ACT 距離計算サービス
サンプルプログラム（Android 版）
解説書

アドバンスド・コア・テクノロジー株式会社

ACT 距離計算サービスサンプルプログラム（Android 版）
解説書
２０１２年　７月１３日　初版作成
２０１２年　７月２４日　第２版
２０１２年　７月２５日　第３版
２０１２年１２月１３日　第４版

Android は Google Inc.の商標または登録商標です。

目次

1. 開発環境・動作環境 1
2. 注意事項 1
3. ファイル一覧 1
3.1. ソースコード 1
3.2. ライブラリファイル 6
3.3. リソースファイル 6
4. ACT 距離計算サービスの利用 9
4.1. サンプルプログラムの概要 9
4.2. 呼び出し手順 11
4.3. 管理サービス 13
4.3.1. AdminService.getUserInformation2 13
4.3.2. AdminService.getUsableNWInfos 15
4.3.3. AdminService.getUsableDCServer 17
4.3.4. AdminService.getUsableGeoServer 19
4.4. 距離計算サービス 21
4.4.1. DCService.calcRoute 21
4.4.2. DCService.calcOptRoute 27
4.4.3. DCService.calcStraightLine 28
4.4.4. DCService.calcArea3 30
4.4.5. DCService.calcOneWay3 35
4.4.6. DCService.getNearestNode 39
4.4.7. DCService.enumNearNode 42
4.4.8. DCService.getNearestLink2 45
4.4.9. DCService.getSpeed2 47
4.4.10. DCService.setSpeed2 49
4.4.11. DCService.getNetworkInfo 51
4.4.12. DCService.getDCWSMessage 53
4.5. 住所検索サービス 54
4.5.1. GCService.searchAddress 54
4.5.2. GCService.searchAddressCode 57
4.5.3. GCService.countLowerAddressByCode 60
4.5.4. GCService.enumLowerAddressByCode 62
4.5.5. GCService.createFullAddressByCode 65

4.5.6. GCService.getNearestAddress 68
4.5.7. GCService.searchZip 71
4.5.8. GCService.countAddressByZip.....74
4.5.9. GCService.enumAddressByZip 76
4.5.10. GCService.getGCWSMessage.....78

1. 開発環境・動作環境

このプログラムは JDK6、Eclipse3.8 で作成されています。
動作環境は Android2.2 以降です。
動作するには Google Play サービスがインストールされている必要があります。
動作確認は Xperia GX SO-04D と ASUS Pad TF300T で行っています。

Eclipse にソースコードをインポートするには、メニューの「ファイル」－「インポート」を選択してください。インポートソースは「一般」－「既存プロジェクトをワークスペースへ」を選択して「次へ」ボタンを押下します。「ルートディレクトリ」にソースコードを解凍したフォルダを指定すると、プロジェクトリストに ActServiceSample が表示されます。チェックをして「完了」ボタンを押下するとインポートされます。

2. 注意事項

このプログラムは地図の表示に Google Maps Android API v2 を使用しています。
Google Play services SDK をプロジェクトで参照するように設定してください（詳しくは https://developers.google.com/maps/documentation/android/を参照してください）。
地図を表示するには Android Maps API Key が必要です。
取得したキーを「¥Sample¥Source¥ActServiceSample¥AndroidManifest.xml」ファイルに設定すると地図が表示されます。
データ量の多い計算（詳細ルート表示設定で青森―鹿児島間を何往復もするなど）を行うと端末によってメモリ不足でアプリケーションが終了する可能性があります。

3. ファイル一覧

3.1. ソースコード

① アプリケーションマニフェスト

パス： ¥Sample¥Source¥ActServiceSample

項番	名前	内容
1	AndroidManifest.xml	サンプルプログラムのマニフェストです。

② ACT 距離計算サービス管理サービス関連

パス： ¥Sample¥Source¥ActServiceSample¥src¥jp¥co¥act_inc¥actservicesample¥adminservice

項番	名前	内容
1	AdminService.java	ACT 距離計算サービスの管理サービ

		スを利用します。
2	Constants.java	定数です。
3	Environ.java	環境設定です。
4	GetUsableDCServerResponse.java	使用可能距離計算サーバー取得の結果です。
5	GetUsableGeoServerResponse.java	使用可能住所検索サーバー取得の結果です。
6	GetUsableNWInfosResponse.java	使用可能計算用道路データ取得の結果です。
7	GetUserInformation2Response.java	ユーザー情報取得の結果です。
8	NWInfo.java	計算用道路データ情報です。
9	Response.java	計算結果インターフェースです。
10	ServiceUtils.java	ACT 距離計算サービスのユーティリティです。
11	SoapException.java	SOAP 例外です。
12	UserInformation2.java	ユーザー情報です。
13	WebException.java	Web 例外です。

③ ACT 距離計算サービス距離計算サービス関連

パス： ¥Sample¥Source¥ActServiceSample¥src¥jp¥co¥act_inc¥actservicesample¥dcservice

項番	名前	内容
1	CA3Param.java	到達圏／流入圏計算用パラメータです。
2	CalcArea3Response.java	到達圏／流入圏計算の結果です。
3	CalcOneWay3Response.java	片道一括計算の結果です。
4	CalcOptRouteResponse.java	最短ルート計算の結果です。
5	CalcResultHolder.java	距離計算結果を格納します。
6	CalcRouteResponse.java	ルート計算の結果です。
7	CalcStraightLineResponse.java	直線距離計算の結果です。
8	COW3Param.java	片道一括計算パラメータです。
9	CRParam.java	ルート計算パラメータです。
10	CLSParam.java	直線距離計算パラメータです。
11	DCService.java	ACT 距離計算サービスの距離計算サ

		ービスを利用します。
12	DistUtils.java	距離計算サービス用ユーティリティです。
13	ENNParam.java	最寄ノード列挙パラメータです。
14	EnumNearNodeResponse.java	最寄ノード列挙の結果です。
15	GetDcwsMessageResponse.java	距離計算用エラーメッセージ取得の結果です。
16	GetNearestLink2Response.java	最寄リンク取得 2 の結果です。
17	GetNearestNodeResponse.java	最寄ノード取得の結果です。
18	GetNetworkInfoResponse.java	計算用道路データ情報取得の結果です。
19	GetSpeed2Response.java	道路速度取得の結果です。
20	GNLParam.java	最寄リンク取得パラメータです。
21	GNNParam.java	最寄ノード取得パラメータです。
22	GSParam.java	道路速度取得パラメータです。
23	Loc.java	地点です。
24	NodePnt.java	ノード付ポイントです。
25	Pnt.java	ポイントです。
26	Rn.java	経由道路です。
27	SetSpeed2Response.java	道路速度設定の結果です。
28	SSParam.java	道路速度設定パラメータです。
29	Trd.java	経由有料道路です。

④　ACT 距離計算サービス住所検索サービス関連

パス：　¥Sample¥Source¥ActServiceSample¥src¥jp¥co¥act_inc¥actservicesample¥gcservice

項番	名前	内容
1	AddressItem.java	住所アイテムです。
2	CommonParam.java	住所検索用共通パラメータです。
3	CountAddressByZipResponse.java	郵便番号対応住所数取得の結果です。
4	CountLowerAddressByCodeResponse.java	下位住所数取得の結果です。
5	CreateFullAddressByCodeItem.java	全住所取得アイテムです。
6	CreateFullAddressByCodeRespons	全住所取得の結果です。

	e.java	
7	EnumAddressByZipResponse.java	郵便番号対応住所列挙の結果です。
8	EnumLowerAddressByCodeRespon se.java	下位住所列挙の結果です。
9	GCService.java	ACT 距離計算サービスの住所検索サービスを利用します。
10	GeoInfo.java	住所データ情報です。
11	GeoResultHolder.java	住所検索結果を格納します。
12	GeoUtils.java	住所検索用ユーティリティです。
13	GetGcwsMessageResponse.java	住所検索用エラーメッセージ取得の結果です。
14	GetNearestAddressItem.java	最寄住所取得アイテムです。
15	GetNearestAddressResponse.java	最寄住所取得の結果です。
16	SearchAddressCodeItem.java	住所コード検索アイテムです。
17	SearchAddressCodeResponse.java	住所コード検索の結果です。
18	SearchAddressItem.java	住所検索アイテムです。
19	SearchAddressResponse.java	住所検索の結果です。
20	SearchZipItem.java	郵便番号検索アイテムです。
21	SearchZipResponse.java	郵便番号検索の結果です。
22	ZipItem.java	郵便番号アイテムです。

⑤　アクティビティとフラグメント関連

パス：　¥Sample¥Source¥ActServiceSample¥src¥jp¥co¥act_inc¥actservicesample¥activity

項番	名前	内容
1	ActMapActivity.java	メイン画面です。
2	ActMapFragment.java	メイン画面です。
3	AddressSearchActivity.java	住所検索画面です。
4	AddressSearchFragment.java	住所検索画面です。
5	CalcArea3Activity.java	到達圏／流入圏計算設定画面です。
6	CalcOneWay3Activity.java	片道一括計算設定画面です。
7	CalcRouteActivity.java	ルート計算画面です。
8	EditDialogFragment.java	エディットダイアログです。
9	InformationDialogFragment.java	メッセージダイアログです。
10	LocActivity.java	地点一覧画面です。

11	LocFragment.java	地点一覧画面です。
12	MainActivity.java	エントリーポイントです。
13	MenuDialogFragment.java	メニューダイアログです。
14	ProgressDialogFragment.java	進捗ダイアログです。
15	QuestionDialogFragment.java	選択ダイアログです。
16	ResultActivity.java	計算結果画面です。
17	RoadSelectionActivity.java	計算用道路データ選択画面です。
18	RoadSelectionFragment.java	計算用道路データ選択画面です。
19	ServicePreferenceActivity.java	サービス設定画面です。
20	TabLocsFragment.java	計算結果の地点タブ画面です。
21	TabParamsFragment.java	計算結果の計算設定タグ画面です。
22	TabRnsFragment.java	計算結果の経由地点タブ画面です。
23	TabTrdsFragment.java	計算結果の経由有料道路タブ画面です。
24	UserInfoActivity.java	ユーザー情報画面です。
25	UserInfoFragment.java	ユーザー情報画面です。

⑥　データベース関連

パス：　¥Sample¥Source¥ActServiceSample¥src¥jp¥co¥act_inc¥actservicesample¥db

項番	名前	内容
1	LocsDBManger.java	地点データベースを管理します。
2	ResultsDBManger.java	計算結果データベースを管理します。

⑦　その他

パス：　¥Sample¥Source¥ActServiceSample¥src¥jp¥co¥act_inc¥actservicesample

項番	名前	内容
1	AdminPrefManager.java	管理サービス関連のプリファレンスを管理します。
2	Attachable.java	アクティビティやフラグメントと結び付けるインターフェースです。
3	BundleKey.java	インテントで使用するキーを定義します。
4	CalcArea3PrefManager.java	到達圏／流入圏計算のプリファレンスを管理します。

5	CalcRoutePrefManager.java	ルート計算のプリファレンスを管理します。
6	LeveledAddressItem.java	住所レベル別住所データです。
7	LocLayer.java	計算地点用レイヤーです。
8	Logger.java	ログを出力します。
9	MyApp.java	アプリケーションクラスです。
10	PolygonLayer.java	ポリゴン用レイヤーです。
11	PolyLineLayer.java	ポリライン用レイヤーです。
12	RangedEditPreference.java	入力範囲が指定できるエディット設定です。
13	Utilities.java	ユーティリティクラスです。

3.2. ライブラリファイル

パス：¥Sample¥Source¥ActServiceSample¥libs

項番	名前	内容
1	android-support-v4.jar	サポートライブラリです。

3.3. リソースファイル

① drawable

パス：¥Sample¥Source¥ActServiceSample¥res¥drawable

項番	名前	内容
1	tool_button.xml	ツールボタンの描画設定です。
2	tool_button_pressed.xml	ツールボタンが押された場合の描画設定です。

② drawable-hdpi

パス：¥Sample¥Source¥ActServiceSample¥res¥drawable-hdpi

項番	名前	内容
1	ic_menu_address.png	住所検索アイコンです。
2	ic_menu_calc.png	距離計算アイコンです。
3	ic_menu_location.png	地点アイコンです。
4	ic_menu_results.png	計算結果アイコンです。

③ layout

パス：¥Sample¥Source¥ActServiceSample¥res¥layout

項番	名前	内容
1	act_map_activity.xml	メイン画面のレイアウトです。
2	act_map_fragment.xml	メイン画面のレイアウトです。

3	address_search_activity.xml	住所検索画面のレイアウトです。
4	address_search_fragment.xml	住所検索画面のレイアウトです。
5	calc_area3_pref.xml	到達圏／流入圏計算設定画面用レイアウトです。
6	calc_route_pref.xml	ルート計算設定画面用レイアウトです。
7	edit_dialog.xml	エディットダイアログ用レイアウトです。
8	grid_item.xml	グリッドアイテム用レイアウトです。
9	loc_activity.xml	地点一覧画面用レイアウトです。
10	loc_fragment.xml	地点一覧画面用レイアウトです。
11	ranged_edit_preference.xml	範囲指定設定用レイアウトです。
12	result_activity.xml	計算結果画面用レイアウトです。
13	road_selection_activity.xml	計算用道路選択画面用レイアウトです。
14	road_selection_fragment.xml	計算用道路選択画面用レイアウトです。
15	row_loc.xml	地点リストアイテム用レイアウトです。
16	row_loc_result.xml	計算結果の地点リストアイテム用レイアウトです。
17	row_rn.xml	経由地点リストアイテム用レイアウトです。
18	row_title.xml	見出し用レイアウトです。
19	row_trd.xml	経由有料道路リストアイテム用レイアウトです。
20	row_user_info.xml	ユーザー情報リストアイテム用レイアウトです。
21	service_pref.xml	サービス設定画面用レイアウトです。
22	tab_locs.xml	計算結果の地点タブ用レイアウトです。
23	tab_param.xml	計算結果の計算設定タブ用レイアウトです。
24	tab_rns.xml	計算結果の経由地点タブ用レイアウトです。
25	tab_trds.xml	計算結果の経由有料道路タブ用レイアウトです。
26	user_info_activity.xml	ユーザー情報画面用レイアウトです。
27	user_info_fragment.xml	ユーザー情報画面用レイアウトです。

④ layout-xlarge

パス：¥Sample¥Source¥ActServiceSample¥res¥layout-xlarge

項番	名前	内容
1	loc_fragment.xml	地点一覧画面用レイアウトです。

2	row_loc_result.xml	計算結果の地点リストアイテム用レイアウトです。
3	row_loc.xml	地点リストアイテム用レイアウトです。
4	row_rn.xml	経由地点リストアイテム用レイアウトです。
5	row_trd.xml	経由有料道路リストアイテム用レイアウトです。
6	tab_locs.xml	計算結果の地点タブ用レイアウトです。
7	tab_rns.xml	計算結果の経由地点タブ用レイアウトです。
8	tab_trds.xml	計算結果の経由有料道路タブ用レイアウトです。

⑤　menu

パス：¥Sample¥Source¥ActServiceSample¥res¥menu

項番	名前	内容
1	context_menu_loc.xml	地点一覧画面用メニューです。
2	menu.xml	メイン画面用メニューです。

⑥　values

パス：¥Sample¥Source¥ActServiceSample¥res¥values

項番	名前	内容
1	arrays.xml	設定画面用配列です。
2	custom_attributes.xml	範囲指定エディットボックス用属性です。
3	key_strings.xml	設定用キー文字列です。
4	strings.xml	メッセージ文字列です。
5	styles.xml	ツールボタン用スタイルです。

⑦　xml

パス：¥Sample¥Source¥ActServiceSample¥res¥xml

項番	名前	内容
1	calc_area3_preference.xml	到達圏／流入圏計算設定画面のレイアウトです。
2	calc_one_way3_preference.xml	片道一括計算設定画面のレイアウトです。
3	calc_route_preference.xml	ルート計算設定画面のレイアウトです。
4	service_preference.xml	サービス設定画面のレイアウトです。

4. ACT 距離計算サービスの利用

4.1. サンプルプログラムの概要

ACT 距離計算サービスの利用方法につきましては、弊社ホームページからダウンロードできる「ACT 距離計算サービス　プログラミング概要」も参照してください。

ACT 距離計算サービスの利用に関するソースコードは、以下のフォルダにまとめられています。

- 管理サービス関連
  ¥Sample¥Source¥ActServiceSample ¥src¥jp¥co¥act_inc¥actservicesample ¥adminservice
- 距離計算サービス関連
  ¥Sample¥Source¥ActServiceSample ¥src¥jp¥co¥act_inc¥actservicesample ¥dcservice
- 住所検索サービス関連
  ¥Sample¥Source¥ActServiceSample ¥src¥jp¥co¥act_inc¥actservicesample ¥gcservice

AdminService クラスのメソッドをコールすることで ACT 距離計算サービスの管理サービスを利用することができます。

項番	メソッド名	説明
1	getUserInformation2	ユーザー情報を取得します。
2	getUsableNWInfos	使用できる計算用道路データを取得します。
3	getUsableDCServer	使用できる距離計算サーバーを取得します。
4	getUsableGeoServer	使用できる住所検索サーバーを取得します。

DCService クラスのメソッドをコールすることで ACT 距離計算サービスの距離計算サービスを利用することができます。

項番	メソッド名	説明
1	calcRoute	ルート計算を実行します。
2	calcOptRoute	最短ルート計算を実行します。
3	calcStraightLine	直線距離計算を実行します。
4	calcArea3	到達圏／流入圏計算を実行します。
5	calcOneWay3	片道一括計算を実行します。
6	getNearestNode	最寄ノード取得を実行します。
7	enumNearNode	近傍ノード列挙を実行します。

8	getNearestLink2	最寄リンク取得を実行します。
9	getSpeed2	道路速度取得を実行します。
10	setSpeed2	道路速度設定を実行します。
11	getNetworkInfo	計算用道路データ情報取得を実行します。
12	getDCWSMessage	距離計算サービスのエラーメッセージを取得します。

GCServiceクラスのメソッドをコールすることでACT距離計算サービスの住所検索サービスを利用することができます。

項番	メソッド名	説明
1	searchAddress	住所検索を実行します。
2	searchAddressCode	住所コード検索を実行します。
3	countLowerAddressByCode	下位住所数取得を実行します。
4	enumLowerAddressByCode	下位住所列挙を実行します。
5	createFullAddressByCode	全住所取得を実行します。
6	getNearestAddress	最寄住所取得を実行します。
7	searchZip	郵便番号検索を実行します。
8	countAddressByZip	郵便番号対応住所数取得を実行します。
9	enumAddressByZip	郵便番号対応住所列挙を実行します。
10	getGCWSMessage	住所検索サービスのエラーメッセージを取得します。

本サンプルプログラムは以下のデータをプリファレンスに保存しています。

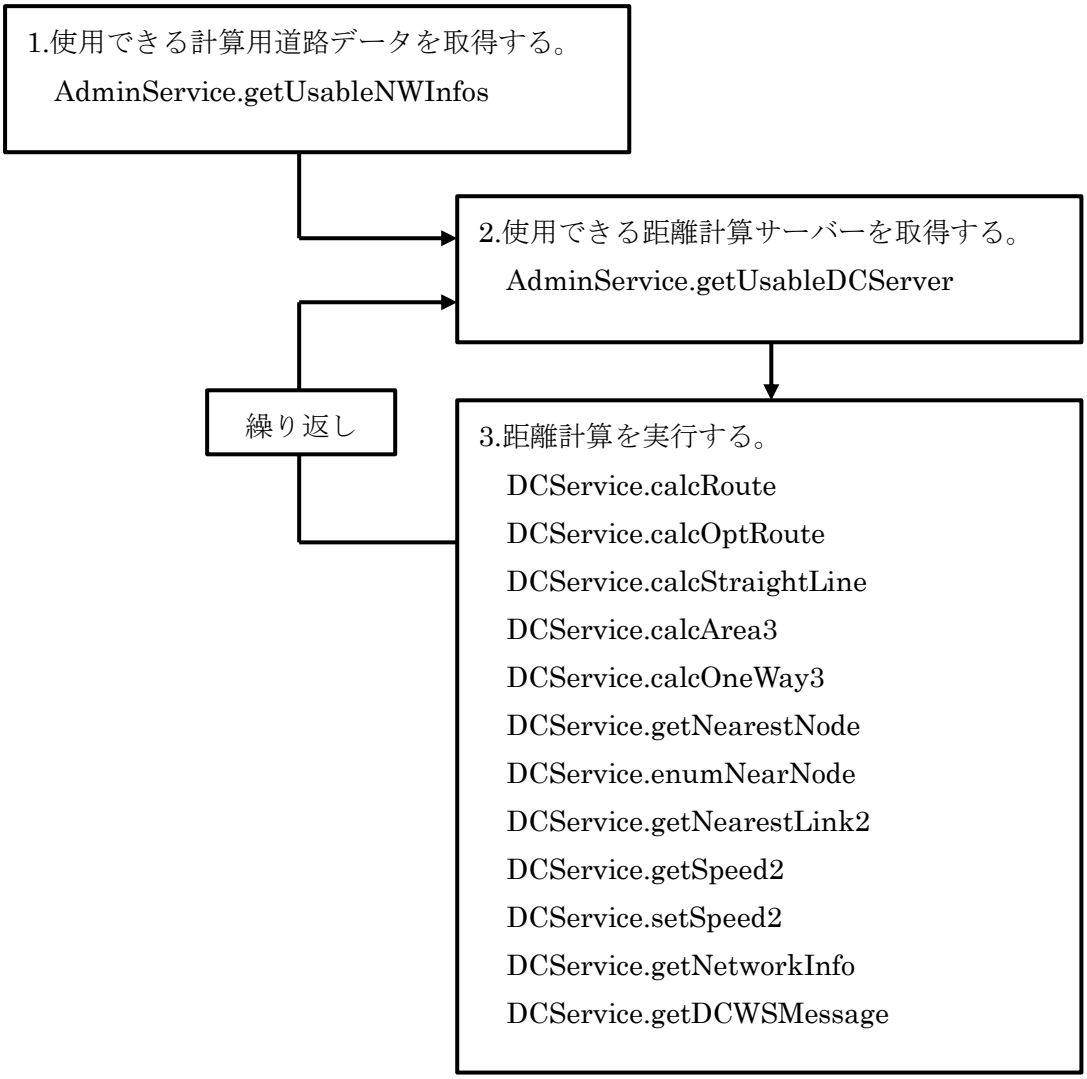
- ログイン設定
- 使用する計算用道路データ情報
- ユーザー情報
- 距離計算設定

本サンプルプログラムは以下のデータをデータベースに保存しています。

- 計算対象地点
- 距離計算結果

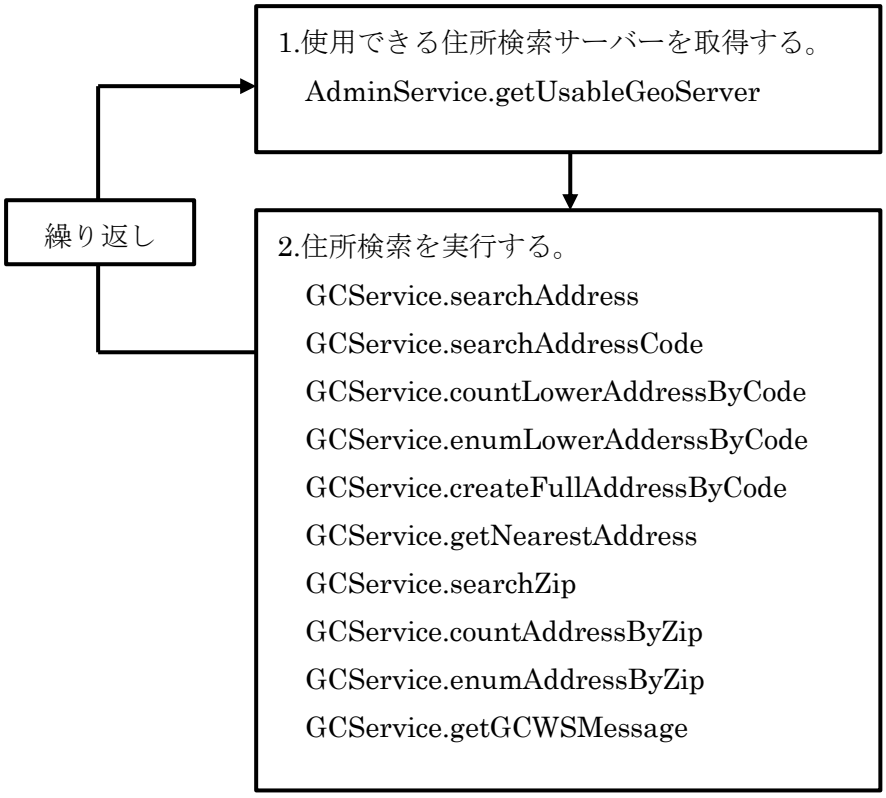
4.2. 呼び出し手順

① 距離計算の場合



（注）計算を繰り返す場合には、事前に AdminService.getUserInformation2 でユーザー情報を取得して、計算間隔の待機処理を行ってください。

② 住所検索の場合



（注）住所検索を繰り返す場合には、事前に AdminService.getUserInformation2 でユーザー情報を取得して、検索間隔の待機処理を行ってください。

4.3. 管理サービス

4.3.1. AdminService.getUserInformation2

```
public static GetUserInformation2Response getUserInformation2(
    String url,
    boolean useSsl,
    Environ environ
)
```

ユーザー情報を取得します。

パラメータ

項番	名前	説明
1	url	管理サーバーURL
2	useSsl	SSL を使用するかどうか
3	environ	環境設定

environ に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	userID	ユーザーID
2	uuID	パスワード

戻り値

結果を格納した GetUserInformation2Response

変更されるフィールドは下表の通りです。

項番	名前	説明
1	result	戻り値 成功時は true です。
2	userInfo	ユーザー情報
3	exception	実行中に発生した例外

成功時に変更される userInfo のフィールドは下表の通りです。

項番	名前	説明
1	memberKind	会員種別
2	expiryDate	会員期限（YYYY/MM/DD）
3	calcInterval	距離計算間隔(単位：秒)
4	calcRouteMaxLocs	ルート計算可能最大地点数
5	calcOptRouteMaxLocs	最短ルート計算可能最大地点数
6	calcAreaMaxTime	到達圏／流入圏計算可能最長時間（-1 は無制限）
7	calcAreaMaxDist	到達圏／流入圏計算可能最長距離（-1 は無制限）
8	calcAreaMaxLocs	到達圏／流入圏計算可能最大地点数
9	geoMaxAddressLevel	最大住所検索レベル(1:都道府県、2:市区町村、4:大字丁目、16:街区、 32:号)
10	geoInterval	住所検索計算間隔（単位：秒、0 は無制限）
11	geoEnumInterval	下位住所列挙計算間隔（単位：秒、0 は無制限）
12	geoMaxLocs	住所検索可能最大地点数
13	calcCount	計算回数
14	maxCalcCount	最大計算回数（-1 は無制限）
15	geoCount	住所検索回数
16	maxGeoCount	最大住所検索回数（-1 は無制限）
17	expand_I32_1	拡張数値１（未使用）
18	expand_I32_2	拡張数値２（未使用）
19	expand_I32_3	拡張数値３（未使用）
20	expand_I32_4	拡張数値４（未使用）
21	expand_Str_1	拡張文字列１（未使用）
22	expand_Str_2	拡張文字列２（未使用）
23	expand_Str_3	拡張文字列３（未使用）
24	expand_Str_4	拡張文字列４（未使用）

4.3.2. AdminService.getUsableNWInfos

```
public static GetUsableNWInfosResponse getUsableNWInfos(
    String url,
    boolean useSsl,
    Environ environ
)
```

使用可能な計算用道路データを取得します。

パラメータ

項番	名前	説明
1	url	管理サーバーURL
2	useSsl	SSL を使用するかどうか
3	environ	環境設定

environ に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	userID	ユーザーID
2	uuID	パスワード

戻り値

結果を格納した GetUsableNWInfosResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	result	戻り値 成功時は true です。
2	nwInfos	計算用道路データ情報リスト
3	exception	実行中に発生した例外

成功時に変更される nwInfos の各要素のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.3.3. AdminService.getUsableDCServer

```
public static GetUsableDCServerResponse getUsableDCServer(
    String url,
    boolean useSsl,
    Environ environ,
    NWInfo nwInfo
)
```

使用可能な距離計算サーバーを取得します。

パラメータ

項番	名前	説明
1	url	管理サーバーURL
2	useSsl	SSL を使用するかどうか
3	environ	環境設定
4	nwInfo	使用する計算用道路データ情報 getUsableNWInfos で取得したリストの中から使用する計算用道路データ情報を設定してください。

environ に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	userID	ユーザーID
2	uuID	パスワード

戻り値

結果を格納した GetUsableDCServerResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	result	戻り値 成功時は true です。
2	environ	距離計算サーバーの情報を格納した環境設定
3	exception	実行中に発生した例外

成功時に変更される environ のフィールドは下表の通りです。

項番	名前	説明
1	userID	ユーザーID
2	uuID	パスワード
3	hwuID	ハードウェア ID
4	wsUrl	距離計算サービスの URL
5	host	距離計算リスナーの IP アドレス
6	port	距離計算リスナーのポート番号

4.3.4. AdminService.getUsableGeoServer

```
public static GetUsableGeoServerResponse getUsableGeoServer(
    String url,
    boolean useSsl,
    Environ environ
)
```

使用可能な住所検索サーバーを取得します。

パラメータ

項番	名前	説明
1	url	管理サーバーURL
2	useSsl	SSL を使用するかどうか
3	environ	環境設定

environ に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	userID	ユーザーID
2	uuID	パスワード

戻り値

結果を格納した GetUsableGeoServerResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	result	戻り値 成功時は true です。
2	environ	住所検索サーバーの情報を格納した環境設定
3	exception	実行中に発生した例外

成功時に変更される environ のフィールドは下表の通りです。

項番	名前	説明
1	userID	ユーザーID
2	uuID	パスワード
3	hwuID	ハードウェア ID
4	wsUrl	住所検索サービスの URL
5	host	住所検索リスナーの IP アドレス
6	port	住所検索リスナーのポート番号

4.4. 距離計算サービス

4.4.1. DCService.calcRoute

```
public static CalcRouteResponse calcRoute(
    boolean useSsl,
    Environ environ,
    CRParam param,
    ArrayList<Loc> locs
)
```

ルート計算を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminService.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	ルート計算パラメータ
4	locs	ルート計算をする地点のリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	calcKind	計算方法 1：時間最短、2：距離最短
4	transport	使用交通機関フラグ 計算用道路データによって使用できる交通機関は異なります。 下記値の論理和です。 0：高速道路を使用しない 1：高速道路を使用する 2：北海道～青森間、沖縄～鹿児島間のみフェリーを使

		用する※1 4：フェリーを使用する 8：鉄道を使用する 16：航空機を使用する 64：内航船を使用する、または旅客鉄道を使用する場合特急を使用する
5	pntsOutput	ポイントリスト出力フラグ 走行ルートリストの出力を指定します。 0：出力しない、1：出力する
6	rnsOutput	経由道路名リスト出力フラグ 経由道路名リストの出力を指定します。 0：出力しない、1：出力する。
7	trdsOutput	経由有料道路リスト出力フラグ 経由有料道路リストの出力を指定します。 0：出力しない、1：出力する
8	highwayOutput	高速道路区間出力フラグ 高速道路区間の出力を指定します。 0：出力しない、1：出力する
9	tollOutput	通行料金出力フラグ 通行料金の出力を指定します。 0：出力しない、1：出力する
10	routeKind	ルート種別指定 走行ルートの種別を指定します。 0：簡易ルート、1：詳細ルート

※1　本州―北海道間、本州―沖縄間の道路距離を簡易的に計算するための設定

locs の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 地点をノードコードで指定する場合に使用します。通常は”0”を設定してください。
2	lon	経度（度単位） node を指定した場合、使用されません。
3	lat	緯度（度単位） node を指定した場合、使用されません。
4	tag	タグ情報 ユーザが自由に使用できる文字列。地点名称などの識別情報を設定できます。

戻り値

結果を格納した CalcRouteResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	locs	地点リスト
3	pnts	ルートポイントリスト
4	rns	経由地点リスト
5	trds	経由有料道路リスト
6	nwInfo	計算用道路データ情報
7	exception	実行中に発生した例外

成功時に変更される locs の各要素のフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 最寄ノードコードが出力されます。
2	lon	経度（度単位） node を指定して計算した場合、最寄ノードの経度が出力されます。
3	lat	緯度（度単位） node を指定して計算した場合、最寄ノードの緯度が出力されます。
4	time	所要時間（分単位）
5	dSec	所要時間（0.1 秒単位）
6	dist	道のり（m 単位）
7	highwayTime	高速道路区間の所要時間（分単位）
8	highwayDSec	高速道路区間の所要時間（0.1 秒単位）
9	highwayDist	高速道路区間の道のり（m 単位）
10	toll_SS	二輪・軽の通行料金（円単位）
11	toll_S	普通車の通行料金（円単位）
12	toll_M	中型車の通行料金（円単位）
13	toll_L	大型車の通行料金（円単位）
14	toll_LL	特大車の通行料金（円単位）
15	tag	タグ情報

成功時に変更される pnts の各要素のフィールドは下表の通りです。

項番	名前	説明
1	lon	経度（度単位）
2	lat	緯度（度単位）

成功時に変更される rns の各要素のフィールドは下表の通りです。

項番	名前	説明
1	locFlag	ロケーションフラグ 1：計算対象地点 0：上記以外
2	linkFlag	リンクフラグ 0：リンク以外、1：リンク
3	nlCode	ノードリンク番号 リンクフラグが 0 の場合、ノードコード リンクフラグが 1 の場合、リンク番号
4	name	名称 ロケーションフラグが 1 の場合、ロケーションの Tag 情報。 リンクフラグが 0 の場合、交差点名称。 リンクフラグが 1 かつインターチェンジタイプが 0 の場合、道路名称。 リンクフラグが 1 かつインターチェンジタイプが 0 でないの場合、インターチェンジ名称
5	time	所要時間（分単位）
6	dSec	所要時間（0.1 秒単位）
7	dist	道のり（m 単位）
8	icType	インターチェンジタイプ 0：インターチェンジではない 0 以外：インターチェンジ
9	toll_SS	二輪・軽の通行料金（円単位）
10	toll_S	普通車の通行料金（円単位）
11	toll_M	中型車の通行料金（円単位）
12	toll_L	大型車の通行料金（円単位）
13	toll_LL	特大車の通行料金（円単位）
14	lon	経度（度単位）
15	lat	緯度（度単位）

成功時に変更される trds の各要素のフィールドは下表の通りです。

項番	名前	説明
1	inRdName	入口または課金の有料道路名称 入口または課金の料金所名称、出口の有料道路名称、出口の料金所名称が空白の場合、ロケーションのタグ情報が格納されます。
2	inICName	入口または課金の料金所名称
3	outRdName	出口の有料道路名称
4	outICName	出口の料金所名称
5	toll_SS	二輪・軽の通行料金（円単位）
6	toll_S	普通車の通行料金（円単位）
7	toll_M	中型車の通行料金（円単位）
8	toll_L	大型車の通行料金（円単位）
9	toll_LL	特大車の通行料金（円単位）
10	dist	距離（km 単位） 有料道路のキロポスト 。 0 または有効な値でない場合があります。

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.2. DCService.calcOptRoute

```
public static CalcOptRouteResponse calcOptRoute(
    boolean useSsl,
    Environ environ,
    CRParam param,
    ArrayList<Loc> locs
)
```

最短ルート計算を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	ルート計算パラメータ
4	locs	ルート計算をする地点のリスト

設定が必要なパラメータは calcRoute と同様なので、calcRoute の情報を参照してください。

戻り値

結果を格納した CalcOptRouteResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙 ACT 距離計算サービスプログラミング概要を参照してください。
2	locs	地点リスト
3	pnts	ルートポイントリスト
4	rns	経由地点リスト
5	trds	経由有料道路リスト
6	nwInfo	計算用道路データ情報
7	exception	実行中に発生した例外

変更されるデータは calcRoute と同様なので、calcRoute の情報を参照してください。

4.4.3. DCService.calcStraightLine

```
public static CalcStraightLineResponse calcStraightLine(
    boolean useSsl,
    Environ environ,
    CSLParam param,
    ArrayList<Loc> locs
)
```

直線距離計算を実行します。
この計算で求める直線距離は、地球を回転楕円体と仮定した場合の最短測地線長です。
実際の長さとは異なる可能性がありますので注意してください。
また１地点からその対蹠点（回転楕円体の裏側）付近への距離計算はできません。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminService.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	直線距離計算パラメータ
4	locs	直線距離計算をする地点のリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

locs の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	lon	経度（度単位）
2	lat	緯度（度単位）
3	tag	タグ情報 ユーザが自由に使用できる文字列。地点名称などの識別情報を設定できます。

戻り値

結果を格納した CalcStraightLineResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	locs	地点リスト
3	pnts	ルートポイントリスト
4	exception	実行中に発生した例外

成功時に変更される locs の各要素のフィールドは下表の通りです。

項番	名前	説明
1	dist	累積直線距離（m 単位）

成功時に変更される pnts の各要素のフィールドは下表の通りです。

項番	名前	説明
1	lon	経度（度単位）
2	lat	緯度（度単位）

4.4.4. DCService.calcArea3

```
public static CalcArea3Response calcArea3(
    boolean useSsl,
    Environ environ,
    CA3Param param
)
```

到達圏／流入圏計算を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	到達圏／流入圏計算パラメータ

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	calcKind	計算方法 1：時間最短、2：距離最短
4	transport	使用交通機関フラグ 計算用道路データによって使用できる交通機関は異なります。 下記値の論理和です。 0：高速道路を使用しない 1：高速道路を使用する 2：北海道～青森間、沖縄～鹿児島間のみフェリーを使用する※1 4：フェリーを使用する 8：鉄道を使用する

		16：航空機を使用する 64：内航船を使用する、または旅客鉄道を使用する場合特急を使用する
5	startPnt	到達圏／流入圏計算の中心地点
6	polygonKind	ポリゴン種別 1：単一ポリゴン、2：複数ポリゴン
7	areaKind	到達圏／流入圏計算種別 計算範囲の種別を指定します。 1：時間圏、2：距離圏
8	areaRange	到達圏／流入圏範囲 到達圏／流入圏計算での範囲を指定します。時間圏の場合、0.1 秒単位、距離圏の場合、m 単位で指定します。
9	polygonLevel	到達圏／流入圏ポリゴンレベル 到達ポリゴンのポリゴンレベルを指定します。 -1：凸型(簡易)ポリゴン 0～20：凹凸型(通常)ポリゴン（大きいほど細かくなります）
10	donutPolygonLevel	到達圏／流入圏ドーナツポリゴンレベル 到達ポリゴン内で到達できない範囲のポリゴンレベルを指定します。 -2：表示しない -1：凸型(簡易)ポリゴン 0～20：凹凸型(通常)ポリゴン（大きいほど細かくなります）
11	areaDirection	範囲方向 到達圏または流入圏を指定します。 0：到達圏、4096：流入圏
12	polygonDetail	ポリゴン詳細設定　下記値の論理和 計算時に使用する機能を指定してください。 0：詳細機能を使用しない 1：中間リンク補間 2：末端リンク補間

※1　本州―北海道間、本州―沖縄間の道路距離を簡易的に計算するための設定

param の startPnt に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 地点をノードコードで指定する場合に使用します。通常は”0”を設定してください。
2	lon	経度（度単位） node を指定した場合、使用されません。
3	lat	緯度（度単位） node を指定した場合、使用されません。

戻り値

結果を格納した CalcArea3Response

変更されるフィールドは下表の通りです。

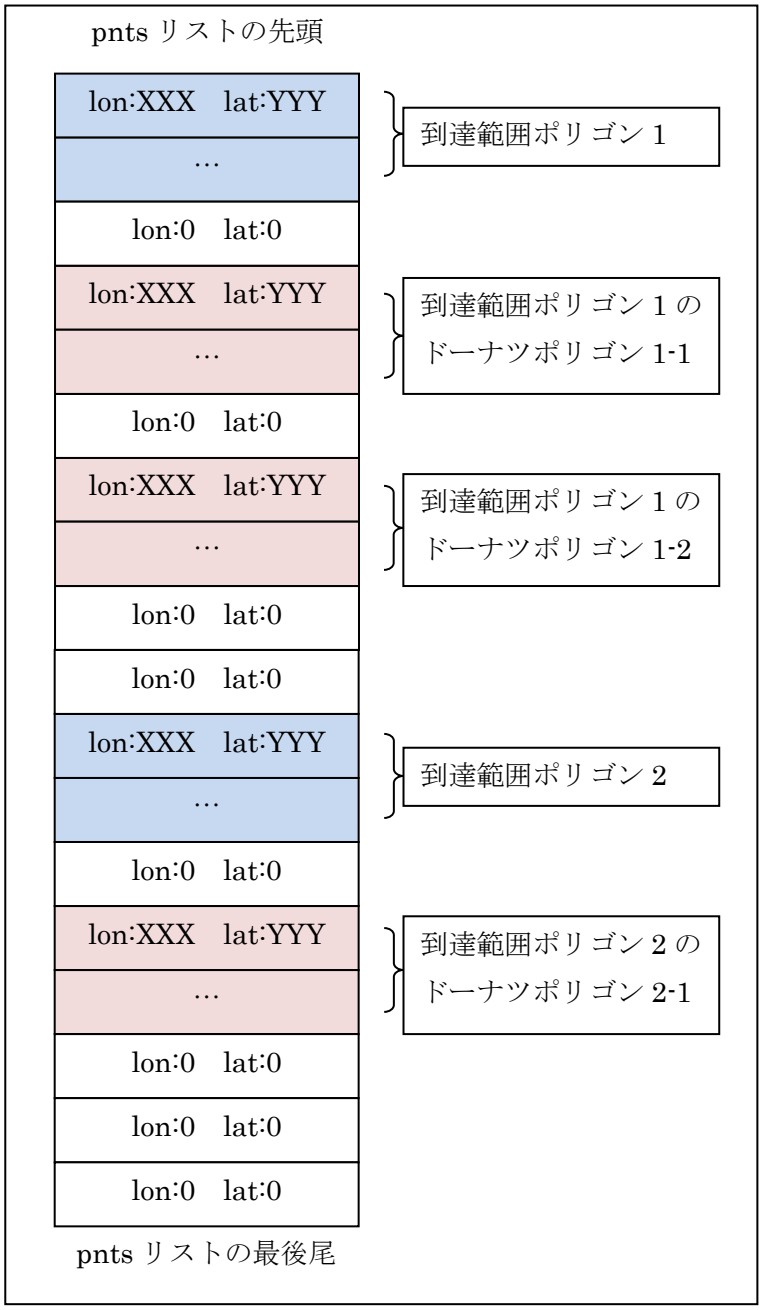
項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	param	計算パラメータ
3	pnts	ポリゴンポイントリスト
4	nwInfo	計算用道路データ情報
5	exception	実行中に発生した例外

成功時に変更される param のフィールドは下表の通りです。

項番	名前	説明
1	polygonDetail	ポリゴン詳細設定 計算後は使用された機能が設定されています。

成功時に変更される pnts の各要素のフィールドは下表の通りです。

項番	名前	説明
1	lon	経度（度単位）
2	lat	緯度（度単位）



ポリゴンデータの格納順は左図のようになっています。

各ポリゴンの区切りとして、lon と lat に数値として 0 が格納されているポイントが使用されます。

まず到達範囲ポリゴンが格納されています。

次に区切りが一つある場合は、直前の到達範囲ポリゴン内のドーナツポリゴンが格納されています。

区切りが二つ続く場合は、次の到達範囲ポリゴンが格納されています。

区切りが三つ続く場合はポリゴンデータの終了を意味します。

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.5. DCService.calcOneWay3

```
public static CalcOneWay3Response calcOneWay3(
    boolean useSsl,
    Environ environ,
    COW3Param param,
    ArrayList<Loc> locs
)
```

片道一括計算を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	片道一括計算パラメータ
4	locs	片道一括計算をする地点のリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	calcKind	計算方法 1：時間最短、2：距離最短
4	transport	使用交通機関フラグ 計算用道路データによって使用できる交通機関は異なります。 下記値の論理和です。 0：高速道路を使用しない 1：高速道路を使用する 2：北海道～青森間、沖縄～鹿児島間のみフェリーを使用する※1

		4：フェリーを使用する 8：鉄道を使用する 16：航空機を使用する 64：内航船を使用する、または旅客鉄道を使用する場合特急を使用する
5	startPnt	到達圏／流入圏計算の中心地点
6	areaKind	到達圏／流入圏計算種別 計算範囲の種別を指定します。 1：時間圏、2：距離圏
7	areaRange	到達圏／流入圏範囲 到達圏／流入圏計算での範囲を指定します。時間圏の場合、0.1 秒単位、距離圏の場合、m 単位で指定します。
8	areaDirection	範囲方向 到達圏または流入圏を指定します。 0：到達圏、4096：流入圏
9	highwayOutput	高速道路区間出力フラグ 高速道路区間の出力を行うか指定します。 0：出力しない、1：出力する
10	tollOutput	通行料金出力フラグ 通行料金の出力を行うか指定します。 0：出力しない、1：出力する

※1　本州―北海道間、本州―沖縄間の道路距離を簡易的に計算するための設定

param の startPnt に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 地点をノードコードで指定する場合に使用します。通常は”0”を設定してください。
2	lon	経度（度単位） node を指定した場合、使用されません。
3	lat	緯度（度単位） node を指定した場合、使用されません。

locs の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 地点をノードコードで指定する場合に使用します。通常は”0”を設定してください。
2	lon	経度（度単位） node を指定した場合、使用されません。
3	lat	緯度（度単位） node を指定した場合、使用されません。
4	tag	タグ情報 ユーザが自由に使用できる文字列。地点名称などの識別情報を設定できます。

戻り値

結果を格納した CalcOneWay3Response

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	locs	地点リスト
3	nwInfo	計算用道路データ情報
4	exception	実行中に発生した例外

成功時に変更される locs の各要素のフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 最寄ノードコードが出力されます。
2	lon	経度（度単位） node を指定して計算した場合、最寄ノードの経度が出力されます。
3	lat	緯度（度単位） node を指定して計算した場合、最寄ノードの緯度が出

		力されます。
4	time	所要時間（分単位）
5	dSec	所要時間（0.1 秒単位）
6	dist	道のり（m 単位）
7	highwayTime	高速道路区間の所要時間（分単位）
8	highwayDSec	高速道路区間の所要時間（0.1 秒単位）
9	highwayDist	高速道路区間の道のり（m 単位）
10	toll_SS	二輪・軽の通行料金（円単位）
11	toll_S	普通車の通行料金（円単位）
12	toll_M	中型車の通行料金（円単位）
13	toll_L	大型車の通行料金（円単位）
14	toll_LL	特大車の通行料金（円単位）
15	tag	タグ情報

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.6. DCService.getNearestNode

```
public static GetNearestNodeResponse getNearestNode(
    boolean useSsl,
    Environ environ,
    GNNParam param,
    ArrayList<Loc> locs
)
```

最寄ノード取得を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	最寄ノード取得パラメータ
4	locs	最寄ノードを取得をする地点のリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	transport	使用交通機関フラグ 計算用道路データによって使用できる交通機関は異なります。 下記値の論理和です。 0：高速道路を使用しない 1：高速道路を使用する 2：北海道～青森間、沖縄～鹿児島間のみフェリーを使用する※1 4：フェリーを使用する 8：鉄道を使用する

		16：航空機を使用する 64：内航船を使用する、または旅客鉄道を使用する場合特急を使用する

※1　本州―北海道間、本州―沖縄間の道路距離を簡易的に計算するための設定

locs の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 地点をノードコードで指定する場合に使用します。通常は”0”を設定してください。
2	lon	経度（度単位） node を指定した場合、使用されません。
3	lat	緯度（度単位） node を指定した場合、使用されません。
4	tag	タグ情報 ユーザが自由に使用できる文字列。地点名称などの識別情報を設定できます。

戻り値

結果を格納した GetNearestNodeResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	locs	地点リスト
3	nwInfo	計算用道路データ情報
4	exception	実行中に発生した例外

成功時に変更される locs の各要素のフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 最寄ノードコードが出力されます。

2	lon	経度（度単位） node を指定して計算した場合、最寄ノードの経度が出力されます。
3	lat	緯度（度単位） node を指定して計算した場合、最寄ノードの緯度が出力されます。
4	tag	タグ情報

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.7. DCService.enumNearNode

```
public static EnumNearNodeResponse enumNearNode(
    boolean useSsl,
    Environ environ,
    ENNParam param
)
```

近傍ノード列挙を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	近傍ノード列挙パラメータ

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	transport	使用交通機関フラグ 計算用道路データによって使用できる交通機関は異なります。 下記値の論理和です。 0：高速道路を使用しない 1：高速道路を使用する 2：北海道～青森間、沖縄～鹿児島間のみフェリーを使用する※1 4：フェリーを使用する 8：鉄道を使用する 16：航空機を使用する 64：内航船を使用する、または旅客鉄道を使用する場

		合特急を使用する
4	startPnt	近傍ノード列挙の中心地点

※１　本州―北海道間、本州―沖縄間の道路距離を簡易的に計算するための設定

param の startPnt に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 地点をノードコードで指定する場合に使用します。通常は”0”を設定してください。
2	lon	経度（度単位） node を指定した場合、使用されません。
3	lat	緯度（度単位） node を指定した場合、使用されません。

戻り値

結果を格納した EnumNearNodeResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	locs	地点リスト
3	nwInfo	計算用道路データ情報
4	exception	実行中に発生した例外

成功時に変更される locs の各要素のフィールドは下表の通りです。

項番	名前	説明
1	node	ノードコード 最寄ノードコードが出力されます。
2	lon	経度（度単位） 最寄ノードの経度が出力されます。
3	lat	緯度（度単位） 最寄ノードの緯度が出力されます。

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.8. DCService.getNearestLink2

```
public static GetNearestLink2Response getNearestLink2(
    boolean useSsl,
    Environ environ,
    GNLParam param
)
```

最寄リンク取得を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminService.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	最寄リンク取得パラメータ

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	transport	使用交通機関フラグ 計算用道路データによって使用できる交通機関は異なります。 下記値の論理和です。 0：高速道路を使用しない 1：高速道路を使用する 2：北海道～青森間、沖縄～鹿児島間のみフェリーを使用する※1 4：フェリーを使用する 8：鉄道を使用する 16：航空機を使用する 64：内航船を使用する、または旅客鉄道を使用する場合

		特急を使用する
4	lon	検索開始経度（度単位）
5	lat	検索開始緯度（度単位）
6	range	検索範囲（m 単位）

※1　本州―北海道間、本州―沖縄間の道路距離を簡易的に計算するための設定

戻り値

結果を格納した GetNearestLink2Response

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	linkCode	最寄リンクコード 見つからなかった場合は 0 になります。
3	nwInfo	計算用道路データ情報
4	exception	実行中に発生した例外

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.9. DCService.getSpeed2

```
public static GetSpeed2Response getSpeed2(
    boolean useSsl,
    Environ environ,
    GSParam param
)
```

道路速度取得を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminService.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	道路速度取得パラメータ

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	linkCode	速度を取得するリンクのリンクコード
4	speedFlag	速度フラグ 取得する速度を指定します。 両方指定した場合はメモリ上の速度が優先されます 16：メモリ上の速度 32：ファイル上の速度

戻り値

結果を格納した GetSpeed2Response

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	speedAB	ノード A から B への速度（0.1km/h 単位）
3	speedBA	ノード B から A への速度（0.1km/h 単位）
4	nodeA	リンクの始点ノード
5	nodeB	リンクの終点ノード
6	nwInfo	計算用道路データ情報
7	exception	実行中に発生した例外

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.10. DCService.setSpeed2

```
public static SetSpeed2Response setSpeed2(
    boolean useSsl,
    Environ environ,
    SSParam param
)
```

道路速度設定を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminService.getUsableDCServer で取得した環境設定を使用して下さい。
3	param	道路速度設定パラメータ

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	linkCode	速度を変更するリンクのリンクコード
2	speedFlag	速度フラグ 取得する速度を指定します。 両方指定した場合はメモリ上の速度が優先されます 16：メモリ上の速度 32：ファイル上の速度
3	speedAB	設定するノード A から B への速度（0.1km/h 単位）
4	speedBA	設定するノード B から A への速度（0.1km/h 単位）

戻り値

結果を格納した SetSpeed2Response

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	OldSpeedAB	変更前のノード A から B への速度（0.1km/h 単位）
3	OldSpeedBA	変更前のノード B から A への速度（0.1km/h 単位）
4	nwInfo	計算用道路データ情報
5	exception	実行中に発生した例外

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID
6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.11. DCService.getNetworkInfo

```
public static GetNetworkInfoResponse getNetworkInfo(
    boolean useSsl,
    Environ environ
)
```

計算用道路データ情報を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminService.getUsableDCServer で取得した環境設定を使用して下さい。

戻り値

結果を格納した GetNetworkInfoResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	nwInfo	計算用道路データ情報
3	exception	実行中に発生した例外

成功時に変更される nwInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	計算用道路データの格納フォルダ
2	note	計算用道路データの名称
3	isDetailExist	詳細ルート出力可能フラグ
4	isTollExist	通行料金出力可能フラグ
5	nwID	計算用道路データを識別するためのネットワーク ID

6	nwID_WithoutSpeed	速度情報を無視した計算用道路データを識別するためのネットワーク ID 番号

4.4.12. DCService.getDCWSMessage

```
public static GetDcwsMessageResponse getDCWSMessage(
    boolean useSsl,
    Environ environ,
    int code
)
```

距離計算サービスのエラーメッセージを取得します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableDCServer で取得した環境設定を使用して下さい。
3	code	エラーコード

戻り値

結果を格納した GetDcwsMessageResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultMessage	エラーメッセージ
2	exception	実行中に発生した例外

4.5. 住所検索サービス

4.5.1. GCService.searchAddress

```
public static SearchAddressResponse searchAddress(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    int searchAddressLevel,
    ArrayList<SearchAddressItem> items
)
```

住所検索を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	searchAddressLevel	最大住所検索レベル 1：都道府県、2：市区町村、8：大字・丁目、16：街区（番地・地番）、32：号（枝番）
5	items	住所検索アイテムのリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	searchFlag	検索フラグ 0：なし 1：全住所も取得する

items の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	searchAddress	検索住所 例　東京都港区新橋３－A
2	tag	タグ情報 ユーザが自由に使用できる文字列。メッシュ番号や届先名称などの識別情報を設定できます。

戻り値

結果を格納した SearchAddressResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	items	住所検索アイテムリスト
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される items の各要素のフィールドは下表の通りです。

項番	名前	説明
1	analyzedAddress	解析住所 SearchAddress の解析できた部分文字列が出力されます。 例　東京都港区新橋３－
2	restAddress	残住所 SearchAddress の解析できなかった部分文字列が出力されます。 例　A
3	addressLevel	結果住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　8

4	addressCode	結果住所コード 例　13103016003
5	lon	結果経度（度単位）
6	lat	結果緯度（度単位）
7	fullAddress	結果全住所 検索フラグで設定されていた場合、都道府県からの住所が出力されます。 例　東京都港区新橋３丁目
8	tag	タグ情報

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.2. GCService.searchAddressCode

```
public static SearchAddressCodeResponse searchAddressCode(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    ArrayList<SearchAddressCodeItem> items
)
```

住所コード検索を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	items	住所コード検索アイテムのリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

items の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	searchAddressCode	検索住所コード
2	tag	タグ情報 ユーザが自由に使用できる文字列。メッシュ番号や届先名称などの識別情報を設定できます。

戻り値

結果を格納した SearchAddressCodeResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	items	住所コード検索アイテムリスト
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される items の各要素のフィールドは下表の通りです。

項番	名前	説明
1	addressItem	住所アイテム 検索結果が出力されます。

addressItem 住所アイテムの内容は以下の通りです。

項番	名前	説明
1	addressCode	住所コード 例　13103016003
2	addressLevel	住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　8
3	upperAddressLevel	上位住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　2
4	lowerAddressLevel	下位住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　16
5	address	住所 例　新橋３丁目

6	kana	ひらがな住所 住所コードによっては登録されていない場合があります。 例　しんばし３ちょうめ
7	katakana	カタカナ住所 住所コードによっては登録されていない場合があります。 例　ｼﾝﾊﾞｼ３ﾁｮｳﾒ
8	zip	郵便番号 住所コードによっては登録されていない場合があります。 例　1050004
9	lon	結果経度（度単位）
10	lat	結果緯度（度単位）

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.3. GCService.countLowerAddressByCode

```
public static CountLowerAddressByCodeResponse countLowerAddressByCode(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    String searchAddressCode
)
```

下位住所数取得を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	searchAddressLevel	検索する住所コード

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

戻り値

結果を格納した CountLowerAddressByCodeResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービ

		スプログラミング概要」を参照してください。
2	count	下位住所数
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.4. GCService.enumLowerAddressByCode

```
public static EnumLowerAddressByCodeResponse enumLowerAddressByCode(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    String searchAddressCode
)
```

下位住所列挙を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	searchAddressLevel	検索する住所コード

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

戻り値

結果を格納した EnumLowerAddressByCodeResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービ

		スプログラミング概要」を参照してください。
2	items	住所アイテムリスト
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される items の各要素のフィールドは下表の通りです。

項番	名前	説明
1	addressCode	住所コード 例　13103016003
2	addressLevel	住所レベル 0：失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　8
3	upperAddressLevel	上位住所レベル 0：なし、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　2
4	lowerAddressLevel	下位住所レベル 0：なし、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　16
5	address	住所 例　新橋３丁目
6	kana	ひらがな住所 住所コードによっては登録されていない場合があります。 例　しんばし３ちょうめ
7	kataKana	カタカナ住所 住所コードによっては登録されていない場合があります。 例　ｼﾝﾊﾞｼ３ﾁｮｳﾒ
8	zip	郵便番号 住所コードによっては登録されていない場合があります。 例　1050004

9	lon	経度（度単位）
10	lat	緯度（度単位）

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.5. GCService.createFullAddressByCode

```
public static CreateFullAddressByCodeResponse createFullAddressByCode(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    ArrayList<CreateFullAddressByCodeItem> items
)
```

全住所取得を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	items	全住所取得アイテムのリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

items の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	searchAddressCode	検索住所コード
2	tag	タグ情報 ユーザが自由に使用できる文字列。メッシュ番号や届先名称などの識別情報を設定できます。

戻り値

結果を格納した CreateFullAddressByCodeResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	items	全住所取得アイテムリスト
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される items の各要素のフィールドは下表の通りです。

項番	名前	説明
1	addressLevel	住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　32
2	addressCode	住所コード 例　13103016003000070003
3	fullAddress	全住所 都道府県からの住所が出力されます。 例　東京都港区新橋３丁目７－４
4	prefAddress	都道府県住所 例　東京都
5	cityAddress	市区町村住所 例　港区
6	streetAddress	大字・丁目住所 例　新橋３丁目
7	blockAddress	番地・地番住所 例　7
8	houseAddress	号・枝番住所 例　4
9	ohterAddress	その他住所

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.6. GCService.getNearestAddress

```
public static GetNearestAddressResponse getNearestAddress(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    ArrayList<GetNearestAddressItem> items
)
```

最寄住所取得を実行します。
指定経緯度に一番近い大字・丁目レベル住所を取得できます。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	items	最寄住所取得アイテムのリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

items の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	searchLon	検索経度（度単位）
2	searchLat	検索緯度（度単位）
3	tag	タグ情報 ユーザが自由に使用できる文字列。メッシュ番号や届先名称などの識別情報を設定できます。

戻り値

結果を格納した GetNearestAddressResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	items	最寄住所取得アイテムリスト
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される items の各要素のフィールドは下表の通りです。

項番	名前	説明
1	fullAddress	全住所 都道府県からの住所が出力されます。
2	addressItem	住所アイテム
3	distance	距離（m 単位） 検索経緯度と結果住所の直線距離です。 計算上の値で、実測値ではありませんので注意してください。

addressItem 住所アイテムの内容は以下の通りです。

項番	名前	説明
1	addressCode	住所コード 例　13103016003
2	addressLevel	住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　8
3	upperAddressLevel	上位住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　2
4	lowerAddressLevel	下位住所レベル

		0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　16
5	address	住所 例　新橋３丁目
6	kana	ひらがな住所 住所コードによっては登録されていない場合があります。 例　しんばし３ちょうめ
7	katakana	カタカナ住所 住所コードによっては登録されていない場合があります。 例　ｼﾝﾊﾞｼ３ﾁｮｳﾒ
8	zip	郵便番号 住所コードによっては登録されていない場合があります。 例　1050004
9	lon	結果経度（度単位）
10	lat	結果緯度（度単位）

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.7. GCService.searchZip

```
public static SearchZipResponse searchZip(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    ArrayList<SearchZipItem> items
)
```

郵便番号検索を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	items	郵便番号検索アイテムのリスト

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。
3	searchFlag	検索フラグ 1：先頭データ。郵便番号に該当する住所データの先頭データを検索します。 8：中心位置。郵便番号に該当する住所の平均経緯度位置を検索します。 16：中心住所。郵便番号に該当する住所の平均経緯度に一番近い住所データを検索します。

items の各要素に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	searchZip	検索郵便番号
2	tag	タグ情報 ユーザが自由に使用できる文字列。メッシュ番号や届先名称などの識別情報を設定できます。

戻り値

結果を格納した SearchZipResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービスプログラミング概要」を参照してください。
2	items	郵便番号検索アイテムリスト
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される items の各要素のフィールドは下表の通りです。

項番	名前	説明
1	addressCode	結果住所コード 例　13103016003
2	addressLevel	結果住所レベル 0:失敗、1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　8
3	fullAddress	結果全住所 例　東京都港区新橋３丁目
4	lon	結果経度（度単位）
5	lat	結果緯度（度単位）

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.8. GCService.countAddressByZip

```
public static CountAddressByZipResponse countAddressByZip(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    String searchZip
)
```

郵便番号対応住所数取得を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	searchZip	検索する郵便番号（7 桁）

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

戻り値

結果を格納した CountAddressByZipResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービ

		スプログラミング概要」を参照してください。
2	count	郵便番号対応住所数
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.9. GCService.enumAddressByZip

```
public static EnumAddressByZipResponse enumAddressByZip(
    boolean useSsl,
    Environ environ,
    CommonParam param,
    String searchZip
)
```

郵便番号対応住所列挙を実行します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	param	住所検索用共通パラメータ
4	searchZip	検索する郵便番号（7 桁）

param に設定する必要があるフィールドは下表の通りです。

項番	名前	説明
1	datum	経緯度の座標系識別子 結果経緯度の測地系を指定します。 0：日本測地系、1：世界測地系
2	coordinateFormat	経緯度の座標フォーマット識別子 現状では 0 を指定してください。

戻り値

結果を格納した EnumAddressByZipResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultCode	戻り値 成功時は 0 です。 エラーコードについては別紙「ACT 距離計算サービ

		スプログラミング概要」を参照してください。
2	items	郵便番号アイテムリスト
3	geoInfo	住所データ情報
4	exception	実行中に発生した例外

成功時に変更される items の各要素のフィールドは下表の通りです。

項番	名前	説明
1	addressLevel	住所レベル 1：都道府県、2：市区町村、8：大字・丁目、16：街区、32：号 例　8
2	addressCode	住所コード 例　13103016003
3	fullAddress	全住所 都道府県からの住所が出力されます。 例　東京都港区新橋３丁目
4	zip	郵便番号 例　1050004
5	lon	経度（度単位）
6	lat	緯度（度単位）

成功時に変更される geoInfo のフィールドは下表の通りです。

項番	名前	説明
1	folder	住所データフォルダパス
2	note	住所データの説明文
3	geoID	住所データ ID １～８桁：住所データリリース年月日 ９～１０桁：住所データの住所レベル １１～１２桁：ユーザーの最大住所レベル １３～３２桁：未使用

4.5.10. GCService.getGCWSMessage

```
public static GetGcwsMessageResponse getGCWSMessage(
    boolean useSsl,
    Environ environ,
    int code
)
```

住所検索サービスのエラーメッセージを取得します。

パラメータ

項番	名前	説明
1	useSsl	SSL を使用するかどうか
2	environ	環境設定 AdminSerivce.getUsableGeoServer で取得した環境設定を使用して下さい。
3	code	エラーコード

戻り値

結果を格納した GetGcwsMessageResponse

変更されるフィールドは下表の通りです。

項番	名前	説明
1	resultMessage	エラーメッセージ
2	exception	実行中に発生した例外