東京駅周辺）

・幅 320px、高さ 240px のサイズを指定した場合



・アイコン設定を指定、及びスケール、中心点アイコンを表示設定とした場合



- ・幅 320px、高さ 240px のサイズ、標準の倍の解像度を指定した場合



サンプルレスポンス（エラー時）

- ・中心緯度経度(lonlat)が未設定の場合

[E03001] mapimage parameter error (lonlat is required)

エラーコード一覧は付則を参照

2.4.検索

2.4.1. 住所検索 : addr

住所検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/addr?key={key}&addr={addr}&gov={gov}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/addr?key={key}&addr={addr}&gov={gov}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
addr	string	必須	検索対象の住所文字列(UTF-8)
gov	int	任意	政令指定都市表示有無設定 表示する場合は 0 (デフォルト) 表示しない場合は 1
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は 500 件となります。
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。

パラメータ	型	必須/任意	説明
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-srch.mapfan.com/v1/addr?key=xxxxx&addr=東京
&gov=0&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	住所検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	住所	ヒットした住所文字列
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度
prefcd	都道府県コード	都道府県コード
citycd	市区町村コード	市区町村コード
tyocd	条町丁目コード	条町丁目コード
bancd	番地コード	番地コード
goucd	号コード	号コード
level	精度レベル	0:都道府県レベル 1:市区町村レベル 2:条町丁目レベル 3:番地レベル 4:号レベル

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(  
  {  
    "status": "success",  
    "hit": 1,  
    "start_pos": 1,  
    "end_pos": 1,  
    "results": [  
      {  
        "name": "¥u6771¥u4eac¥u90fd",  
        "lon": 139.69171646012,  
        "lat": 35.689456581983,  
        "prefcd": "13",  
        "citycd": "",  
        "tyocd": "",  
        "bancd": "",  
        "goucd": ""  
      },  
    ],  
    "level": 0  
  })
```

サンプルレスポンス（正常時:XML 返却）

```
<addrResponse>
  <status>success</status>
  <hit>1</hit>
  <start_pos>1</start_pos>
  <end_pos>1</end_pos>
  <result>
    <name>東京都</name>
    <lon>139.69171646012</lon>
    <lat>35.689456581983</lat>
    <prefcd>13</prefcd>
    <citycd/>
    <tyocd/>
    <bancd/>
    <goucd/>
  </result>
  <level>0</level>
</addrResponse>
```

サンプルレスポンス（エラー時）

- ・ 検索対象の住所文字列(addr)が未設定の場合

```
jsonp_callback_function (  
  {"status": "[E03001] addr parameter error (addr is required)"}  
)
```

エラーコード一覧は付則を参照

2.4.2. 住所階層検索 : addrlist

住所階層検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/addrlist?key={key}&level={level}&prefcd={prefcd}&citycd={citycd}&tyocd={tyocd}&bancd={bancd}&gov={gov}&fmt={fmt}& num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/addrlist?key={key}&level={level}&prefcd={prefcd}&citycd={citycd}&tyocd={tyocd}&bancd={bancd}&gov={gov}&fmt={fmt}& num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
level	int	必須	検索階層 都道府県一覧の場合は 0 市区町村一覧の場合は 1 条町丁目一覧の場合は 2 番地一覧の場合は 3 号一覧の場合は 4
prefcd	string	必須(*1)/ 任意	都道府県コード(数字 2 桁) 01~47 (*1) level に 1,2,3,4 を指定した場合は必須
citycd	string	必須(*2)/ 任意	市区町村コード(数字 3 桁) (*2) level に 2,3,4 を指定した場合は必須
tyocd	string	必須(*3)/ 任意	条町丁目コード(英数 6 桁) (*3) level に 3,4 を指定した場合は必須
bancd	string	必須(*4)/ 任意	番地コード(英数 5 桁) (*4) level に 4 を指定した場合は必須
gov	int	任意	政令指定都市表示有無設定 表示する場合は 0 (デフォルト) 表示しない場合は 1

パラメータ	型	必須/任意	説明
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は、全件となります。
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-  
srch.mapfan.com/v1/addrlist?key=xxxxx&level=1&prefcd=13&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	住所階層検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
level	精度レベル	ヒットした住所階層レベル
name	住所	住所文字列
yomi	読み	住所文字列読み
prefcd	都道府県コード	都道府県コード
citycd	市区町村コード	市区町村コード
tyocd	条町丁目コード	条町丁目コード
bancd	番地コード	番地コード
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(  
{  
  "status": "success",  
  "hit": 62,  
  "start_pos": 1,  
  "end_pos": 62,  
  "results": [  
    {  
      "level": 1,  
      "name": "¥u9752¥u30f6¥u5cf6¥u6751",  
      "yomi": "¥u30a2¥u30aa¥u30ac¥u30b7¥u30de¥u30e0¥u30e9",  
      "prefcd": "13",  
      "citycd": "402",  
      "tyocd": "000000",  
      "bancd": "00000",  
      "lon": 139.76340263602,  
      "lat": 32.466531399186  
    },  
    {  
      "level": 1,  
      "name": "¥u662d¥u5cf6¥u5e02",  
      "yomi": "¥u30a2¥u30ad¥u30b7¥u30de¥u30b7",  
      "prefcd": "13",  
      "citycd": "207",  
      "tyocd": "000000",  
      "bancd": "00000",  
      "lon": 139.35363373487,  
      "lat": 35.705696157369  
    },  
    ..  
  ]  
})
```

サンプルレスポンス（正常時:XML 返却）

```
<addrlistResponse>
  <status>success</status>
  <hit>62</hit>
  <start_pos>1</start_pos>
  <end_pos>62</end_pos>
  <result>
    <level>1</level>
    <name>青ヶ島村</name>
    <yomi>アオガシマムラ</yomi>
    <prefcd>13</prefcd>
    <citycd>402</citycd>
    <tyocd>000000</tyocd>
    <bancd>00000</bancd>
    <lon>139.76340263602</lon>
    <lat>32.466531399186</lat>
  </result>
  <result>
    <level>1</level>
    <name>昭島市</name>
    <yomi>アキシマシ</yomi>
    <prefcd>13</prefcd>
    <citycd>207</citycd>
    <tyocd>000000</tyocd>
    <bancd>00000</bancd>
    <lon>139.35363373487</lon>
    <lat>35.705696157369</lat>
  </result>
  ..
</addrlistResponse>
```

サンプルレスポンス（エラー時）

- ・ 検索階層(level)が未設定の場合

```
jsonp_callback_function (  
  {"status": "[E03001] addrlist parameter error (level is required)"}  
)
```

エラーコード一覧は付則を参照

2.4.3. 住所逆引き検索 : addrname

緯度経度から住所情報を取得する住所逆引き検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/addrname?key={key}&fmt={fmt}&callback={callback}&lonlat={lonlat}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/addrname?key={key}&fmt={fmt}&callback={callback}&lonlat={lonlat}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。
lonlat	string	必須	緯度経度 測地系: 世界測地系(JGD2000) 値の表現: degree 形式 記述形式: [経度の値]+","+[緯度の値] 記述例: "139.767231,35.681196"

サンプルリクエスト

```
http://api-
srch.mapfan.com/v1/addname?key=xxxxx&callback=jsonp_callback_function&lonlat=139.76
7231,35.681196
```

レスポンス

パラメータ	値	説明
status	ステータス	住所逆引き検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメ ッセージ”を返します。
address	住所文字列	指定された緯度経度付近の住所の文字列 都道府県、市区町村、町丁目が記述されます。 文字コードは JSON/JSONP のときは UTF-16、 XML のときは UTF-8。
zipcode	郵便番号	郵便番号 住所文字列の住所に該当する郵便番号。

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(
  { "status": "success", "address": "¥u6771¥u4eac¥u90fd ¥u5343¥u4ee3¥u7530¥u533a
¥u4e38¥u306e¥u5185¥uff11", "zipcode": "1000005" }
)
```

サンプルレスポンス（正常時:XML 返却）

```
<addnameResponse>
  <status>success</status>
  <address>東京都 千代田区 丸の内1</address>
  <zipcode>1000005</zipcode>
</addnameResponse>
```

サンプルレスポンス（エラー時）

- ・緯度経度(lonlat)が未設定の場合

```
jsonp_callback_function (  
  {"status":"[E03001] addrname parameter error (lonlat is required)"}  
)
```

エラーコード一覧は付則を参照

2.4.4. スポット検索 : spot

スポット検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/spot?key={key}&name={name}&gnrcd={gnrcd}&prefcd={prefcd}&citycd={citycd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/spot?key={key}&name={name}&gnrcd={gnrcd}&prefcd={prefcd}&citycd={citycd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
name	string	必須	検索対象の文字列(UTF-8) スペース区切りで複数のキーワードを指定可能です。 複数指定した場合は、AND 検索となります。
gnrcd	string	任意	ジャンルコード 特定ジャンルのスポットのみを対象として検索を行う場合に、検索対象のジャンルコードを指定します。 カンマ区切りで複数のジャンルコードを指定可能です。 複数指定した場合は OR 検索となります。
prefcd	string	任意	都道府県コード 特定の都道府県のスポットのみを対象として検索を行う場合に、検索対象の都道府県コードを指定します。 ※市区町村コードを指定した場合は必須
citycd	string	任意	市区町村コード 特定の市区町村のスポットのみを対象として検索を行う場合に、検索対象の市区町村コードを指定します。

パラメータ	型	必須/任意	説明
field	string	任意	<p>検索フィールド</p> <p>スポットの特定の属性のみを対象としてキーワード検索を行う場合に、対象とする属性(検索フィールド)を指定します。</p> <p>カンマ区切りで複数の検索フィールドを指定可能です。</p> <p>複数指定した場合は OR 検索となります。</p> <p>指定可能な検索フィールド</p> <ul style="list-style-type: none"> •name(スポット名) •yomi(スポット名読み) •gnr_name(ジャンル名) •addr(住所) •tel(電話番号) •comment(コメント) •access(アクセス情報※) <p>※スポットの最寄りの鉄道路線・駅、インターチェンジ</p>
fmt	string	任意	<p>出力フォーマット</p> <p>JSON 出力の場合は json (デフォルト)</p> <p>XML 出力の場合は xml</p> <p>なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。</p>
num	int	任意	<p>取得件数</p> <p>検索結果の返却件数を指定します。</p> <p>最大で 500 件まで指定可能。</p> <p>指定がない場合は、50 件となります。</p>
pos	int	任意	<p>取得開始位置</p> <p>検索結果の返却開始位置を指定します。</p> <p>指定がない場合は、1 件目から取得件数分の結果を返却します。</p>

パラメータ	型	必須/任意	説明
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-srch.mapfan.com/v1/spot?key=xxxxx&name=東京  
&gnrcd=5&prefcd=13&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	スポット検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	スポット名	ヒットしたスポットの名称
yomi	スポット名読み	ヒットしたスポットの名称読み
gnr_info	ジャンル情報	ヒットしたスポットのジャンル情報
gnr_stepno	ジャンル階層番号	ジャンル情報の階層番号 1:第一階層ジャンル 2:第二階層ジャンル
gnr_name	ジャンル名	ヒットしたスポットのジャンル名
addr	住所	ヒットしたスポットの住所
tel	電話番号	ヒットしたスポットの電話番号
comment	コメント	ヒットしたスポットのスポット情報(コメント形式)
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(
{
  "status":"success",
  "hit":10214,
  "start_pos":1,
  "end_pos":50,
  "results":[
    {
      "name":"¥u6771¥u4eac¥u30bd¥u30e9¥u30de¥u30c1",
      "yomi":"¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30bd¥u30e9¥u30de¥u30c1",
      "gnr_info":[
        {
          "gnr_stepno":1,
          "gnr_name":"¥u8cb7¥u3046"
        },
        {
          "gnr_stepno":2,
          "gnr_name":"¥u30b7¥u30e7¥u30c3¥u30d4¥u30f3¥u30b0¥u30e2¥u30fc¥u30eb"
        }
      ],
      "addr":"¥u6771¥u4eac¥u90fd¥u58a8¥u7530¥u533a¥u62bc¥u4e0a¥uff11¥uff0d¥uff11¥uff0d¥uff12",
      "tel":"0570-55-0102",
      "comment":"¥u6771¥u4eac¥u306e¥u65b0¥u540d¥u6240¥uff62¥u6771¥u4eac¥uff7d¥uff76¥uff72¥uff82¥uff98¥uff70¥uff63¥u306e¥u8db3¥u5143¥u306b¥u96c6¥u307e¥u308b¥u5546¥u696d¥u65bd¥u8a2d¥uff61¥u7d04310¥u306e¥u5e97¥u8217¥u304c¥u96c6¥u307e¥u308a¥uff64¥uff7d¥uff76¥uff72¥uff82¥uff98¥uff70¥u306e¥u89b3¥u5149¥u3068¥u3042¥u308f¥u305b¥u3066¥uff7c¥uff6e¥uff6f¥uff8b¥uff9f¥uff9d¥uff78¥uff9e¥u3092¥u697d¥u3057¥u3082¥u3046¥uff61¥n¥u3010¥u55b6¥u3011 10:00-21:00(¥u5e97¥u8217¥u306b¥u3088¥u308a¥u7570¥u306a¥u308b)¥n¥u3010¥u4f11¥u3011 ¥u4e0d¥u5b9a¥u4f11¥n¥u3010¥uff30¥u3011 ¥u7d041000¥u53f0",
      "lon":139.80951861981,
      "lat":35.709960068465
    }
  ],
}
```

```
{
  "name": "¥u5927¥u4e38¥u6771¥u4eac¥u5e97",
  "yomi": "¥u30c0¥u30a4¥u30de¥u30eb¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30c6¥u
30f3",
  "gnr_info": [
    {
      "gnr_stepno": 1,
      "gnr_name": "¥u8cb7¥u3046"
    },
    {
      "gnr_stepno": 2,
      "gnr_name": "¥u30c7¥u30d1¥u30fc¥u30c8"
    }
  ],
  "addr": "¥u6771¥u4eac¥u90fd¥u5343¥u4ee3¥u7530¥u533a¥u4e38¥u306e¥u5185¥uf
f11¥uff0d¥uff19¥uff0d¥uff11",
  "tel": "03-3212-8011",
  "comment": null,
  "lon": 139.76901514648,
  "lat": 35.681681377209
},
..
)
```

サンプルレスポンス（正常時:XML 返却）

```
<spotResponse>
  <status>success</status>
  <hit>10214</hit>
  <start_pos>1</start_pos>
  <end_pos>50</end_pos>
  <result>
    <name>東京ソラマチ</name>
    <yomi>トウキョウソラマチ</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>買う</gnr_name>
    </gnr_info>
    <gnr_info>
      <gnr_stepno>2</gnr_stepno>
      <gnr_name>ショッピングモール</gnr_name>
    </gnr_info>
    <addr>東京都墨田区押上1-1-2</addr>
    <tel>0570-55-0102</tel>
    <comment>東京の新名所「東京スカイツリー」の足元に集まる商業施設。約 310 の店舗が
      集まり、スカイツリーの観光とあわせてショッピングを楽しもう。
    </comment>
    【営】 10:00-21:00(店舗により異なる)
    【休】 不定休
    【P】 約 1000 台</comment>
    <lon>139.80951861981</lon>
    <lat>35.709960068465</lat>
  </result>
  <result>
    <name>大丸東京店</name>
    <yomi>ダイマルトウキョウテン</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>買う</gnr_name>
    </gnr_info>
    <gnr_info>
```

```
<gnr_stepno>2</gnr_stepno>
<gnr_name>デパート</gnr_name>
</gnr_info>
<addr>東京都千代田区丸の内1-9-1</addr>
<tel>03-3212-8011</tel>
<comment/>
<lon>139.76901514648</lon>
<lat>35.681681377209</lat>
</result>
..
</spotResponse>
```

サンプルレスポンス（エラー時）

- ・ 検索対象の文字列(name)が未設定の場合

```
jsonp_callback_function (
  {"status": "[E03001] spot parameter error (name is required)"}
)
```

エラーコード一覧は付則を参照

2.4.5. スポット検索（タウンページデータ検索オプション）：spotopt ※ご利用には別途オプション契約が必要です

タウンページデータ(NTT タウンページ株式会社提供)を検索対象に加えたスポット検索機能を提供します。

本機能のご利用には、別途オプション契約が必要となります。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/spotopt?key={key}&name={name}&gnrcd={gnrcd}&prefcd={prefcd}&citycd={citycd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/spotopt?key={key}&name={name}&gnrcd={gnrcd}&prefcd={prefcd}&citycd={citycd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
name	string	必須	検索対象の文字列(UTF-8) スペース区切りで複数のキーワードを指定可能です。 複数指定した場合は、AND 検索となります。
gnrcd	string	任意	ジャンルコード 特定ジャンルのスポットのみを対象として検索を行う場合に、検索対象のジャンルコードを指定します。 カンマ区切りで複数のジャンルコードを指定可能です。 複数指定した場合は OR 検索となります。
prefcd	string	任意	都道府県コード 特定の都道府県のスポットのみを対象として検索を行う場合に、検索対象の都道府県コードを指定します。 ※市区町村コードを指定した場合は必須

パラメータ	型	必須/任意	説明
citycd	string	任意	市区町村コード 特定の市区町村のスポットのみを対象として検索を行う場合に、検索対象の市区町村コードを指定します。
field	string	任意	検索フィールド スポットの特定の属性のみを対象としてキーワード検索を行う場合に、対象とする属性(検索フィールド)を指定します。 カンマ区切りで複数の検索フィールドを指定可能です。 複数指定した場合は OR 検索となります。 指定可能な検索フィールド <ul style="list-style-type: none"> •name(スポット名) •yomi(スポット名読み) •gnr_name(ジャンル名) •addr(住所) •tel(電話番号) •comment(コメント) •access(アクセス情報※) ※スポットの最寄りの鉄道路線・駅、インターチェンジ
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は、50 件となります。

パラメータ	型	必須/任意	説明
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-srch.mapfan.com/v1/spotopt?key=xxxxx&name=東京  
&gnrcd=5&prefcd=13&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	スポット検索(タウンページデータ検索オプション) API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメ ッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	スポット名	ヒットしたスポットの名称
yomi	スポット名読み	ヒットしたスポットの名称読み
gnr_info	ジャンル情報	ヒットしたスポットのジャンル情報
gnr_stepno	ジャンル階層番号	ジャンル情報の階層番号 1:第一階層ジャンル 2:第二階層ジャンル
gnr_name	ジャンル名	ヒットしたスポットのジャンル名
addr	住所	ヒットしたスポットの住所
tel	電話番号	ヒットしたスポットの電話番号
comment	コメント	ヒットしたスポットのスポット情報(コメント形式)
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

```

jsonp_callback_function(
{
  "status":"success",
  "hit":10214,
  "start_pos":1,
  "end_pos":50,
  "results":[
    {
      "name":"¥u6771¥u4eac¥u30bd¥u30e9¥u30de¥u30c1",
      "yomi":"¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30bd¥u30e9¥u30de¥u30c1",
      "gnr_info":[
        {
          "gnr_stepno":1,
          "gnr_name":"¥u8cb7¥u3046"
        },
        {
          "gnr_stepno":2,
          "gnr_name":"¥u30b7¥u30e7¥u30c3¥u30d4¥u30f3¥u30b0¥u30e2¥u30fc¥u30eb"
        }
      ],
      "addr":"¥u6771¥u4eac¥u90fd¥u58a8¥u7530¥u533a¥u62bc¥u4e0a¥uff11¥uff0d¥uff11¥uff0d¥uff12",
      "tel":"0570-55-0102",
      "comment":"¥u6771¥u4eac¥u306e¥u65b0¥u540d¥u6240¥uff62¥u6771¥u4eac¥uff7d¥uff76¥uff72¥uff98¥uff70¥uff63¥u306e¥u8db3¥u5143¥u306b¥u96c6¥u307e¥u308b¥u5546¥u696d¥u65bd¥u8a2d¥uff61¥u7d04310¥u306e¥u5e97¥u8217¥u304c¥u96c6¥u307e¥u308a¥uff64¥uff7d¥uff76¥uff72¥uff98¥uff70¥u306e¥u89b3¥u5149¥u3068¥u3042¥u308f¥u305b¥u3066¥uff7c¥uff6e¥uff6f¥uff8b¥uff9f¥uff9d¥uff78¥uff9e¥u3092¥u697d¥u3057¥u3082¥u3046¥uff61¥n¥u3010¥u55b6¥u3011 10:00-21:00(¥u5e97¥u8217¥u306b¥u3088¥u308a¥u7570¥u306a¥u308b)¥n¥u3010¥u4f11¥u3011 ¥u4e0d¥u5b9a¥u4f11¥n¥u3010¥uff30¥u3011 ¥u7d041000¥u53f0",
      "lon":139.80951861981,
      "lat":35.709960068465
    },
  ],
}

```

```
{
  "name": "¥u5927¥u4e38¥u6771¥u4eac¥u5e97",
  "yomi": "¥u30c0¥u30a4¥u30de¥u30eb¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30c6¥u
30f3",
  "gnr_info": [
    {
      "gnr_stepno": 1,
      "gnr_name": "¥u8cb7¥u3046"
    },
    {
      "gnr_stepno": 2,
      "gnr_name": "¥u30c7¥u30d1¥u30fc¥u30c8"
    }
  ],
  "addr": "¥u6771¥u4eac¥u90fd¥u5343¥u4ee3¥u7530¥u533a¥u4e38¥u306e¥u5185¥uf
f11¥uff0d¥uff19¥uff0d¥uff11",
  "tel": "03-3212-8011",
  "comment": null,
  "lon": 139.76901514648,
  "lat": 35.681681377209
},
..
)
```

サンプルレスポンス（正常時:XML 返却）

```
<spotoptResponse>
  <status>success</status>
  <hit>103201</hit>
  <start_pos>1</start_pos>
  <end_pos>50</end_pos>
  <result>
    <name>東京SASSY</name>
    <yomi>トウキョウサッシー</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>買う</gnr_name>
    </gnr_info>
    <gnr_info>
      <gnr_stepno>2</gnr_stepno>
      <gnr_name>衣料品店</gnr_name>
    </gnr_info>
    <addr>東京都世田谷区上用賀3丁目7-18</addr>
    <tel>03-5797-9926</tel>
    <comment/>
    <lon>139.62905285839</lon>
    <lat>35.632316729617</lat>
  </result>
  <result>
    <name>東京堂</name>
    <yomi>トウキョウドウ</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>買う</gnr_name>
    </gnr_info>
    <gnr_info>
      <gnr_stepno>2</gnr_stepno>
      <gnr_name>リサイクルショップ</gnr_name>
    </gnr_info>
    <addr>東京都世田谷区玉川台2丁目35-11</addr>
```

```
<tel>03-3709-5450</tel>
<comment/>
<lon>139.62856120294</lon>
<lat>35.626323539024</lat>
</result>
..
</spotoptResponse>
```

サンプルレスポンス（エラー時）

- ・ 検索対象の文字列(name)が未設定の場合

```
jsonp_callback_function (
  {"status": "[E03001] spotopt parameter error (name is required)"}
)
```

エラーコード一覧は付則を参照

2.4.6. スポット周辺検索 : spotarea

指定された緯度経度を中心とした周辺スポットの検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/spotarea?key={key}&lonlat={lonlat}&rad={rad}&name={name}&gnrcd={gnrcd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/spotarea?key={key}&lonlat={lonlat}&rad={rad}&name={name}&gnrcd={gnrcd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
lonlat	string	必須	緯度経度 測地系: 世界測地系(JGD2000) 値の表現: degree 形式 記述形式: [経度の値]+", "+[緯度の値] 記述例: "139.767231,35.681196"
rad	int	任意	検索範囲 検索半径をメートル単位で指定します。 最大で 50000 (50km) まで指定可能。 指定がない場合は、1000 (1km) として結果を返却します。
name	string	任意	検索対象の絞り込み文字列(UTF-8) スペース区切りで複数のキーワードを指定可能です。 複数指定した場合は、AND 検索となります。 ※検索フィールドを指定した場合は必須

パラメータ	型	必須/任意	説明
gnrcd	string	任意	<p>ジャンルコード</p> <p>特定ジャンルのスポットのみを対象として検索を行う場合に、検索対象のジャンルコードを指定します。</p> <p>カンマ区切りで複数のジャンルコードを指定可能です。</p> <p>複数指定した場合は OR 検索となります。</p>
field	string	任意	<p>検索フィールド</p> <p>スポットの特定の属性のみを対象としてキーワード検索を行う場合に、対象とする属性(検索フィールド)を指定します。</p> <p>カンマ区切りで複数の検索フィールドを指定可能です。</p> <p>複数指定した場合は OR 検索となります。</p> <p>指定可能な検索フィールド</p> <ul style="list-style-type: none"> •name(スポット名) •yomi(スポット名読み) •gnr_name(ジャンル名) •addr(住所) •tel(電話番号) •comment(コメント) •access(アクセス情報※) <p>※スポットの最寄りの鉄道路線・駅、インターチェンジ</p>
fmt	string	任意	<p>出力フォーマット</p> <p>JSON 出力の場合は json (デフォルト)</p> <p>XML 出力の場合は xml</p> <p>なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。</p>

パラメータ	型	必須/任意	説明
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は、50 件となります。
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-
srch.mapfan.com/v1/spotarea?key=xxxxx&lonlat=139.767231,35.681196&rad=5000&name=
ローソン&gnrcd=15&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	スポット周辺検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	スポット名	ヒットしたスポットの名称
yomi	スポット名読み	ヒットしたスポットの名称読み
gnr_info	ジャンル情報	ヒットしたスポットのジャンル情報
gnr_stepno	ジャンル階層番号	ジャンル情報の階層番号 1:第一階層ジャンル 2:第二階層ジャンル
gnr_name	ジャンル名	ヒットしたスポットのジャンル名
addr	住所	ヒットしたスポットの住所
tel	電話番号	ヒットしたスポットの電話番号
comment	コメント	ヒットしたスポットのスポット情報(コメント形式)
distance	距離	指定した緯度経度からの距離(メートル)
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(
{
  "status": "success",
  "hit": 354,
  "start_pos": 1,
  "end_pos": 50,
  "results": [
    {
      "name": "¥u30ed¥u30fc¥u30bd¥u30f3¥u516b¥u91cd¥u6d32¥u4e2d¥u592e¥u53e3¥u5e97",
      "yomi": "¥u30ed¥u30fc¥u30bd¥u30f3¥u30e4¥u30a8¥u30b9¥u30c1¥u30e5¥u30a6¥u30aa¥u30a6¥u30b0¥u30c1¥u30c6¥u30f3",
      "gnr_info": [
        {
          "gnr_stepno": 1,
          "gnr_name": "¥u8cb7¥u3046"
        },
        {
          "gnr_stepno": 2,
          "gnr_name": "¥u30b3¥u30f3¥u30d3¥u30cb¥u30a8¥u30f3¥u30b9¥u30b9¥u30c8¥u30a2"
        }
      ],
      "addr": "¥u6771¥u4eac¥u90fd¥u4e2d¥u592e¥u533a¥u516b¥u91cd¥u6d32¥uff12¥uff0d¥uff11¥u516b¥u91cd¥u6d32¥u5730¥u4e0b¥u8857¥u4e2d¥uff13",
      "tel": "03-3231-6300",
      "comment": "¥u3010¥u55b6¥u3011 ¥u5168¥u65e5 7:00-22:00¥n¥u3010¥u4f11¥u3011¥u7121¥u4f11¥n¥u3010¥uff30¥u3011 ¥u306a¥u3057¥n¥u3010ATM¥u3011¥u306a¥u3057",
      "distance": 252,
      "lon": 139.76944613873,
      "lat": 35.679819588244
    },
    {

```

```

"name": "\uff2a\u30ed\u30fc\u30bd\u30f3\u6771\u4eac\u4e2d\u592e\u5c40
\u5e97",
"yomi": "\u30b8\u30a7\u30fc\u30d4\u30fc\u30ed\u30fc\u30bd\u30f3\u30c8\u30a
6\u30ad\u30e7\u30a6\u30c1\u30e5\u30a6\u30aa\u30a6\u30ad\u30e7\u30af\u30c
6\u30f3",
"gnr_info": [
  {
    "gnr_stepno": 1,
    "gnr_name": "\u8cb7\u3046"
  },
  {
    "gnr_stepno": 2,
    "gnr_name": "\u30b3\u30f3\u30d3\u30cb\u30a8\u30f3\u30b9\u30b9\u30c8\u3
0a2"
  }
],
"addr": "\u6771\u4eac\u90fd\u5343\u4ee3\u7530\u533a\u4e38\u306e\u5185\u512
2\u5122\u5127\u5122\u5122",
"tel": "03-3287-9001",
"comment": "\u3010\u55b6\u3011 \u5168\u65e5 7:00-23:00\u3010\u4f11\u3011
\u4e0d\u5b9a\u4f11\u3010\u3011 \u306a\u3057\u3010ATM\u3011
\u4e00\u90e8\u63d0\u643a\u91d1\u878d\u6a5f\u95a2",
"distance": 260,
"lon": 139.7648295011,
"lat": 35.679900455225
},
..
)

```

サンプルレスポンス（正常時:XML 返却）

```
<spotareaResponse>
  <status>success</status>
  <hit>354</hit>
  <start_pos>1</start_pos>
  <end_pos>50</end_pos>
  <result>
    <name>ローソン八重洲中央口店</name>
    <yomi>ローソンヤエスチュウオウグチテン</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>買う</gnr_name>
    </gnr_info>
    <gnr_info>
      <gnr_stepno>2</gnr_stepno><gnr_name>コンビニエンスストア
      </gnr_name></gnr_info><addr>東京都中央区八重洲2-1八重洲地下街中3
      </addr><tel>03-3231-6300</tel><comment>【営】 全日 7:00-22:00
      【休】 無休
      【P】 なし
      【ATM】 なし</comment>
    </gnr_info>
    <distance>252</distance>
    <lon>139.76944613873</lon>
    <lat>35.679819588244</lat>
  </result>
  <result>
    <name>JPローソン東京中央局店</name>
    <yomi>ジェーピーローソントウキョウチュウオウキョクテン</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>買う</gnr_name>
    </gnr_info>
    <gnr_info>
      <gnr_stepno>2</gnr_stepno>
      <gnr_name>コンビニエンスストア</gnr_name>
    </gnr_info>
```

```
<addr>東京都千代田区丸の内2-7-2</addr>
<tel>03-3287-9001</tel>
<comment>【営】 全日 7:00-23:00
【休】 不定休
【P】 なし
【ATM】 一部提携金融機関</comment>
<distance>260</distance>
<lon>139.7648295011</lon>
<lat>35.679900455225</lat>
</result>
..
</spotareaResponse>
```

サンプルレスポンス（エラー時）

- ・緯度経度 (lonlat)が未設定の場合

```
jsonp_callback_function (
  {"status":"[E03001] spotarea parameter error (lonlat is required)"}
)
```

エラーコード一覧は付則を参照

2.4.7. スポット周辺検索（タウンページデータ検索オプション）：spotareaopt ※ ご利用には別途オプション契約が必要です

タウンページデータ(NTT タウンページ株式会社提供)を検索対象に加えた周辺スポットの検索機能を提供します。

本機能のご利用には、別途オプション契約が必要となります。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/spotareaopt?key={key}&lonlat={lonlat}&rad={rad}&name={name}&gnrcd={gnrcd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/spotareaopt?key={key}&lonlat={lonlat}&rad={rad}&name={name}&gnrcd={gnrcd}&field={field}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
lonlat	string	必須	緯度経度 測地系: 世界測地系(JGD2000) 値の表現: degree 形式 記述形式: [経度の値]+","+[緯度の値] 記述例: "139.767231,35.681196"
rad	int	任意	検索範囲 検索半径をメートル単位で指定します。 最大で 50000 (50km) まで指定可能。 指定がない場合は、1000 (1km) として結果を返却します。
name	string	任意	検索対象の絞り込み文字列(UTF-8) スペース区切りで複数のキーワードを指定可能です。 複数指定した場合は、AND 検索となります。 ※検索フィールドを指定した場合は必須

パラメータ	型	必須/任意	説明
gnrcd	string	任意	<p>ジャンルコード</p> <p>特定ジャンルのスポットのみを対象として検索を行う場合に、検索対象のジャンルコードを指定します。</p> <p>カンマ区切りで複数のジャンルコードを指定可能です。</p> <p>複数指定した場合は OR 検索となります。</p>
field	string	任意	<p>検索フィールド</p> <p>スポットの特定の属性のみを対象としてキーワード検索を行う場合に、対象とする属性(検索フィールド)を指定します。</p> <p>カンマ区切りで複数の検索フィールドを指定可能です。</p> <p>複数指定した場合は OR 検索となります。</p> <p>指定可能な検索フィールド</p> <ul style="list-style-type: none"> •name(スポット名) •yomi(スポット名読み) •gnr_name(ジャンル名) •addr(住所) •tel(電話番号) •comment(コメント) •access(アクセス情報※) <p>※スポットの最寄りの鉄道路線・駅、インターチェンジ</p>
fmt	string	任意	<p>出力フォーマット</p> <p>JSON 出力の場合は json (デフォルト)</p> <p>XML 出力の場合は xml</p> <p>なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。</p>

パラメータ	型	必須/任意	説明
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は、50 件となります。
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-
srch.mapfan.com/v1/spotareaopt?key=xxxxx&lonlat=139.767231,35.681196&rad=5000&na
me=三菱東京 UFJ&gnrcd=48&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	スポット周辺検索(タウンページデータ検索オプション)API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	スポット名	ヒットしたスポットの名称
yomi	スポット名読み	ヒットしたスポットの名称読み
gnr_info	ジャンル情報	ヒットしたスポットのジャンル情報
gnr_stepno	ジャンル階層番号	ジャンル情報の階層番号 1:第一階層ジャンル 2:第二階層ジャンル
gnr_name	ジャンル名	ヒットしたスポットのジャンル名
addr	住所	ヒットしたスポットの住所
tel	電話番号	ヒットしたスポットの電話番号
comment	コメント	ヒットしたスポットのスポット情報(コメント形式)
distance	距離	指定した緯度経度からの距離(メートル)
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(
{
  "status": "success",
  "hit": 204, "start_pos": 1,
  "end_pos": 50,
  "results": [
    {
      "name": "¥u4e09¥u83f1¥u6771¥u4eac¥uff35¥uff26¥uff2a¥u9280¥u884c¥u6771¥u4eac¥u99c5",
      "yomi": "¥u30df¥u30c4¥u30d3¥u30b7¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30e6¥u30fc¥u30a8¥u30d5¥u30b8¥u30a7¥u30fc¥u30ae¥u30f3¥u30b3¥u30a6¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30a8¥u30ad",
      "gnr_info": [
        {
          "gnr_stepno": 1,
          "gnr_name": "¥u751f¥u6d3b"
        },
        {
          "gnr_stepno": 2,
          "gnr_name": "¥uff21¥uff34¥uff2d"
        }
      ],
      "addr": "¥u6771¥u4eac¥u90fd¥u5343¥u4ee3¥u7530¥u533a¥u4e38¥u306e¥u5185¥uff11¥uff0d¥uff19¥uff0d¥uff11¥u3000¥uff11¥u968e",
      "tel": null,
      "comment": "¥u3010¥u55b6¥u3011 ¥u5168¥u65e5 7:00–24:00¥n¥u3000¥u3000¥u3000 ¥u7b2c2¥u571f¥u66dc 7:00–21:00¥n¥u3010¥u4f11¥u3011 ¥u7121¥u4f11",
      "distance": 151,
      "lon": 139.76802724783,
      "lat": 35.679995531166
    },
    {
```

```

"name": "¥u4e09¥u83f1¥u6771¥u4eac¥uff35¥uff26¥uff2a¥u9280¥u884c¥u5927¥u4e3
8¥u6771¥u4eac¥u5e97¥u30ce¥u30fc¥u30b9¥u30bf¥u30ef¥u30fc",
"yomi": "¥u30df¥u30c4¥u30d3¥u30b7¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30e6¥u3
0fc¥u30a8¥u30d5¥u30b8¥u30a7¥u30fc¥u30ae¥u30f3¥u30b3¥u30a6¥u30c0¥u30a4¥u30d
e¥u30eb¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30c6¥u30f3¥u30ce¥u30fc¥u30b9¥u30bf¥
u30ef¥u30fc",
"gnr_info":[
{
"gnr_stepno":1,
"gnr_name":"¥u751f¥u6d3b"
},
{
"gnr_stepno":2,
"gnr_name":"¥uff21¥uff34¥uff2d"
}
],
"addr":"¥u6771¥u4eac¥u90fd¥u5343¥u4ee3¥u7530¥u533a¥u4e38¥u306e¥u5185¥uff
11¥uff0d¥uff19¥uff0d¥uff11¥u3000¥u5730¥u4e0b¥uff11¥u968e",
"tel":null,
"comment":"¥u3010¥u55b6¥u3011 ¥u5168¥u65e5 10:00-
19:00¥n¥u3010¥u4f11¥u3011 ¥u4e0d¥u5b9a¥u4f11",
"distance":164,
"lon":139.76896215119,
"lat":35.681630381229
},
..
)

```

サンプルレスポンス（正常時:XML 返却）

```
<spotareaoptResponse>
  <status>success</status>
  <hit>204</hit>
  <start_pos>1</start_pos>
  <end_pos>50</end_pos>
  <result>
    <name>三菱東京UFJ銀行東京駅</name>
    <yomi>ミツビシトウキョウユーエフジェーギンコウトウキョウエキ</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>生活</gnr_name>
    </gnr_info>
    <gnr_info>
      <gnr_stepno>2</gnr_stepno>
      <gnr_name>ATM</gnr_name>
    </gnr_info>
    <addr>東京都千代田区丸の内1-9-1 1階</addr>
    <tel/>
    <comment>【営】 全日 7:00-24:00
      第2土曜 7:00-21:00
    【休】 無休</comment>
    <distance>151</distance>
    <lon>139.76802724783</lon>
    <lat>35.679995531166</lat>
  </result>
  <result>
    <name>三菱東京UFJ銀行大丸東京店ノースタワー</name>
    <yomi>ミツビシトウキョウユーエフジェーギンコウダイマルトウキョウテンノースタワ
    ー</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>生活</gnr_name>
    </gnr_info>
    <gnr_info>
```

```
<gnr_stepno>2</gnr_stepno>
<gnr_name>ATM</gnr_name>
</gnr_info>
<addr>東京都千代田区丸の内1-9-1 地下1階</addr>
<tel/>
<comment>【営】 全日 10:00-19:00
【休】 不定休</comment>
<distance>164</distance>
<lon>139.76896215119</lon>
<lat>35.681630381229</lat>
</result>
..
</spotareaoptResponse>
```

サンプルレスポンス（エラー時）

- ・ 緯度経度 (lonlat)が未設定の場合

```
jsonp_callback_function (
  {"status": "[E03001] spotareaopt parameter error (lonlat is required)"}
)
```

エラーコード一覧は付則を参照

2.4.8. 駅検索 : sta

駅検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/sta?key={key}&name={name}&gateway={gateway}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/sta?key={key}&name={name}&gateway={gateway}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
name	string	必須	検索対象の駅名称文字列(UTF-8) スペース区切りで複数のキーワードを指定可能です。 複数指定した場合は、AND 検索となります。 末尾の“駅”は省略可能ですが、省略することで異なる検索結果となる場合があります。
gateway	int	任意	駅出入口検索設定 駅出入口を検索対象に含めるかを指定します。 含めない場合は 0 (デフォルト) 含める場合は 1
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。

パラメータ	型	必須/任意	説明
num	int	任意	<p>取得件数</p> <p>検索結果の返却件数を指定します。</p> <p>最大で 500 件まで指定可能。</p> <p>指定がない場合は、50 件となります。</p> <p>駅出入口を検索対象に含めた場合、駅検索結果、駅出入口検索結果それぞれの返却件数に適用されます。</p>
pos	int	任意	<p>取得開始位置</p> <p>検索結果の返却開始位置を指定します。</p> <p>指定がない場合は、1 件目から取得件数分の結果を返却します。</p>
callback	string	任意	<p>JSONP のコールバック用関数名</p> <p>関数名が指定されている場合は JSONP 形式で結果を返却します。</p> <p>指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。</p>

サンプルリクエスト

```
http://api-srch.mapfan.com/v1/sta?key=xxxxx&name =川崎駅&gateway=0
&callback=jsonp_callback_function
```

レスポンス

駅出入口を検索対象に含めない場合

パラメータ	値	説明
status	ステータス	駅検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	駅名(路線名)	ヒットした駅名称文字列
yomi	駅名読み	ヒットした駅名称文字列読み
typecd	鉄道分類コード	ヒットした駅の鉄道分類コード
corpcd	鉄道会社コード	ヒットした駅の鉄道会社コード
linecd	鉄道路線コード	ヒットした駅の鉄道路線コード
lineseq	路線内シーケンス 番号	路線内における駅の並び順
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

駅出入口を検索対象に含める場合

パラメータ	値	説明
status	ステータス	駅検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
station	駅検索データ	駅の検索結果、検索データを格納

パラメータ	値	説明
status	ステータス	駅検索結果 正常に完了した場合は“success”を返します。 該当データがない場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	駅名(路線名)	ヒットした駅の駅名称文字列
yomi	駅名読み	ヒットした駅の駅名称文字列読み
typecd	鉄道分類コード	ヒットした駅の鉄道分類コード
corpcd	鉄道会社コード	ヒットした駅の鉄道会社コード
linecd	鉄道路線コード	ヒットした駅の鉄道路線コード
lineseq	路線内シーケンス番号	路線内における駅の並び順
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度
stationGateway	駅出入口検索データ	駅出入口の検索結果、検索データを格納
status	ステータス	駅出入口検索結果 正常に完了した場合は“success”を返します。 該当データがない場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	名称	ヒットした駅出入口の駅名称と駅出入口名称を連結した文字列
yomi	読み	ヒットした駅出入口の駅名称読みと駅出入口名称読みを連結した文字列
train_infos	鉄道情報群	駅出入口に該当する鉄道情報群
num	件数	鉄道情報群の件数
train_info	鉄道情報	駅出入口に該当する鉄道情報を階層分(鉄道分類、鉄道会社、鉄道路線、駅、駅出入口)格納

パラメータ	値	説明
num	件数	鉄道情報の件数
tree_code	階層コード	鉄道情報の階層コード 鉄道分類の場合は 0 鉄道会社の場合は 1 鉄道路線の場合は 2 駅の場合は 3 駅出入口の場合は 4
code	コード	階層コードに該当する鉄道情報のコード
name	名称	階層コードに該当する鉄道情報の名称
yomi	読み	階層コードに該当する鉄道情報の名称読み
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

駅出入口を検索対象に含めない場合

```
jsonp_callback_function(  
{  
  "status": "success",  
  "hit": 11,  
  "start_pos": 1,  
  "end_pos": 11,  
  "results": [  
    {  
      "name": "¥u5ddd¥u5d0e¥u99c5¥uff08¥u5357¥u6b66¥u7dda¥uff09",  
      "yomi": "¥u30ab¥u30ef¥u30b5¥u30ad¥u30a8¥u30ad",  
      "typecd": "01",  
      "corpcd": "002",  
      "linecd": "028",  
      "lineseq": 1,  
      "lon": 139.69694514902,  
      "lat": 35.531366287756  
    },  
    {  
      "name": "¥u5ddd¥u5d0e¥u99c5¥uff08¥u4eac¥u6d5c¥u6771¥u5317¥u7dda¥uff09",  
      "yomi": "¥u30ab¥u30ef¥u30b5¥u30ad¥u30a8¥u30ad",  
      "typecd": "01",  
      "corpcd": "002",  
      "linecd": "033",  
      "lineseq": 31,  
      "lon": 139.69694514902,  
      "lat": 35.531366287756  
    },  
    ..  
  ]  
})
```

駅出入口を検索対象に含める場合

```
jsonp_callback_function(
{
  "status": "success",
  "station":
  {
    "status": "success",
    "hit": 4,
    "start_pos": 1,
    "end_pos": 4,
    "results": [
      {
        "name": "¥u6b66¥u8535¥u5c0f¥u6749¥u99c5¥uff08¥u6771¥u6025¥u76ee¥u9ed2¥u7dda¥uff09",
        "yomi": "¥u30e0¥u30b5¥u30b7¥u30b3¥u30b9¥u30ae¥u30a8¥u30ad",
        "typecd": "02",
        "corpcd": "072",
        "linecd": "002",
        "lineseq": 11,
        "lon": 139.65957391133,
        "lat": 35.575615836259
      },
      ..
    ]
  },
  "stationGateway":
  {
    "status": "success",
    "hit": 7,
    "start_pos": 1,
    "end_pos": 7,
    "results": [
      {
        "name": "¥u6b66¥u8535¥u5c0f¥u6749¥u99c5¥u5317¥u53e3",
        "yomi": "¥u30e0¥u30b5¥u30b7¥u30b3¥u30b9¥u30ae¥u30a8¥u30ad¥u30ad¥u30bf¥u30b0¥u30c1",

```

```

"train_infos":
{
  "num":1,
  "item":[
    {
      "train_info":
      {
        "num":5,
        "item":[
          {
            "tree_code":0,
            "code":"01",
            "name":"¥uff2a¥uff32",
            "yomi":"¥u30b8¥u30a7¥u30a4¥u30a2¥u30fc¥u30eb"
          },
          {
            "tree_code":1,
            "code":"002",
            "name":"¥u5e79¥u7dda",
            "yomi":"¥u30ab¥u30f3¥u30bb¥u30f3"
          },
          {
            "tree_code":2,
            "code":"028",
            "name":"¥u5357¥u6b66¥u7dda",
            "yomi":"¥u30ca¥u30f3¥u30d6¥u30bb¥u30f3"
          },
          {
            "tree_code":3,
            "code":7,
            "name":"¥u6b66¥u8535¥u5c0f¥u6749¥u99c5",
            "yomi":"¥u30e0¥u30b5¥u30b7¥u30b3¥u30b9¥u30ae¥u30a8¥u
30ad"
          },
          {
            "tree_code":4,

```

```

        "code":4,
        "name":"¥u5317¥u53e3",
        "yomi":"¥u30ad¥u30bf¥u30b0¥u30c1"
    }
]
}
}
]
},
"lon":139.6594697922,
"lat":35.576846575312
},
..
]
},
)

```

サンプルレスポンス（正常時:XML 返却）

駅出入口を検索対象に含めない場合

```

<staResponse>
  <status>success</status>
  <hit>11</hit>
  <start_pos>1</start_pos>
  <end_pos>11</end_pos>
  <result>
    <name>川崎駅(南武線)</name>
    <yomi>カワサキエキ</yomi>
    <typecd>01</typecd>
    <corpcd>002</corpcd>
    <linecd>028</linecd>
    <lineseq>1</lineseq>
    <lon>139.69694514902</lon>
    <lat>35.531366287756</lat>
  </result>
  <result>

```

```
<name>川崎駅(京浜東北線)</name>
<yomi>カワサキエキ</yomi>
<typecd>01</typecd>
<corpcd>002</corpcd>
<linecd>033</linecd>
<lineseq>31</lineseq>
<lon>139.69694514902</lon>
<lat>35.531366287756</lat>
</result>
..
</staResponse>
```

駅出入口を検索対象に含める場合

```
<staResponse>
  <status>success</status>
  <station>
    <status>success</status>
    <hit>4</hit>
    <start_pos>1</start_pos>
    <end_pos>4</end_pos>
    <result>
      <name>武蔵小杉駅(東急目黒線)</name>
      <yomi>ムサシコスギエキ</yomi>
      <typecd>02</typecd>
      <corpcd>072</corpcd>
      <linecd>002</linecd>
      <lineseq>11</lineseq>
      <lon>139.65957391133</lon>
      <lat>35.575615836259</lat>
    </result>
    ..
  </station>
  <stationGateway>
    <status>success</status>
    <hit>7</hit>
    <start_pos>1</start_pos>
```



```
<end_pos>7</end_pos>
<result>
  <name>武蔵小杉駅北口</name>
  <yomi>ムサシコスギエキタグチ</yomi>
  <train_infos>
    <num>1</num>
    <item>
      <train_info>
        <num>5</num>
        <item>
          <tree_code>0</tree_code>
          <code>01</code>
          <name>JR</name>
          <yomi>ジェイアール</yomi>
        </item>
        <item>
          <tree_code>1</tree_code>
          <code>002</code>
          <name>幹線</name>
          <yomi>カンセン</yomi>
        </item>
        <item>
          <tree_code>2</tree_code>
          <code>028</code>
          <name>南武線</name>
          <yomi>ナンブセン</yomi>
        </item>
        <item>
          <tree_code>3</tree_code>
          <code>7</code>
          <name>武蔵小杉駅</name>
          <yomi>ムサシコスギエキ</yomi>
        </item>
        <item>
          <tree_code>4</tree_code>
          <code>4</code>
```

```
        <name>北口</name>
        <yomi>キタグチ</yomi>
      </item>
    </train_info>
  </item>
</train_infos>
<lon>139.6594697922</lon>
<lat>35.576846575312</lat>
</result>
..
</stationGateway>
</staResponse>
```

サンプルレスポンス（エラー時）

- ・ 検索対象の駅名称文字列(name)が未設定の場合

```
jsonp_callback_function (
  {"status": "[E03001] sta parameter error (name is required)"}
)
```

エラーコード一覧は付則を参照

2.4.9. 駅階層検索 : stalist

駅階層検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/stalist?key={key}&stepno={stepno}&typecd={typecd}&corpcd={corpcd}&linecd={linecd}&lineseq={lineseq}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/stalist?key={key}&stepno={stepno}&typecd={typecd}&corpcd={corpcd}&linecd={linecd}&lineseq={lineseq}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
stepno	int	必須	検索階層 鉄道分類一覧の場合は 0 鉄道会社一覧の場合は 1 鉄道路線一覧の場合は 2 駅一覧の場合は 3 駅出入口一覧の場合は 4
typecd	string	必須(*1)/ 任意	鉄道分類コード(2 桁) (*1) stepno に 1,2,3,4 を指定した場合は必須
corpcd	string	必須(*2)/ 任意	鉄道会社コード(3 桁) (*2) stepno に 2,3,4 を指定した場合は必須
linecd	string	必須(*3)/ 任意	鉄道路線コード(3 桁) (*3) stepno に 3,4 を指定した場合は必須
lineseq	int	必須(*4)/ 任意	路線内シーケンス (*4) stepno に 4 を指定した場合は必須

パラメータ	型	必須/任意	説明
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は、全件となります。
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-  
srch.mapfan.com/v1/stalist?key=xxxxx&stepno=3&typecd=01&corpcd=002&linecd=044&cal  
lback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	駅階層検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
stepno	階層番号	ヒットした鉄道情報の階層
name	名称	ヒットした鉄道情報の名称
yomi	読み	ヒットした鉄道情報の名称読み
typecd	鉄道分類コード	ヒットした駅の鉄道分類コード
corpcd	鉄道会社コード	ヒットした駅の鉄道会社コード
linecd	鉄道路線コード	ヒットした駅の鉄道路線コード
lineseq(*1)	路線内シーケンス番号	路線内における駅の並び順
stationseq(*2)	駅内シーケンス番号	駅内における出入口の番号
lon(*1)	経度	世界測地系(JGD2000)degree 形式の経度
lat(*1)	緯度	世界測地系(JGD2000)degree 形式の緯度

*1 : stepno が 3（駅一覧）又は 4（駅出入口一覧）の場合のみ、出力されます

*2 : stepno が 4（駅出入口一覧）の場合のみ、出力されます

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(  
{  
  "status": "success",  
  "hit": 154,  
  "start_pos": 1,  
  "end_pos": 154,  
  "results": [  
    {  
      "stepno": 3,  
      "name": "¥u6771¥u4eac¥u99c5¥uff08¥u6771¥u6d77¥u9053¥u672c¥u7dda¥uff09",  
      "yomi": "¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30a8¥u30ad",  
      "typecd": "01",  
      "corpcd": "002",  
      "linecd": "044",  
      "lineseq": 1,  
      "lon": 139.76729328863,  
      "lat": 35.680958407977  
    },  
    {  
      "stepno": 3,  
      "name": "¥u65b0¥u6a4b¥u99c5¥uff08¥u6771¥u6d77¥u9053¥u672c¥u7dda¥uff09",  
      "yomi": "¥u30b7¥u30f3¥u30d0¥u30b7¥u30a8¥u30ad",  
      "typecd": "01",  
      "corpcd": "002",  
      "linecd": "044",  
      "lineseq": 2,  
      "lon": 139.75814153711,  
      "lat": 35.666347760906  
    },  
    ..  
  ]  
})
```

サンプルレスポンス（正常時:XML 返却）

```
<stalistResponse>
  <status>success</status>
  <hit>154</hit>
  <start_pos>1</start_pos>
  <end_pos>154</end_pos>
  <result>
    <stepno>3</stepno>
    <name>東京駅(東海道本線)</name>
    <yomi>トウキョウエキ</yomi>
    <typecd>01</typecd>
    <corpcd>002</corpcd>
    <linecd>044</linecd>
    <lineseq>1</lineseq>
    <lon>139.76729328863</lon>
    <lat>35.680958407977</lat>
  </result>
  <result>
    <stepno>3</stepno>
    <name>新橋駅(東海道本線)</name>
    <yomi>シンバシエキ</yomi>
    <typecd>01</typecd>
    <corpcd>002</corpcd>
    <linecd>044</linecd>
    <lineseq>2</lineseq>
    <lon>139.75814153711</lon>
    <lat>35.666347760906</lat>
  </result>
  ..
</stalistResponse>
```

サンプルレスポンス（エラー時）

- ・ 検索階層(stepno)が未設定の場合

```
jsonp_callback_function (
{"status": "[E03001] stalist parameter error (stepno is required)"}
)
```

エラーコード一覧は付則を参照

2.4.10.最寄駅検索 : staarea

指定された緯度経度を中心とした周辺駅の検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/staarea?key={key}&lonlat={lonlat}&rad={rad}&gateway={gateway}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/staarea?key={key}&lonlat={lonlat}&rad={rad}&gateway={gateway}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
lonlat	string	必須	緯度経度 測地系: 世界測地系(JGD2000) 値の表現: degree 形式 記述形式: [経度の値]+","+[緯度の値] 記述例: "139.767231,35.681196"
rad	int	任意	検索範囲 検索半径をメートル単位で指定します。 最大で 50000 (50km) まで指定可能。 指定がない場合は、1000 (1km) として結果を返却します。
gateway	int	任意	駅出入口検索設定 駅出入口を検索対象に含めるかを指定します。 含めない場合は 0 (デフォルト) 含める場合は 1
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。

パラメータ	型	必須/任意	説明
num	int	任意	<p>取得件数</p> <p>検索結果の返却件数を指定します。</p> <p>最大で 500 件まで指定可能。</p> <p>指定がない場合は、50 件となります。</p> <p>駅出入口を検索対象に含めた場合、駅検索結果、駅出入口検索結果それぞれの返却件数に適用されます。</p>
pos	int	任意	<p>取得開始位置</p> <p>検索結果の返却開始位置を指定します。</p> <p>指定がない場合は、1 件目から取得件数分の結果を返却します。</p>
callback	string	任意	<p>JSONP のコールバック用関数名</p> <p>関数名が指定されている場合は JSONP 形式で結果を返却します。</p> <p>指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。</p>

サンプルリクエスト

```
http://api-
srch.mapfan.com/v1/staarea?key=xxxxx&lonlat=139.767231,35.681196&rad=5000&gateway
=0&callback=jsonp_callback_function
```

レスポンス

駅出入口を検索対象に含めない場合

パラメータ	値	説明
status	ステータス	最寄駅検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラー メッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	駅名(路線名)	ヒットした駅名称文字列
yomi	駅名読み	ヒットした駅名称文字列読み
distance	距離	指定した緯度経度からの距離(メートル)
typecd	鉄道分類コード	ヒットした駅の鉄道分類コード
corpcd	鉄道会社コード	ヒットした駅の鉄道会社コード
linecd	鉄道路線コード	ヒットした駅の鉄道路線コード
lineseq	路線内シーケンス 番号	路線内における駅の並び順
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

駅出入口を検索対象に含める場合

パラメータ	値	説明
status	ステータス	最寄駅検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメ ッセージ”を返します。
station	駅検索データ	駅の検索結果、検索データを格納

パラメータ	値	説明
status	ステータス	駅検索結果 正常に完了した場合は“success”を返します。 該当データがない場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	駅名(路線名)	ヒットした駅の駅名称文字列
yomi	駅名読み	ヒットした駅の駅名称文字列読み
distance	距離	指定した緯度経度からの距離(メートル)
typecd	鉄道分類コード	ヒットした駅の鉄道分類コード
corpcd	鉄道会社コード	ヒットした駅の鉄道会社コード
linecd	鉄道路線コード	ヒットした駅の鉄道路線コード
lineseq	路線内シーケンス番号	路線内における駅の並び順
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度
stationGateway	駅出入口検索データ	駅出入口の検索結果、検索データを格納
status	ステータス	駅出入口検索結果 正常に完了した場合は“success”を返します。 該当データがない場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	名称	ヒットした駅出入口の駅名称と駅出入口名称を連結した文字列
yomi	読み	ヒットした駅出入口の駅名称読みと駅出入口名称読みを連結した文字列
distance	距離	指定した緯度経度からの距離(メートル)
train_infos	鉄道情報群	駅出入口に該当する鉄道情報群
num	件数	鉄道情報群の件数

パラメータ	値	説明
train_info	鉄道情報	駅出入口に該当する鉄道情報を階層分(鉄道分類、鉄道会社、鉄道路線、駅、駅出入口)格納
num	件数	鉄道情報の件数
tree_code	階層コード	鉄道情報の階層コード 鉄道分類の場合は 0 鉄道会社の場合は 1 鉄道路線の場合は 2 駅の場合は 3 駅出入口の場合は 4
code	コード	階層コードに該当する鉄道情報のコード
name	名称	階層コードに該当する鉄道情報の名称
yomi	読み	階層コードに該当する鉄道情報の名称読み
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

駅出入口を検索対象に含めない場合

```
jsonp_callback_function(  
  {  
    "status": "success",  
    "hit": 220,  
    "start_pos": 1,  
    "end_pos": 50,  
    "results": [  
      {  
        "name": "¥u6771¥u4eac¥u99c5¥uff08¥u5c71¥u624b¥u7dda¥uff09",  
        "yomi": "¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30a8¥u30ad",  
        "distance": 27,  
        "typecd": "01",  
        "corpcd": "002",  
        "linecd": "031",  
        "lineseq": 13,  
        "lon": 139.76729328863,  
        "lat": 35.680958407977  
      },  
      {  
        "name": "¥u6771¥u4eac¥u99c5¥uff08¥u6771¥u5317¥u7e26¥u8cab¥u7dda¥uff09",  
        "yomi": "¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30a8¥u30ad",  
        "distance": 27,  
        "typecd": "01",  
        "corpcd": "002",  
        "linecd": "090",  
        "lineseq": 1,  
        "lon": 139.76729328863,  
        "lat": 35.680958407977  
      },  
      ..  
    ]  
  }  
)
```

駅出入口を検索対象に含める場合

```
jsonp_callback_function(
{
  "status": "success",
  "station":
  {
    "status": "success",
    "hit": 5,
    "start_pos": 1,
    "end_pos": 5,
    "results": [
      {
        "name": "¥u5ddd¥u5d0e¥u99c5¥uff08¥u5357¥u6b66¥u7dda¥uff09",
        "yomi": "¥u30ab¥u30ef¥u30b5¥u30ad¥u30a8¥u30ad",
        "distance": 20,
        "typecd": "01",
        "corpcd": "002",
        "linecd": "028",
        "lineseq": 1,
        "lon": 139.69694514902,
        "lat": 35.531366287756
      },
      ..
    ]
  },
  "stationGateway":
  {
    "status": "success",
    "hit": 6,
    "start_pos": 1,
    "end_pos": 6,
    "results": [
      {
        "name": "¥u5ddd¥u5d0e¥u99c5¥u897f¥u53e3",
        "yomi":
¥u30ab¥u30ef¥u30b5¥u30ad¥u30a8¥u30ad¥u30cb¥u30b7¥u30b0¥u30c1",
```

```

    "distance": 46,
    "train_infos":
    {
        "num": 3,
        "item": [
            {
                "train_info":
                {
                    "num": 5,
                    "item": [
                        {
                            "tree_code": 0,
                            "code": "01",
                            "name": "¥uff2a¥uff32",
                            "yomi": "¥u30b8¥u30a7¥u30a4¥u30a2¥u30fc¥u30eb"
                        },
                        {
                            "tree_code": 1,
                            "code": "002",
                            "name": "¥u5e79¥u7dda",
                            "yomi": "¥u30ab¥u30f3¥u30bb¥u30f3"
                        },
                        {
                            "tree_code": 2,
                            "code": "044",
                            "name": "¥u6771¥u6d77¥u9053¥u672c¥u7dda",
                            "yomi": "¥u30c8¥u30a6¥u30ab¥u30a4¥u30c9¥u30a6¥u30db¥u
30f3¥u30bb¥u30f3"
                        },
                        {
                            "tree_code": 3,
                            "code": 4,
                            "name": "¥u5ddd¥u5d0e¥u99c5",
                            "yomi": "¥u30ab¥u30ef¥u30b5¥u30ad¥u30a8¥u30ad"
                        },
                    ],
                }
            }
        ]
    }

```



```
        "tree_code": 4,
        "code": 1,
        "name": "¥u897f¥u53e3",
        "yomi": "¥u30cb¥u30b7¥u30b0¥u30c1"
    }
]
}
},
..
]
},
"lon": 139.69668716429,
"lat": 35.531695245615
},
..
]
}
}
)
```

サンプルレスポンス（正常時:XML 返却）

駅出入口を検索対象に含めない場合

```
<staareaResponse>
  <status>success</status>
  <hit>220</hit>
  <start_pos>1</start_pos>
  <end_pos>50</end_pos>
  <result>
    <name>東京駅(山手線)</name>
    <yomi>トウキョウエキ</yomi>
    <distance>27</distance>
    <typecd>01</typecd>
    <corpcd>002</corpcd>
    <linecd>031</linecd>
    <lineseq>13</lineseq>
    <lon>139.76729328863</lon>
    <lat>35.680958407977</lat>
  </result>
  <result>
    <name>東京駅(東北縦貫線)</name>
    <yomi>トウキョウエキ</yomi>
    <distance>27</distance>
    <typecd>01</typecd>
    <corpcd>002</corpcd>
    <linecd>090</linecd>
    <lineseq>1</lineseq>
    <lon>139.76729328863</lon>
    <lat>35.680958407977</lat>
  </result>
  ..
</staareaResponse>
```

駅出入口を検索対象に含める場合

```
<staareaResponse>
  <status>success</status>
  <station>
    <status>success</status>
    <hit>5</hit>
    <start_pos>1</start_pos>
    <end_pos>5</end_pos>
    <result>
      <name>川崎駅(南武線)</name>
      <yomi>カワサキエキ</yomi>
      <distance>20</distance>
      <typecd>01</typecd>
      <corpcd>002</corpcd>
      <linecd>028</linecd>
      <lineseq>1</lineseq>
      <lon>139.69694514902</lon>
      <lat>35.531366287756</lat>
    </result>
  </station>
  <stationGateway>
    <status>success</status>
    <hit>6</hit>
    <start_pos>1</start_pos>
    <end_pos>6</end_pos>
    <result>
      <name>川崎駅西口</name>
      <yomi>カワサキエキニシグチ</yomi>
      <distance>46</distance>
      <train_infos>
        <num>3</num>
        <item>
          <train_info>
            <num>5</num>
            <item>
```

```
<tree_code>0</tree_code>
<code>01</code>
<name>JR</name>
<yomi>ジェイアール</yomi>
</item>
<item>
  <tree_code>1</tree_code>
  <code>002</code>
  <name>幹線</name>
  <yomi>カンセン</yomi>
</item>
<item>
  <tree_code>2</tree_code>
  <code>044</code>
  <name>東海道本線</name>
  <yomi>トウカイドウホンセン</yomi>
</item>
<item>
  <tree_code>3</tree_code>
  <code>4</code>
  <name>川崎駅</name>
  <yomi>カワサキエキ</yomi>
</item>
<item>
  <tree_code>4</tree_code>
  <code>1</code>
  <name>西口</name>
  <yomi>ニシグチ</yomi>
</item>
</train_info>
</item>
..
</train_infos>
<lon>139.69668716429</lon>
<lat>35.531695245615</lat>
</result>
```

```
..  
</stationGateway>  
</staareaResponse>
```

サンプルレスポンス（エラー時）

- ・緯度経度 (lonlat)が未設定の場合

```
jsonp_callback_function (  
  {"status":"[E03001] staarea parameter error (lonlat is required)"}  
)
```

エラーコード一覧は付則を参照

2.4.11.郵便番号検索 : zip

郵便番号による住所検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/zip?key={key}&zipcd={zipcd}&ot={ot}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/zip?key={key}&zipcd={zipcd}&ot={ot}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
zipcd	string	必須	検索対象の郵便番号(UTF-8) 3 桁または 7 桁の郵便番号を指定します。 7 桁の場合、半角ハイフンを含む郵便番号、ハイフンなしの郵便番号の入力が可能です。
ot	int	任意	住所出力形式 0:大口事業所、ビル名または住所 1:住所のみを出力 指定がない場合は、0(デフォルト)として大口事業所、ビル名を優先して出力します。
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は、50 件となります。

パラメータ	型	必須/任意	説明
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-srch.mapfan.com/v1/zip?key=xxxxx&zipcd=210-0024&ot=0&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	郵便番号検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
zipcd	郵便番号	ヒットした郵便番号(半角ハイフンあり7桁)
name	住所	ヒットした住所文字列
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度
prefcd	都道府県コード	都道府県コード
citycd	市区町村コード	市区町村コード
tyocd	条町丁目コード	条町丁目コード

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(  
  {  
    "status": "success",  
    "hit": 1,  
    "start_pos": 1,  
    "end_pos": 1,  
    "results": [  
      {  
        "zipcd": "210-0024",  
        "name": "〒795e〒5948〒5ddd〒770c〒5ddd〒5d0e〒5e02〒5ddd〒5d0e〒533a〒  
65e5〒9032〒753a",  
        "lon": 139.6951224283,  
        "lat": 35.528449772395,  
        "prefcd": "14",  
        "citycd": "131",  
        "tyocd": "050000"  
      }  
    ]  
  }  
)
```

サンプルレスポンス（正常時:XML 返却）

```
<zipResponse>
  <status>success</status>
  <hit>1</hit>
  <start_pos>1</start_pos>
  <end_pos>1</end_pos>
  <result>
    <zipcd>210-0024</zipcd>
    <name>神奈川県川崎市川崎区日進町</name>
    <lon>139.6951224283</lon>
    <lat>35.528449772395</lat>
    <prefcd>14</prefcd>
    <citycd>131</citycd>
    <tyocd>050000</tyocd>
  </result>
</zipResponse>
```

サンプルレスポンス（エラー時）

- ・ 検索対象の郵便番号（zipcd）が未設定の場合

```
jsonp_callback_function (
  {"status": "[E03001] zip parameter error (zipcd is required)"}
)
```

エラーコード一覧は付則を参照

2.4.12. 多言語スポット周辺検索 : spotareaml※ご利用には別途オプション契約が必要です

検索結果に外国語を加えた周辺スポットの検索機能を提供します。

本機能のご利用には、別途オプション契約が必要となります。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/spotareaml?key={key}&lonlat={lonlat}&rad={rad}&lang={lang}&gnrcd={gnrcd}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/spotareaml?key={key}&lonlat={lonlat}&rad={rad}&lang={lang}&gnrcd={gnrcd}&fmt={fmt}&num={num}&pos={pos}&callback={callback}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得
lonlat	string	必須	緯度経度 測地系: 世界測地系(JGD2000) 値の表現: degree 形式 記述形式: [経度の値]+", "+[緯度の値] 記述例: "139.767231,35.681196"
rad	int	任意	検索範囲 検索半径をメートル単位で指定します。 最大で 50000 (50km) まで指定可能。 指定がない場合は、1000 (1km) として結果を返却します。
gnrcd	string	必須	ジャンルコード 特定ジャンルのスポットのみを対象として検索を行う場合に、検索対象のジャンルコードを指定します。 カンマ区切りで複数のジャンルコードを指定可能です。 複数指定した場合は OR 検索となります。

パラメータ	型	必須/任意	説明
lang	string	必須	言語 レスポンスで返されるスポット名称(外国語)、ジャンル名称(外国語)に格納される言語を指定します。 設定値: "en"(英語) "zh2"(中国語簡体字) "zh1"(中国語繁体字) "ko"(韓国語)
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
num	int	任意	取得件数 検索結果の返却件数を指定します。 最大で 500 件まで指定可能。 指定がない場合は、50 件となります。
pos	int	任意	取得開始位置 検索結果の返却開始位置を指定します。 指定がない場合は、1 件目から取得件数分の結果を返却します。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

サンプルリクエスト

```
http://api-
srch.mapfan.com/v1/spotareaml?key=xxxxx&lonlat=139.767231,35.681196&rad=5000&lang
=zh1&gnrcd=1&callback=jsonp_callback_function
```

レスポンス

パラメータ	値	説明
status	ステータス	多言語スポット周辺検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
hit	出力件数	ヒットした検索件数
start_pos	開始位置	検索結果の出力開始位置
end_pos	終了位置	検索結果の出力終了位置
name	スポット名称	ヒットしたスポットの名称
name_ml	スポット名称(外国語)	ヒットしたスポットの外国語名称
yomi	スポット名称読み	ヒットしたスポットの名称読み
gnr_info	ジャンル情報	ヒットしたスポットのジャンル情報
gnr_stepno	ジャンル階層番号	ジャンル情報の階層番号 1:第一階層ジャンル 2:第二階層ジャンル
gnr_name	ジャンル名称	ヒットしたスポットのジャンル名称
gnr_name_ml	ジャンル名称(外国語)	ヒットしたスポットの外国語ジャンル名称
addr	住所	ヒットしたスポットの住所
tel	電話番号	ヒットしたスポットの電話番号
comment	コメント	ヒットしたスポットのスポット情報(コメント形式)
distance	距離	指定した緯度経度からの距離(メートル)
lon	経度	世界測地系(JGD2000)degree 形式の経度
lat	緯度	世界測地系(JGD2000)degree 形式の緯度

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(
{
  "status": "success",
  "hit": 31,
  "start_pos": 1,
  "end_pos": 31,
  "results": [[
    "name": "¥u6771¥u4eac¥u99c5¥uff08¥u6771¥u5317¥u65b0¥u5e79¥u7dda¥uff09",
    "name_ml": "¥u6771¥u4eac¥u7ad9 (¥u6771¥u5317¥u65b0¥u5e79¥u7dda)",
    "yomi": "¥u30c8¥u30a6¥u30ad¥u30e7¥u30a6¥u30a8¥u30ad",
    "gnr_info": [{
      "gnr_stepno": 1,
      "gnr_name": "¥u8eca¥u30fb¥u4ea4¥u901a",
      "gnr_name_ml": "¥u8eca¥u00b7¥u4ea4¥u901a"
    }, {
      "gnr_stepno": 2,
      "gnr_name": "¥u99c5",
      "gnr_name_ml": "¥u8eca¥u7ad9"
    }],
    "addr":
    "¥u6771¥u4eac¥u90fd¥u5343¥u4ee3¥u7530¥u533a¥u4e38¥u306e¥u5185¥uff11¥u4e01¥u7
    6ee¥uff19¥uff0d¥uff11",
    "tel": null,
    "comment": null,
    "distance": 27,
    "lon": 139.76729328863,
    "lat": 35.680958407977
  ], {
    "name":
    "¥u6771¥u4eac¥u99c5¥uff08¥u6771¥u6d77¥u9053¥u65b0¥u5e79¥u7dda¥uff09",
    ..
  })
```

サンプルレスポンス（正常時:XML 返却）

```
<spotareamlResponse>
  <status>success</status>
  <hit>31</hit>
  <start_pos>1</start_pos>
  <end_pos>31</end_pos>
  <result>
    <name>東京駅(東北新幹線)</name>
    <name_ml>東京站 (東北新幹線)</name_ml>
    <yomi>トウキョウエキ</yomi>
    <gnr_info>
      <gnr_stepno>1</gnr_stepno>
      <gnr_name>車・交通</gnr_name>
      <gnr_name_ml>車・交通</gnr_name_ml>
    </gnr_info>
    <gnr_info>
      <gnr_stepno>2</gnr_stepno>
      <gnr_name>駅</gnr_name>
      <gnr_name_ml>車站</gnr_name_ml>
    </gnr_info>
    <addr>東京都千代田区丸の内1丁目9－1</addr>
    <tel/>
    <comment/>
    <distance>27</distance>
    <lon>139.76729328863</lon>
    <lat>35.680958407977</lat>
  </result>
  <result>
    <name>東京駅(東海道新幹線)</name>
    ..
  </spotareamlResponse>
```

サンプルレスポンス（エラー時）

- ・緯度経度 (lonlat)が未設定の場合

```
jsonp_callback_function (
  {"status": "[E03001] spotareaml parameter error (lonlat is required)"}
)
```

エラーコード一覧は付則を参照

2.5. ルート機能

2.5.1. ルート検索 : calcroute

ルート検索機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-pre.mapfan.com/v1/calcroute?key={key}&fmt={fmt}&callback={callback}&routeid={routeid}&start={start}&via={via}&destination={destination}&date={date}&priority={priority}&ferry={ferry}&smartic={smartic}&tollway={tollway}&cartype={cartype}&normalspeed={normalspeed}&highwayspeed={highwayspeed}&tollwayspeed={tollwayspeed}&vehicletype={vehicletype}&height={height}&weight={weight}&width={width}&danger={danger}&daytime={daytime}&generalroad={generalroad}&tollroad={tollroad}&travel={travel}&passablearea={passablearea}&impassablearea={impassablearea}&tolltarget={tolltarget}		
本番利用 URL	http(s)://api-route.mapfan.com/v1/calcroute?key={key}&fmt={fmt}&callback={callback}&routeid={routeid}&start={start}&via={via}&destination={destination}&date={date}&priority={priority}&ferry={ferry}&smartic={smartic}&tollway={tollway}&cartype={cartype}&normalspeed={normalspeed}&highwayspeed={highwayspeed}&tollwayspeed={tollwayspeed}&vehicletype={vehicletype}&height={height}&weight={weight}&width={width}&danger={danger}&daytime={daytime}&generalroad={generalroad}&tollroad={tollroad}&travel={travel}&passablearea={passablearea}&impassablearea={impassablearea}&tolltarget={tolltarget}		
パラメータ	型	必須/任意	説明
key	string	必須	アクセスに使用するキー ※認証 API より取得

パラメータ	型	必須/任意	説明
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。
routeid	string	任意	ルート識別子 ルート検索を識別する任意の文字列を指定します。 検索結果に特に影響はしません。 文字コードは UTF-8。 半角のダブルクォーテーションの使用は禁止ですが、それ以外の半角記号や全角文字は URL エンコードを行っていれば使用可能となります。
start	string	必須	出発地緯度経度 測地系: 世界測地系(JGD2000) 値の表現: degree 形式 記述形式: [経度] + “,” + [緯度] 又は [経度] + “,” + [緯度] + “,” + [地点種別] 地点種別設定値: 一般道・有料道の場合 0 一般道のみの場合 1 有料道のみの場合 2 記述例: “139.767231,35.681196,2” 経度緯度の小数点以下は最大 8 桁までとします。 地点種別が未指定の場合、地点種別に 0 を指定した場合と同じになります。

パラメータ	型	必須/任意	説明
via	string	任意	<p>経由地緯度経度</p> <p>記述形式:</p> <p>[経由地#1] + " " + [経由地#2] + " " ... [経由地#n]</p> <p>記述例:</p> <p>"139.738909,35.629104,1 139.710369,35.728974 139.715785,35.634264,2"</p> <p>n は経由地の個数で、最大 30 件まで指定可能。 経由地の測地系や値の記述形式は start と同様。</p>
destination	string	必須	<p>目的地緯度経度</p> <p>測地系や値の記述形式は start と同様。</p>
date	string	任意	<p>出発日時</p> <p>記述形式:</p> <p>[年 4 桁]+[月 2 桁]+[日 2 桁]+"_"+[時 2 桁]+[分 2 桁]+[秒 2 桁]</p> <p>記述例:</p> <p>2014 年 8 月 9 日午後 5 時 16 分 21 秒 = "20140809_171621"</p> <p>ルート検索時の交通規制を考慮するときの日時を指定します。なお、本指定がない場合、API を実行した日時を適用して、ルート検索を行います。</p>
priority	int	任意	<p>基本条件</p> <p>標準の場合は 0(デフォルト)</p> <p>距離優先の場合は 1</p> <p>直進優先の場合は 2</p> <p>簡易歩行者の場合は 3</p> <p>道幅優先の場合は 4</p> <p>歩行者標準(分かり易い)の場合は 100</p> <p>歩行者距離優先の場合は 101</p> <p>歩行者屋根優先(屋根が多い)の場合は 102</p> <p>歩行者段差回避(階段が少ない)の場合は 103</p> <p>巡回ルートによる検索を有効とした場合、歩行者(100 ~ 103)の指定は出来ません。</p>

パラメータ	型	必須/任意	説明
ferry	int	任意	フェリー使用条件 標準の場合は 0(デフォルト) 優先の場合は 1 回避の場合は 2※ 絶対回避の場合は 3※ ※フェリーを使わないとルートが引けない条件で、 回避を選んだ場合、フェリーを使ったルートが結果として返りますが、絶対回避を選んだ場合、エラー(I00107)となります。
smartic	int	任意	スマート IC 利用有無 スマート IC 利用しない場合は 0 スマート IC 利用する場合は 1(デフォルト)
tollway	int	任意	有料道路使用条件 標準の場合は 0(デフォルト) 優先の場合は 1 回避の場合は 2※ 絶対回避の場合は 3※ ※有料道路を使わないとルートが引けない条件で、 回避を選んだ場合、有料道路を使ったルートが結果として返りますが、絶対回避を選んだ場合、エラー(I00108)となります。
cartype	int	任意	有料道路利用時の車種 軽自動車の場合は 0 普通車の場合は 1(デフォルト) 中型車の場合は 2 大型車の場合は 3 特大車の場合は 4 本設定は有料道路の料金計算のみに使用します。
normalspeed	float	任意	一般道の走行速度 速度単位: km/h 記述例: 25.0 本設定は一般道の走行時間の計算に用います。経路種別との関係およびデフォルト値は付則.経路種別資料一覧を参照。

パラメータ	型	必須/任意	説明
highwayspeed	float	任意	<p>高速道の走行速度</p> <p>速度単位 : km/h</p> <p>記述例 : 90.0</p> <p>本設定は高速道の走行時間の計算に用います。経路種別との関係およびデフォルト値は付則.経路種別資料一覧を参照。</p>
tollwayspeed	float	任意	<p>有料道の走行速度</p> <p>速度単位 : km/h</p> <p>記述例 : 55.0</p> <p>本設定は有料道の走行時間の計算に用います。経路種別との関係およびデフォルト値は付則.経路種別資料一覧を参照。</p>
vehicletype	int	任意	<p>車種</p> <p>本設定は、道路規制に使用します。</p> <p>値は付則.規制車種一覧を参照。</p>
height	int	任意	<p>車高</p> <p>単位 : cm</p> <p>本設定は、道路規制に使用します。0cm 以上の値を設定して下さい。</p>
weight	int	任意	<p>車重</p> <p>単位 : kg</p> <p>本設定は、道路規制に使用します。0kg 以上の値を設定して下さい。</p>
width	int	任意	<p>車幅</p> <p>単位 : cm</p> <p>本設定は、道路規制に使用します。0cm 以上の値を設定して下さい。</p>
danger	int	任意	<p>危険物積載車両</p> <p>危険物積載車両でない場合は 0（デフォルト）</p> <p>危険物積載車両である場合は 1</p> <p>本設定は、道路規制に使用します。</p>

パラメータ	型	必須/任意	説明
daytime	int	任意	<p>日時規制</p> <p>日時規制を無効にする場合は 0</p> <p>日時規制を有効にする場合は 1 (デフォルト)</p> <p>なお、一般道規制と高速道規制を共に無効にすると日時規制は無効になります。</p>
generalroad	int	任意	<p>一般道規制</p> <p>一般道規制を無効にする場合は 0</p> <p>一般道規制を有効にする場合は 1 (デフォルト)</p> <p>無効とした場合、一般道の規制を無視したルート検索が可能となります。</p>
tollroad	int	任意	<p>高速道規制</p> <p>高速道規制を無効にする場合は 0</p> <p>高速道規制を有効にする場合は 1 (デフォルト)</p> <p>無効とした場合、高速道の規制を無視したルート検索が可能となります。</p> <p>高速道規制の対象となる道は道路種別 100 番台と 11 番の道です。</p> <p>道路種別は付則.道路種別一覧を参照。</p>
travel	int	任意	<p>巡回ルート設定</p> <p>巡回ルートによる検索を無効にする場合は 0 (デフォルト)</p> <p>巡回ルートによる検索を有効にする場合は 1</p> <p>有効とした場合、効率よく経由地を回るルート検索を行うことができます。</p> <p>歩行者(priority=100~103)、通行可能エリア、通行不可エリアが指定されている場合、巡回ルートによる検索を有効にする指定は出来ません。</p>

パラメータ	型	必須/任意	説明
passablearea	string	任意	<p>通行可能エリア</p> <p>本設定はルート検索対象となるエリアを矩形で指定します。</p> <p>記述形式:</p> <p>[エリア#1] + " " + [エリア#2] + " " ... [エリア#n]</p> <p>エリア記述形式:</p> <p>[南西経度の値] + "," + [南西緯度の値] + "," + [北東経度の値] + "," + [北東緯度の値]</p> <p>記述例:</p> <p>139.6965,35.6867,139.7029,35.6921 139.7000,35.6818,139.7044,35.6879</p> <p>経度、緯度の値は世界測地系(JGD2000)の degree 形式。</p> <p>n はエリアの個数で、最大 10 件まで指定可能。</p> <p>小数点以下は最大 8 桁までとします。</p> <p>通行可能エリアと通行不可エリアで重なった部分については、通行不可エリアが優先されます。</p> <p>巡回ルートによる検索を有効とした場合、本設定は指定は出来ません。</p>
impassablearea	string	任意	<p>通行不可エリア</p> <p>本設定はルート検索対象外となるエリアを矩形で指定します。</p> <p>記述形式は通行可能エリアと同様。</p> <p>通行可能エリアと通行不可エリアで重なった部分については、通行不可エリアが優先されます。</p> <p>巡回ルートによる検索を有効とした場合、本設定は指定は出来ません。</p>
tolltarget	int	任意	<p>料金計算対象</p> <p>通常料金の場合は 0(デフォルト)</p> <p>通常料金+ETC 割引料金の場合は 1</p>

サンプルリクエスト

```
http://api-  
route.mapfan.com/v1/calcroute?key=xxxxx&callback=jsonp_callback_function&routeid=calcrou  
tesample&start=139.767231,35.681196,2&via=139.738909,35.629104|139.713369,35.728974,2|1  
39.715785,35.634264&destination=139.700494,35.689957&date=20140809_171621&priority=1&f  
erry=3&smartic=1&tollway=2&cartype=1&normalspeed=25.0&highwayspeed=90.0&tollwayspeed  
=55.0&vehicletype=1&height=129&weight=1420&width=190&danger=0&daytime=1&generalroad  
=1&tollroad=1&travel=0&passablearea=139.68800254,35.62266068,139.74671073,35.75344058|  
139.73847098,35.67176313,139.77692313,35.74814663&impassablearea=139.73847098,35.6831  
9743,139.74980063,35.6884957|139.7175283,35.66674267,139.73023124,35.67566771&tolltarge  
t=1
```


レスポンス

●は必須。○は status が success の場合、必須となります。

No.	パラメータ	必須	値	説明
1	routeId		ルート識別子	ルート検索 API 入力パラメータ“ルート識別子(routeid)”で指定した値が格納されます。
2	status	●	ステータス	ルート検索 API の処理結果 正常に完了した場合は“success”を返します。 エラーで終了した場合は“[エラーコード] エラーメッセージ”を返します。
3	summary	○	検索結果サマリー	検索結果サマリーをオブジェクトに格納。
3.1	totalDistance	○	総距離	ルート全体の総距離。単位はメートル。実数で記述。
3.2	totalTravelTime	○	総所要時間	ルート全体の総所要時間。単位は秒。実数で記述。
3.3	totalToll	○	総料金データ	総料金データをオブジェクトに格納。
3.3.1	toll	○	料金	ルート全体の有料道路料金。単位は円。整数で記述。
3.4	totalTollEtc		総料金(ETC 割引)データ	総料金(ETC 割引)データをオブジェクトに格納。 料金計算対象で通常料金+ETC 割引料金を指定した場合に出力されます。
3.4.1	toll	○	料金	ルート全体の有料道路 ETC 割引料金。単位は円。整数で記述。
3.5	departureTime	○	出発日時	出発日時をオブジェクトに格納。
3.5.1	date	○	日付	出発日。年=4 桁、月=2 桁、日=2 桁の合計 8 桁の整数で記述。 例) 2015 年 1 月 24 日=“20150124”

No.	パラメータ	必須	値	説明
3.5.2	time	○	時刻	出発時刻。時=2 桁、分=2 桁、秒=2 桁の整数で記述。なお、ルート検索では秒の設定は無視しており、本出力では秒の値は 0 固定となります。 例) 15:30:00="153000", 00:25:00="002500"
4	guide[]		誘導データ	誘導データを誘導区間単位の配列に格納。 歩行者(priority=100~103)の場合は格納されません。
4.1	type	○	誘導ポイント タイプ	誘導ポイントのタイプを格納。整数で記述。 0: 案内ポイント 1: 出発地 2: 目的地 3: 経由地
4.2	guidePoints[]	○	緯度経度点 列	緯度経度点列を配列に格納。 誘導ポイントタイプが 0 の場合 1 番目:案内ポイントの緯度経度 誘導ポイントタイプが 1,2,3 の場合 1 番目:入力で指定した緯度経度 2 番目:上記地点に一番近い道路上の緯度経度
4.2.1	lon	○	経度	経度。世界測地系(JGD2000)、 degree 形式の実数で記述。
4.2.2	lat	○	緯度	緯度。世界測地系(JGD2000)、 degree 形式の実数で記述。
4.3	guideInfo		誘導情報	誘導情報をオブジェクトに格納。 誘導ポイントタイプが案内ポイントの場合は、本情報が必ず格納されます。

No.	パラメータ	必須	値	説明
4.3.1	guideDirection	○	誘導コード	誘導コード(道路の進行方向)を格納。 整数で記述。 誘導コードは付則.誘導コード一覧を参照
4.3.2	roadType	○	道路種別	誘導区間先頭リンクの道路種別を格納。 整数で記述。 道路種別は付則.道路種別一覧を参照
4.3.3	distance	○	区間距離	区間ごとの距離。単位はメートル。 実数で記述。
4.3.4	travelTime	○	区間所要時間	区間ごとの所要時間。単位は秒。 実数で記述。
4.3.5	guideDetail		詳細誘導データ	誘導案内に必要な詳細情報データのオブジェクトに格納。
4.3.5.1	code	○	誘導詳細コード	誘導詳細コード 32：高速(有料)道路の入口 33：高速(有料)道路の出口 34：SA/PA(目的地または経由地に指定した場合のみ格納) 48：フェリーターミナル
4.3.5.2	name	○	誘導詳細テキスト	誘導詳細テキスト 誘導詳細コードが入口または出口の場合のみ、高速(有料)道路の施設名称が格納されます。
4.3.6	guideHighway		高速道路関連の誘導データ	高速道路関連の誘導データをオブジェクトに格納。
4.3.6.1	facilities[]		有料道路関連の施設情報	有料道路関連の施設情報を、施設単位の配列で格納。

No.	パラメータ	必須	値	説明
4.3.6.1.1	type	○	施設タイプ	施設タイプを格納。 1 : SA. 2 : PA. 3 : ジャンクション 4 : ランプ 5 : IC. 7 : スマート IC
4.3.6.1.2	name	○	施設名称	施設名称を格納。名称がない場合、空文字列が格納されます。
4.3.6.1.3	info	○	施設情報	施設の設備有無をビットフラグで表現した数値を格納。 ビット割り付けは付則.施設情報を参照。
4.3.7	guideCrossing		交差点に関する誘導データ	交差点に関する誘導データをオブジェクトに格納。
4.3.7.1	name	○	交差点名称	交差点名称を格納。名称がない場合、空文字列が格納されます。
4.3.8	guideRoad		道路に関する誘導データ	道路に関する誘導データをオブジェクトに格納。
4.3.8.1	number	○	道路番号	道路番号を格納。整数で記述。道路番号がない場合、0 が格納されます。
4.3.8.2	name	○	道路名称	道路名称を格納。名称がない場合、空文字列が格納されます。
4.3.9	guideToll		料金に関する誘導データ	料金に関する誘導データをオブジェクトに格納。
4.3.9.1	tollGateCode	○	料金所機能コード	料金所機能コードを格納。整数で記述。 1 : 発券機能 2 : 精算機能 3 : 単独料金所 4 : 単独料金所 + 発券機能 5 : 単独料金所 + 精算機能 6 : Uターン用検札機能 7 : 単独料金無効発券機能 8 : 精算機能 + 発券機能

No.	パラメータ	必須	値	説明
4.3.9.2	toll	○	料金	有料道路料金。単位は円。整数で記述。
4.3.9.3	name	○	料金所の施設名称	料金所の施設名称を格納。
4.3.10	guideTollEtc		料金(ETC 割引)に関する誘導データ	料金(ETC 割引)に関する誘導データをオブジェクトに格納。
4.3.10.1	tollGateCode	○	料金所機能コード	料金所機能コードを格納。整数で記述。 1: 発券機能 2: 精算機能 3: 単独料金所 4: 単独料金所＋発券機能 5: 単独料金所＋精算機能 6: Uターン用検札機能 7: 単独料金無効発券機能 8: 精算機能＋発券機能
4.3.10.2	toll	○	料金	有料道路料金。単位は円。整数で記述。
4.3.10.3	name	○	料金所の施設名称	料金所の施設名称を格納。
4.3.10.4	etcCode	○	ETC 施設コード	ETC 施設コード 0: ETC 非対応 1: ゲート 2: アンテナ
5	shape[]	○	ルート形状データ	ルート形状データをリンク単位の配列で格納。
5.1	type	○	形状ポイントタイプ	形状ポイントタイプを格納。整数で記述。 4: 道路形状 5: 区間開始(出発地、経由地) 6: 区間終了(経由地、目的地)

No.	パラメータ	必須	値	説明
5.2	shapeInfo		ルート形状に関する情報	ルート形状に関する情報をオブジェクトに格納。 形状ポイントタイプが道路形状の場合は、本情報が必ず格納されます。
5.2.1	roadType	○	道路種別	誘導区間先頭リンクの道路種別を格納。 整数で記述。 道路種別は付則.道路種別一覧を参照
5.2.2	dataId	○	データ識別子	道路リンクか歩行者リンクかを判別する識別子を格納。 0：道路リンク 1：歩行者リンク
5.2.3	info	○	付加情報	歩行者リンクに関する付加情報をビットフラグで表現した数値を格納。 歩行者リンクでない場合は 0 を格納。 付加情報は付則.歩行者リンク付加情報を参照。
5.3	shapePoints[]	○	緯度経度点列	緯度経度点列をリンク補完点(=最小形状)単位で配列に格納。
5.3.1	lon	○	経度	経度。世界測地系(JGD2000)、degree 形式の実数で記述。
5.3.2	lat	○	緯度	緯度。世界測地系(JGD2000)、degree 形式の実数で記述。

No.	パラメータ	必須	値	説明
6	order		設定地点の 巡回順	<p>巡回ルート設定が有効の場合のみ出力します。</p> <p>リクエストパラメータの出発地、経由地、目的地を格納した入力地点配列のインデックス値が巡回ルート順で格納されます。</p> <p>入力地点配列は“start”、“via”、“destination”の順で格納します。</p> <p>入力地点例：</p> <p>[0] => “start”</p> <p>[1] => “via1”</p> <p>[2] => “via2”</p> <p>[3] => “via3”</p> <p>[4] => “destination”</p> <p>巡回ルート検索結果の order 出力が以下の場合、</p> <p>“order”: [0,3,1,2,4]</p> <p>入力された 0 番目の地点(start) →入力された 3 番目の地点(via3) →入力された 1 番目の地点(via1) →入力された 2 番目の地点(via2) →入力された 4 番目の地点 (destination)</p> <p>上記順番で通るルート検索結果となったことを意味します。</p>

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function({  
  "routeId": "calcroutesample",  
  "status": "success",  
  "summary": {  
    "totalDistance": 46191.456261891,  
    "totalTravelTime": 5938.4292777098,  
    "totalToll": {  
      "toll": 1300  
    },  
    "totalTollEtc": {  
      "toll": 300  
    },  
    "departureTime": {  
      "date": "20140809",  
      "time": "171600"  
    }  
  },  
  "guide": [  
    {  
      "type": 1,  
      "guidePoints": [  
        {  
          "lon": 139.767231,  
          "lat": 35.681196  
        },  
        {  
          "lon": 139.76902982801,  
          "lat": 35.680466103167  
        }  
      ]  
    },  
    {  
      "type": 0,  
      "guidePoints": [  

```



```
{
  "lon": 139.76902982801,
  "lat": 35.680466103167
},
"guideInfo": {
  "guideDirection": 0,
  "roadType": 102,
  "distance": 129.20962524414,
  "travelTime": 5.1683850288391,
  "guideRoad": {
    "number": 0,
    "name": "¥u9996¥u90fd¥u9ad8¥u901f¥u516b¥u91cd¥u6d32¥u7dda"
  }
},
...
{
  "type": 2,
  "guidePoints": [
    {
      "lon": 139.700494,
      "lat": 35.689957
    },
    {
      "lon": 139.70123902399,
      "lat": 35.690083033846
    }
  ]
},
], ],
"shape": [
  {
    "type": 5,
    "shapeInfo": {
      "roadType": 102,
```

```
        "dataId": 0,
        "info": 0
    },
    "shapePoints": [
        {
            "lon": 139.76902982801,
            "lat": 35.680466103167
        },
        {
            "lon": 139.76966959635,
            "lat": 35.681506415473
        }
    ]
},
...
{
    "type": 6,
    "shapeInfo": {
        "roadType": 9,
        "dataId": 0,
        "info": 0
    },
    "shapePoints": [
        {
            "lon": 139.70146321615,
            "lat": 35.690725708008
        },
        ...
    ]
}
], ],
})
```

サンプルレスポンス（正常時:XML 返却）

```
<calcrouteResponse>
  <routeId>calcroutesample</routeId>
  <status>success</status>
  <summary>
    <totalDistance>46191.456261891</totalDistance>
    <totalTravelTime>5938.4292777098</totalTravelTime>
    <totalToll>
      <toll>1300</toll>
    </totalToll>
    <totalTollEtc>
      <toll>300</toll>
    </totalTollEtc>
    <departureTime>
      <date>20140809</date>
    </departureTime>
  </summary>
  <guide>
    <type>1</type>
    <guidePoints>
      <lon>139.767231</lon>
      <lat>35.681196</lat>
    </guidePoints>
    <guidePoints>
      <lon>139.76902982801</lon>
      <lat>35.680466103167</lat>
    </guidePoints>
  </guide>
  <guide>
    <type>0</type>
    <guidePoints>
      <lon>139.76902982801</lon>
      <lat>35.680466103167</lat>
    </guidePoints>
    <guideInfo>
```

```

    <guideDirection>0</guideDirection>
    <roadType>102</roadType>
    <distance>129.20962524414</distance>
    <travelTime>5.1683850288391</travelTime>
    <guideRoad>
      <number>0</number>
      <name>首都高速八重洲線</name>
    </guideRoad>
  </guideInfo>
</guide>
...
<guide>
  <type>2</type>
  <guidePoints>
    <lon>139.700494</lon>
    <lat>35.689957</lat>
  </guidePoints>
  <guidePoints>
    <lon>139.70123902399</lon>
    <lat>35.690083033846</lat>
  </guidePoints>
</guide>
<shape>
  <type>5</type>
  <shapeInfo>
    <roadType>102</roadType>
    <dataId>0</dataId>
    <info>0</info>
  </shapeInfo>
  <shapePoints>
    <lon>139.76902982801</lon>
    <lat>35.680466103167</lat>
  </shapePoints>
  <shapePoints>
    <lon>139.76966959635</lon>
    <lat>35.681506415473</lat>

```

```
</shapePoints>
</shape>
<shape>
  <type>6</type>
  <shapeInfo>
    <roadType>9</roadType>
    <dataId>0</dataId>
    <info>0</info>
  </shapeInfo>
  <shapePoints>
    <lon>139.70146321615</lon>
    <lat>35.690725708008</lat>
  </shapePoints>
  ...
  <shapePoints>
    <lon>139.70123902399</lon>
    <lat>35.690083033846</lat>
  </shapePoints>
</shape>
</calcrouteResponse>
```

サンプルレスポンス（エラー時）

- ・出発地緯度経度(start)が未設定の場合

```
jsonp_callback_function (  
  {"status": "[E03001] calcroute parameter error (start is required)"}  
)
```

エラーコード一覧は付則を参照

補足

ルート検索 API を実行する際、距離の長さや検索条件によってはレスポンスが返却されるまで時間がかかる場合があります。

ブラウザが IE の場合、URL 長制限のため探索条件によってはリクエストできない場合がございます。

巡回ルート設定を有効にし、設定した地点が同一道路上で近距離の場合、エラーコード「I00101」エラーメッセージ「too short between starting point and destination or via point」が返却されることがあります。その際は離れた地点を設定してください。

設定された出発地、経由地、目的地間の直線距離の合計が 2,000km を超える場合は、エラーコード「I00102」エラーメッセージ「too long between starting point and destination or via point」が返却されます。

但し、巡回ルート設定が有効の場合は 500km、歩行者ルート(priority=100~103)の場合は 5km を超える際に同様のエラーが返却されます。

レスポンスで返される料金につきましては、フェリーの料金は含まれません。

ETC 割引料金につきましては、検索時に実施されている ETC 割引を考慮した料金となります。料金につきましては、出発日時(指定なしの場合 API 実行日時)の曜日・時間に該当する割引での計算となります。(祝日は考慮されません)

制限回数のある割引につきましては検索内で利用回数がカウントされ、制限回数に従った計算となります。

2.6.ユーティリティ

2.6.1. 緯度経度測地系変換 : transform

緯度経度情報の測地系および表記形式を変換する機能を提供します。

リクエスト

テスト利用 URL	http(s)://api-srch-pre.mapfan.com/v1/transform?key={key}&fmt={fmt}&callback={callback}&lonlat={lonlat}&type_from={type_from}&type_to={type_to}&datum_from={datum_from}&datum_to={datum_to}		
本番利用 URL	http(s)://api-srch.mapfan.com/v1/transform?key={key}&fmt={fmt}&callback={callback}&lonlat={lonlat}&type_from={type_from}&type_to={type_to}&datum_from={datum_from}&datum_to={datum_to}		
パラメータ	型	必須/任意	説明
Key	string	必須	アクセスに使用するキー ※認証 API より取得
fmt	string	任意	出力フォーマット JSON 出力の場合は json (デフォルト) XML 出力の場合は xml なお、コールバック用関数名のパラメータが設定されている場合、本パラメータの設定が破棄され、JSONP 形式が適用されます。
callback	string	任意	JSONP のコールバック用関数名 関数名が指定されている場合は JSONP 形式で結果を返却します。 指定されていない場合は、fmt パラメータで指定された形式で結果を返却します。

パラメータ	型	必須/任意	説明
lonlat	String	必須	<p>緯度経度</p> <p>■DEGREE 形式 (MapFan API 標準) の場合 記述形式: [経度の値]+","+[緯度の値] 記述例: "139.767231,35.681196"</p> <p>■DMS10 形式 (MapFan onPage 標準) の場合 記述形式: [経度 1/10 秒精度 DMS 形式]+ [緯度 1/10 秒精度 DMS 形式] 1/10 秒精度 DMS 形式: [方角]+[度]+". "+[分]+". "+[秒]+". " +[秒の小数点第一位] 方角は"E"(経度)、"N"(緯度)の文字列。 他は整数値。 記述例: "E139.45.28.1N35.34.13.9"</p> <p>■範囲 経度は E(東経)のみ。 緯度は N(北緯)のみ。 最小: 経度=120, 緯度=20 最大: 経度=155, 緯度=46</p>
type_from	String	任意(*1)	<p>入力表記形式 変換前の表記形式を指定します。 設定値: "DEGREE"=度単位の実数形式 (MapFan API の標準形式) (デフォルト) 設定値: "DMS10"=1/10 秒精度の度分秒 (MapFan onPage 標準形式)</p>
type_to	String	任意(*1)	<p>出力表記形式 変換後の表記形式を指定します。 設定値は、type_from と同様。</p>

パラメータ	型	必須/任意	説明
datum_from	String	任意(*2)	入力測地系 変換前の測地系を指定します。 設定値: "JGD" = 世界測地系(JGD2000) (MapFan API の標準形式) 設定値: "TKY" = 旧日本測地系(東京測地系) (MapFan onPage 標準形式)
datum_to	String	任意(*2)	出力測地系 変換後の測地系を指定します。 設定値は、datum_from と同様。

- (*1) 入力表記形式及び出力表記形式はどちらかのみ指定した場合はエラーとなります。
なお表記形式変換を行わず、かつ表記形式が DEGREE の場合は両方のパラメータを指定しなくても良いです。
- (*2) 入力測地系及び出力測地系はどちらかのみ指定した場合はエラーとなります。
なお測地系変換を行わない場合は両方のパラメータを指定しなくても良いです。

サンプルリクエスト

```
http://api-
srch.mapfan.com/v1/transform?key=xxxxx&fmt=jsonp&callback=jsonp_callback_function&lonlat=
E139.42.24.0N35.41.26.0&type_from=DMS10&type_to=DEGREE&datum_from=TKY&datum_to=J
GD
```

レスポンス

パラメータ	値	説明
status	ステータス	緯度経度測地系変換 API の処理結果 正常に完了した場合は"success"を返します。 エラーで終了した場合は"[エラーコード] エラーメッ セージ"を返します。
lonlat	緯度経度	変換した緯度経度

サンプルレスポンス（正常時:JSONP 返却）

```
jsonp_callback_function(  
  {  
    "status": "success",  
    "lonlat": "139.703437919663, 35.693789947933"  
  }  
)
```

サンプルレスポンス（正常時:XML 返却）

```
<transformResponse>  
  <status>success</status>  
  <lonlat>139.703437919663,35.693789947933</lonlat>  
</transformResponse>
```

サンプルレスポンス（エラー時）

- ・ 緯度経度(lonlat)が未設定の場合

```
jsonp_callback_function (  
  {"status": "[E03001] transform parameter error (lonlat is required)"}  
)
```

エラーコード一覧は付則を参照

付則

サーバーホスト一覧

分類	API 名	テスト環境ホスト	本番環境ホスト
認証	認証	api-auth-pre.mapfan.com	api-auth.mapfan.com
地図	地図画像取得	api-map-pre.mapfan.com	api-map.mapfan.com
	サイズ指定地図画像取得		
検索	住所検索	api-srch-pre.mapfan.com	api-srch.mapfan.com
	住所階層検索		
	住所逆引き検索		
	スポット検索		
	スポット検索(タウンページデータ検索オプション)		
	スポット周辺検索		
	スポット周辺検索(タウンページデータ検索オプション)		
	駅検索		
	駅階層検索		
	最寄駅検索		
	郵便番号検索		
	多言語スポット周辺検索		
ルート	ルート検索	api-route-pre.mapfan.com	api-route.mapfan.com
ユーティリティ	緯度経度測地系変換	api-srch-pre.mapfan.com	api-srch.mapfan.com

エラーコード一覧

出力文字列: xxxxx=API 名, *****=パラメータ名 API 毎の対応: ○=エラー判定対象

認証 API

コード	返却文字列	内容	auth
E01001	api unexpected error (xxxxx)	API 実行エラー	○
E01003	system unexpected error (xxxxx)	システム内部の予期せぬエラー	○
E01004	system unexpected error (accesskey creation failed)	アクセスキー生成失敗によるエラー	○
E02004	authentication error (appid is invalid)	認証 ID が壊れているなど、有効でない場合に発生	○
E03001	xxxxx parameter error (***** is required)	必須パラメータが未入力	○
E03002	xxxxx parameter error (***** is invalid)	パラメータの値が有効でない場合に発生	○
E03003	xxxxx parameter error (parameter is too long)	リクエスト URL の長さが長すぎる場合に発生	○

地図 API

コード	返却文字列	内容	map	mapimage
E01001	api unexpected error (xxxxx)	API 実行エラー	○	○
E01002	system connect error (xxxxx)	システム内部接続エラー	○	○
E01003	system unexpected error (xxxxx)	システム内部の予期せぬエラー	○	○
E02001	authentication error (service has expired)	アクセスキー有効期限切れ	○	○
E02002	authentication error (unable to use a function)	契約範囲外の機能を利用しようとした場合に発生	○	○
E02003	authentication error (key is invalid)	アクセスキーが壊れているなど、有効でない場合に発生	○	○
E03001	xxxxx parameter error (***** is required)	必須パラメータが未入力	○	○
E03002	xxxxx parameter error (***** is invalid)	パラメータの値が有効でない場合に発生	○	○
E03003	xxxxx parameter error (parameter is too long)	リクエスト URL の長さが長すぎる場合に発生	○	○
E03004	xxxxx parameter error (invalid combination)	地図デザインの組み合わせ指定が有効でない場合に発生	○	○

ルート API

コード	返却文字列	内容	calcroute
I00101	too short between starting point and destination or via point	該当するルートがない: 出発地と目的地(経由地)が近すぎるため	○
I00102	too long between starting point and destination or via point	該当するルートがない: 出発地と目的地(経由地)が遠すぎるため	○
I00103	not found route	該当するルートがない: その他の理由	○
I00104	no road link near starting point	該当するルートがない: 出発地付近に道路がないため	○
I00105	no road link near destination or waypoint	該当するルートがない: 目的地または経由地付近に道路がないため	○
I00106	cause of solitary island	該当するルートがない: 離島が原因でルートが見つからないため	○
I00107	no route without ferry route	該当するルートがない: フェリー航路を使うルートしか見つからないため	○
I00108	no route without toll way	該当するルートがない: 有料道路を使うルートしか見つからないため	○

コード	返却文字列	内容	calcroute
E01001	api unexpected error (xxxxx)	API 実行エラー	○
E01002	system connect error (xxxxx)	システム内部接続エラー	○
E01003	system unexpected error (xxxxx)	システム内部の予期せぬエラー	○
E02001	authentication error (service has expired)	アクセスキー有効期限切れ	○
E02002	authentication error (unable to use a function)	契約範囲外の機能を利用しようとした場合に発生	○
E02003	authentication error (key is invalid)	アクセスキーが壊れているなど、有効でない場合に発生	○
E03001	xxxxx parameter error (***** is required)	必須パラメータが未入力	○
E03002	xxxxx parameter error (***** is invalid)	パラメータの値が有効でない場合に発生	○
E03003	xxxxx parameter error (parameter is too long)	リクエスト URL の長さが長すぎる場合に発生	○
E03004	xxxxx parameter error (invalid combination)	検索条件の組み合わせ指定が有効でない場合に発生	○

検索 API

コード	返却文字列	内容	addr	addrlist	addrname	spot/ spotopt	spotarea/ spotareaopt	sta	stalist	staarea	zip	spotareaml
I00001	no hit	検索結果に該当箇所がない	○	○	○	○	○	○	○	○	○	○
E01001	api unexpected error (xxxxx)	API 実行エラー	○	○	○	○	○	○	○	○	○	○
E01002	system connect error (xxxxx)	システム内部接続エラー	○	○	○	○	○	○	○	○	○	○
E01003	system unexpected error (xxxxx)	システム内部の予期せぬエラー	○	○	○	○	○	○	○	○	○	○
E02001	authentication error (service has expired)	アクセスキー有効期限切れ	○	○	○	○	○	○	○	○	○	○
E02002	authentication error (unable to use a function)	契約範囲外の機能を利用しようとした場合に発生	○	○	○	○	○	○	○	○	○	○
E02003	authentication error (key is invalid)	アクセスキーが壊れているなど、有効でない場合に発生	○	○	○	○	○	○	○	○	○	○

コード	返却文字列	内容	addr	addrlist	addrname	spot/ spotopt	spotarea/ spotareaopt	sta	stalist	staarea	zip	spotareaml
E03001	xxxxx parameter error (***** is required)	必須パラメータが 未入力	○	○	○	○	○	○	○	○	○	○
E03002	xxxxx parameter error (***** is invalid)	パラメータの値が 有効でない場合 に発生	○	○	○	○	○	○	○	○	○	○
E03003	xxxxx parameter error (parameter is too long)	リクエスト URL の 長さが長すぎる 場合に発生	○	○	○	○	○	○	○	○	○	○
E03005	xxxxx parameter combinate error (***** is required)	条件絞り込み時 の必須パラメータ 未指定時に発生	-	○	-	○	○	-	○	-	-	-

コード	返却文字列	内容	addr	addrlist	addrname	spot/ spotopt	spotarea/ spotareaopt	sta	stalist	staarea	zip	spotareaml
E03006	xxxxx parameter combinate error (***** is invalid)	条件絞り込み時 のパラメータの値 が有効でない場 合に発生	-	○	-	-	-	-	○	-	-	-

ユーティリティ API

コード	返却文字列	内容	transform
E01001	api unexpected error (xxxxx)	API 実行エラー	○
E01002	system connect error (xxxxx)	システム内部接続エラー	○
E01003	system unexpected error (xxxxx)	システム内部の予期せぬエラー	○
E02001	authentication error (service has expired)	アクセスキー有効期限切れ	○
E02002	authentication error (unable to use a function)	契約範囲外の機能を利用しようとした場合に発生	○
E02003	authentication error (key is invalid)	アクセスキーが壊れているなど、有効でない場合に発生	○
E03001	xxxxx parameter error (***** is required)	必須パラメータが未入力	○
E03002	xxxxx parameter error (***** is invalid)	パラメータの値が有効でない場合に発生	○
E03003	xxxxx parameter error (parameter is too long)	リクエスト URL の長さが長すぎる場合に発生	○
E03005	xxxxx parameter combinate error (***** is required)	条件絞り込み時の必須パラメータ未指定時に発生	○

地図画像取得

アイコン種別一覧

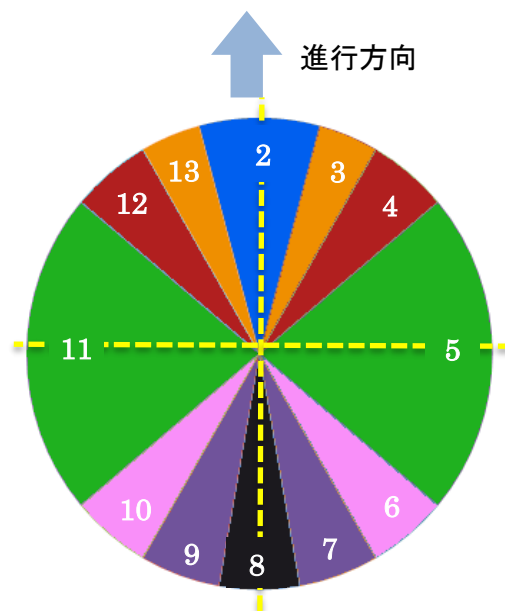
コード	タイプ名	デザイン
1	標準サイズ 赤色アイコン	
2	標準サイズ 青色アイコン	
3	標準サイズ 緑色アイコン	
4	標準サイズ 黄色アイコン	
5	標準サイズ 白色アイコン	
6	標準サイズ 黒色アイコン	
101	大きいサイズ 赤色アイコン	
102	大きいサイズ 青色アイコン	

コード	タイプ名	デザイン
103	大きいサイズ 緑色アイコン	
104	大きいサイズ 黄色アイコン	
105	大きいサイズ 白色アイコン	
106	大きいサイズ 黒色アイコン	

ルート機能

誘導コード一覧

コード	内容
0	不明
1	道なり
2	直進
3	右斜め
4	右斜め
5	右
6	右斜め後ろ
7	右斜め後ろ
8	Uターン
9	左斜め後ろ
10	左斜め後ろ
11	左
12	左斜め
13	左斜め



経路種別一覧

ルート検索の走行速度条件で利用します。

コード	内容	走行速度[km/h]	分類
0	ルート検索対象外道路	—	—
1	一般国道	35	一般道
2	主要地方道(都府県)	30	一般道
3	主要地方道(指定市)	30	一般道
4	一般都道府県道	30	一般道
5	指定市の一般市道	25	一般道
6	その他道路	25	一般道
101	高速自動車道路(有料)	75	高速道
102	都市高速道路(有料)	50	高速道
103	一般国道(有料)	50	有料道
104	主要地方道(都府県)(有料)	50	有料道
105	主要地方道(指定市)(有料)	50	有料道
106	一般都道府県道(有料)	50	有料道
107	指定市の一般市道(有料)	40	有料道
108	その他道路(有料)	40	有料道
201	フェリー航路	27	—
403	接続国道※	30	一般道

※走行速度はルートの所要時間の計算に使用します。ただし、一般道の走行速度、高速道の走行速度、有料道の走行速度を指定した場合は、各分類に指定した速度が適用されます。

道路種別一覧

ルート検索結果の各誘導区間の道路種別を示します。

・道路リンク

道路の道塗は以下の通り分類します。

コード	道塗りカテゴリ
0-99	無料一般道、無料高速道
100-199	有料一般道、有料高速道
200-299	フェリー
300-399	その他一般道

・歩行者リンク

コード	道路種別説明
0	設定なし
1	歩道
2	みなし横断歩道
3	横断歩道
99	その他歩道
100	人車共用リンク
101	接続リンク
200	歩道橋
201	橋
202	踏切
300	施設敷地内(屋外)
301	施設敷地内(屋内)
400	地下通路
401	横断地下道
500	歩行者専用フェリー

規制車種一覧

コード	施設
0	指定なし(デフォルト)
1	大型乗用自動車
6	大型貨物自動車
11	大型特殊自動車

施設情報

施設の設備有無のビット情報

bit	施設
7-0	(リザーブ)
8	トイレ
9	身障者用トイレ
10	レストラン
11	軽食
12	売店
13	休憩所
14	仮眠休憩所
15	対人案内所
16	インフォメーション
17	シャワー施設
18	コインランドリー
19	公衆浴場
20	FAX
21	郵便ポスト
22	キャッシュディスペンサーサービス
23	ハイウェイオアシス
24	コイン洗車場
25	ガソリンスタンド

コード	施設の有無
0	存在しない
1	存在する

歩行者リンク付加情報

歩行者リンクのビット情報

bit	属性
0	オートウォーク
1	階段
2	スロープ
3	エスカレータ
4	屋根付き
5	トンネル
6	広場
7	エレベータ
11-8	(リザーブ)
15-12	通行禁止種別
19-16	一方通行種別

コード	説明
0	でない
1	である

通行禁止種別

値	説明
0	通行禁止なし
1	条件なし通行禁止
2	条件付通行禁止
3	通行不可
4	条件なし通行禁止(関係者以外)
5	条件付通行禁止(関係者以外)

一方通行種別

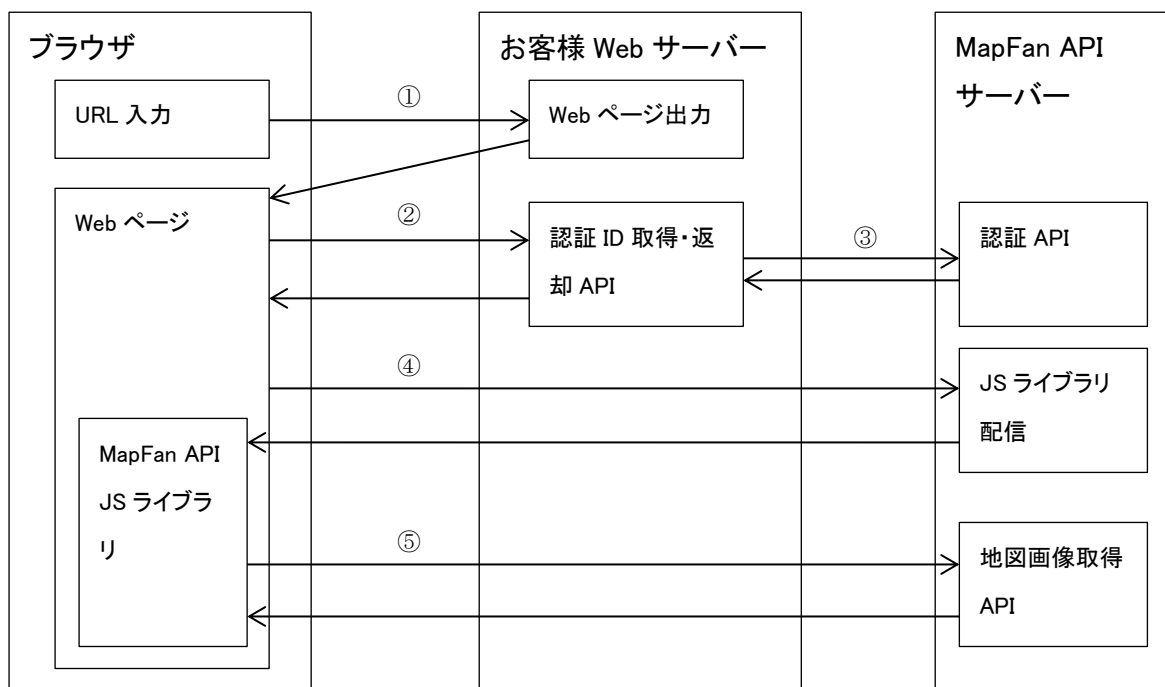
値	説明
0	一方通行なし
1	条件なし一方通行(正方向)
2	条件なし一方通行(逆方向)
3	条件付一方通行(正方向)

値	説明
4	条件付一方通行(逆方向)
5	条件付一方通行(正逆あり)

システムの推奨構成

JavaScript を用いた Web ページ等で MapFan API をご利用いただく場合は、弊社より提供します「認証 ID」が HTML ソースから第三者に参照されることのないシステム構成を推奨します。

システム構成例 (MapFan API の JS ライブラリを利用して地図を表示する場合)



①ブラウザからの Web ページ要求によりお客様ページを作成、出力します。

②認証 ID 取得・返却 API (例) をお客様側でご用意いただき、Ajax 等で取得要求を受け、③の結果で得られるアクセスキーを返却します。

③お客様の認証 ID をパラメータとして、MapFan API の認証 API を要求していただき、発行されるアクセスキーを受信してください。

※受信したアクセスキーをお客様のサーバーで保管し再度使用する事は、MapFan API の利用規約上、禁止とさせていただいております。

④MapFan API の JS ライブラリを読み込みます。

⑤②でブラウザに渡されたアクセスキーをパラメータとして、MapFan API の地図画像取得 API を要求してください。

PV の定義

MapFan API は、以下の条件で PV(ページビュー)をカウントします。

地図表示(MapFan API JavaScript ライブラリをご利用の場合)

1 回の地図ページ表示で1カウントされます。

表示された地図ページ内で地図スクロールや縮尺変更処理を行ってもカウントはされません。

地図表示(MapFan API JavaScript ライブラリを利用されない場合)

MapFan API への認証(アクセスキー発行)で1カウントされます。

ただし、認証のみ行い、地図が一枚もアクセスされない場合はカウントされません。

地図表示(サイズ指定地図画像取得 API をご利用の場合)

サイズ指定地図画像取得 API1 回実行につき、1カウントされます。

検索

各種検索 API1 回実行につき、1カウントされます。

ルート

ルート検索 API1 回実行につき、1カウントされます。

VICS

VICS 渋滞情報 API に関しては、PV としてカウントされません。

ユーティリティ

緯度経度測地系変換 API1 回実行につき、1カウントされます。

改版履歴

版数	発行日	改訂履歴
1.0.0	2014/10/16	・初版発行
1.0.1	2015/01/13	・付則「システムの推奨構成」を追加 ・付則「PV の定義」を追加
1.1.0	2015/02/03	・2.4.3.住所逆引き検索の項目を追加 ・2.5 ルート検索の項目を追加 ・付則.サーバーホスト一覧の項目を追加 ・付則.エラーコード一覧にルート検索で使用するコードを追加 ・付則.誘導コード一覧、道路種別一覧、施設情報の項目を追加 ・付則.PV の定義にルート検索 API の項目を追加
1.2.0	2015/05/12	・2.3.1. 地図画像取得 API のスケール最大値を更新 ・2.3.2.サイズ指定地図画像取得 API の項目を追加 ・2.4.検索にスポット検索 API、スポット検索(タウンページデータ検索オプション)API、スポット周辺検索 API、スポット周辺検索(タウンページデータ検索オプション)API、駅検索 API、駅階層検索 API、最寄駅検索 API、郵便番号検索 API を追加 ・2.6.ユーティリティを追加 ・付則 エラーコード一覧に新規 API のエラーコードを追加 ・PV の定義 新規 API に関する PV の定義を追加
1.2.1	2015/06/22	・2.4.1. 住所検索 レスポンスの level について誤記修正
1.3.0	2015/09/17	・2.4.4. リクエストパラメータに市区町村コード・検索フィールドを追加 ・2.4.5. リクエストパラメータに市区町村コード・検索フィールドを追加 ・2.4.6. リクエストパラメータに検索フィールドを追加 ・2.4.7. リクエストパラメータに検索フィールドを追加 ・付則.エラーコード一覧に検索関連 API のエラーコードを一部修正
1.4.0	2015/12/03	・2.5.1.リクエストパラメータに車種、車高、車重、車幅、危険物積載車両条件追加 ・付則 経路種別一覧、道路種別一覧、規制車種一覧を追加
1.5.0	2016/05/18	・2.5.1. リクエストパラメータに日時規制、一般道規制、高速道規制、巡回ルート設定条件追加 ・2.5.1. レスポンスパラメータに設定地点の巡回順追加、出発日時を数値出力から文字列出力に変更

版数	発行日	改訂履歴
1.6.0	2016/09/06	<ul style="list-style-type: none"> ・2.3.1. リクエストパラメータに解像度パラメータ追加 ・2.3.2. リクエストパラメータに解像度パラメータ、アイコン設定、スケール表示、中心点アイコン表示追加 ・2.4.1. レスポンスパラメータに都道府県コード、市区町村コード、条町丁目コード、番地コード、号コード追加 ・2.4.11. レスポンスパラメータに都道府県コード、市区町村コード、条町丁目コード追加 ・2.4.12. 多言語スポット周辺検索: spotareaml を追加 ・2.5.1. リクエストパラメータの出発地緯度経度、経由地緯度経度、目的地緯度経度に地点種別の記載を追加 ・2.5.1. リクエストパラメータに通行可能エリア、通行不可エリア追加
1.7.0	2016/11/01	<ul style="list-style-type: none"> ・リファレンスに関するご質問のユーザサポート窓口宛先を修正
2.0.0	2017/04/05	<p>2.3.1 地図画像取得 API</p> <ul style="list-style-type: none"> ・リクエストに地図の表示内容、ランドマークアイコン、角度パラメータを追加、及び解像度パラメータに 4 の値を追加 <p>2.3.2 サイズ指定地図画像取得</p> <ul style="list-style-type: none"> ・リクエストに地図の表示内容、ランドマークアイコン、角度パラメータを追加 <p>2.4.2 住所階層検索</p> <ul style="list-style-type: none"> ・リクエストの政令指定都市表示有無設定の記載不備を修正 <p>2.4.3 住所逆引き検索</p> <ul style="list-style-type: none"> ・レスポンスに郵便番号を追加 <p>2.4.8 駅検索、2.2.9 駅階層検索、2.4.10 最寄駅検索</p> <ul style="list-style-type: none"> ・駅出入口検索機能を追加 <p>2.5 ルート機能</p> <ul style="list-style-type: none"> ・ETC 割引料金の検索機能を追加
2.0.1	2017/06/12	<p>2.4.8. 駅検索、2.4.11 郵便番号検索のリクエスト num パラメータの説明を修正</p>
2.1.0	2018/01/10	<p>2.5 ルート機能</p> <ul style="list-style-type: none"> ・歩行者ルートの検索機能を追加 <p>付則に歩行者リンク付加情報の項目を追加、及び道路種別一覧に歩行者リンクにおけるコード一覧を追加</p>