



MPACT LOCATION & ANALYTICS

**Server API
Reference Guide**

VERSION 2.0.0



TABLE OF CONTENTS

Chapter 1 MPact SERVER API Overview

Chapter 2 MPact Server API

2.1 MPact Server APIs	2-1
2.1.1 Query Filter (q)	2-1
2.2 Placeholder Types	2-2
2.3 Setting-up Hierarchies	2-2
2.4 Adding a Complex Hierarchy	2-2
2.5 Request Logging	2-5
2.6 Sample Type Query Using a Ruby Client	2-6
2.7 Sample Type Query Using a Java Client	2-8
2.8 Sample Query Workflow	2-9
2.9 Running Queries	2-14
2.9.1 Type Queries List	2-14
2.9.2 View Queries List	2-14
2.9.3 MPact APIs Using the Request Post Method	2-14
2.10 MPact Server API Examples	2-15
2.10.1 Ap - Retrieves a list of all Access Points	2-15
2.10.2 Cat - Retrieves a list of all Categories	2-16
2.10.3 Client Status - Retrieves the Client status	2-17
2.10.4 CVL - Retrieves a list of Category Values	2-18
2.10.5 Floor - Retrieves a list of all floors	2-20
2.10.6 Placeholder - Retrieves a list of all the PlaceHolders	2-22
2.10.7 Site - Retrieves a list of all the Sites	2-23
2.10.8 Tag- Retrieves a list of all the Beacons	2-25
2.10.9 TagStatus- Retrieves the status of all Beacons	2-27
2.10.10 cvlInCat - Retrieves Category Values in a Category	2-28
2.10.11 floorsInSite - Retrieves Floors in the Site	2-29
2.10.12 hierarchyInCountry - Retrieves Hierarchy for a Country	2-30
2.10.13 tagsInSite - Retrieve Beacons in Site	2-33
2.11 MPact APIs Using the Request Post Method	2-35

2.11.1 Create Site with Default Floor2-35
2.11.2 Install Beacon2-36
2.11.3 Update Status for Neighboring Beacons2-37

Appendix A Customer Support

TABLE OF CONTENTS

This chapter is organized into the following sections:

- *Using the Documentation*
- *Zebra Technologies Corporation ("Zebra") End-User Software License Agreement*

Using the Documentation

The following sections provide information about the document and notational conventions used in the guides, and provides a list of related documentation:

Document Conventions

The following conventions are used in this manual to draw your attention to important information:



NOTE: Indicates tips or special requirements.



CAUTION: Indicates conditions that can cause equipment damage or data loss.



WARNING! Indicates a condition or procedure that could result in personal injury or equipment damage.

Revision History

This guide has the following release and revision milestone history:

Release	Date	Change
1.0.1 Revision A	December, 2014	Released to reflect 1.0.1 feature baseline.
1.0.2 Revision B	March, 2015	Released to reflect 1.0.2 feature baseline.
1.1 Revision B	May, 2015	Released to reflect 1.1.0 feature baseline.
2.0.0 Revision A	February, 2016	Updated to 2.0.0 feature baseline and incorporated new Zebra templates.

Notational Conventions

The following notational conventions are used in this document:

- Italics are used to highlight specific items in the general text, and to identify chapters and sections in this and related documents
- Bullets (•) indicate:
 - lists of alternatives
 - lists of required steps that are not necessarily sequential
 - action items
- Sequential lists (those describing step-by-step procedures) appear as numbered lists

Related Documentation

MPact Location and Analytics documentation includes the following:

- *MPact Location & Analytics Deployment Guide*
- *MPact Location & Analytics Server Reference Guide*
- *MPact Location & Analytics Android Toolbox User Guide*
- *MPact Location & Analytics iOS Toolbox User Guide*
- *MPact Location & Analytics Client Software Development Kit*
- *MPact Location & Analytics Server API Reference Guide*
- *MPact Location & Analytics Hardware Installation Guide*

Zebra Technologies Corporation ("Zebra") End-User Software License Agreement

BY INSTALLING AND/OR USING THIS PRODUCT, YOU ACKNOWLEDGE THAT YOU HAVE READ THIS AGREEMENT, UNDERSTAND IT AND AGREE TO BE BOUND ITS TERMS. IF YOU DO NOT AGREE TO THE TERMS OF THIS AGREEMENT, ZEBRA IS NOT WILLING TO LICENSE THE PRODUCT TO YOU, AND YOU MUST NOT INSTALL OR USE THIS PRODUCT.

Definitions

Grant of License

Zebra Technologies Corporation ("Zebra") grants you ("Licensee" or "you") a personal, nonexclusive, nontransferable, revocable, nonassignable, limited license to use the software and documentation ("Product(s)") subject to the terms and conditions of this Agreement. You shall use the Products only for your internal business purposes, exclusively to support Zebra devices. Any use of the Products outside of the conditions set forth herein is strictly prohibited and will be deemed a breach of this Agreement resulting in immediate termination of your License. In the event of a breach of this Agreement, Zebra will be entitled to all available remedies at law or in equity (including immediate termination of the license without notice, immediate injunctive relief and repossession of all Products unless Licensee is a Federal agency of the United States Government).

You shall not distribute, sublicense, rent, loan, lease, export, re-export, resell, ship or divert or cause to be exported, re-exported, resold, shipped or diverted, directly or indirectly, the Products under this Agreement. You shall not, and shall not permit others to: (i) modify, translate, decompile, bootleg, reverse engineer, disassemble, or extract the inner workings of the Products; (ii) copy the look-and-feel or functionality of the Products; (iii) remove any proprietary notices, marks, labels, or logos from the Products; (iv) rent or transfer all or some of the Products to any other party without Zebra's prior written consent; or (v) utilize any computer software or hardware which is designed to defeat any copy protection device, should the Products be equipped with such a protection device.

Title to all copies of Products will not pass to Licensee at any time and remains vested exclusively in Zebra. All intellectual property developed, originated, or prepared by Zebra in connection with the Products remain vested exclusively in Zebra, and this Agreement does not grant to Licensee any intellectual property rights.

Portions of the Products are protected by United States patent and copyright laws, international treaty provisions, and other applicable laws. Therefore, you must treat the Products like any other copyrighted material (e.g., a book or musical recording) except that you may make one copy of the Product solely for back-up purposes. Unauthorized duplication of the Products constitutes copyright infringement, and in the United States is punishable in federal court by fine and imprisonment.

Limited Warranty

Zebra warrants for a period of ninety (90) days from your receipt of the Products to you that the Software, under normal use, will perform substantially in accordance with Zebra's published specifications for that release level of the Software. The written materials are provided "AS IS" and without warranty of any kind. Zebra's entire liability and your sole and exclusive remedy for any breach of the foregoing limited warranty will be, at Zebra's option, the provision of a downloadable patch or replacement code, or a refund of the unused portion of your bargained for contractual benefit up to the amount paid for the Products.

Disclaimer

THIS LIMITED WARRANTY IS THE ONLY WARRANTY PROVIDED BY ZEBRA, AND ZEBRA MAKES, AND YOU RECEIVE, NO OTHER WARRANTIES OF ANY KIND, WHETHER EXPRESS, IMPLIED, STATUTORY, OR IN ANY COMMUNICATION WITH YOU. ZEBRA SPECIFICALLY DISCLAIMS ANY WARRANTY INCLUDING THE IMPLIED WARRANTIES OF MERCHANTABILITY, NONINFRINGEMENT, OR FITNESS FOR A PARTICULAR PURPOSE. ZEBRA DOES NOT WARRANT THAT THE PRODUCTS WILL MEET YOUR REQUIREMENTS, OR THAT THE OPERATION OF THE PRODUCTS WILL BE UNINTERRUPTED OR ERROR FREE, OR THAT DEFECTS IN THE PRODUCTS WILL BE CORRECTED. ZEBRA MAKES NO WARRANTY WITH RESPECT TO THE

CORRECTNESS, ACCURACY, OR RELIABILITY OF THE PRODUCTS. Some jurisdictions do not allow the exclusion of implied warranties, so the above exclusion may not apply to you.

Limitation of Liability

THE TOTAL LIABILITY OF ZEBRA UNDER THIS AGREEMENT FOR DAMAGES SHALL NOT EXCEED THE FAIR MARKET VALUE OF THE PRODUCTS LICENSED UNDER THIS AGREEMENT. IN NO EVENT WILL ZEBRA BE LIABLE IN ANY WAY FOR INCIDENTAL, CONSEQUENTIAL, INDIRECT, SPECIAL OR PUNITIVE DAMAGES OF ANY NATURE, INCLUDING WITHOUT LIMITATION, LOST BUSINESS PROFITS, OR LIABILITY OR INJURY TO THIRD PERSONS, WHETHER FORESEEABLE OR NOT, REGARDLESS OF WHETHER ZEBRA HAS BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGES. Some jurisdictions do not permit limitations of liability for incidental or consequential damages, so the above exclusions may not apply to you. This Limitation of Liability provision survives the termination of this Agreement and applies notwithstanding any contrary provision in this Agreement. Licensee must bring any action under this Agreement within one (1) year after the cause of action arises.

Maintenance

Unless provided for in a separate agreement, Zebra shall not be responsible for maintenance or field service of the Products.

High Risk Activities

The Products are not fault-tolerant and are not designed, manufactured or intended for use or resale as on-line control software in hazardous environments requiring fail-safe performance, such as in the operation of nuclear facilities, aircraft navigation or communication systems, air traffic control, direct life support machines, or weapons systems, in which the failure of the Products could lead directly to death, personal injury, or severe physical or environmental damage ("High Risk Activities"). Zebra and its suppliers specifically disclaim any express or implied warranty of fitness for High Risk Activities, and if you elect to use the Products in any High Risk Activities, you agree to indemnify, defend, and hold Zebra harmless from and against any and all costs, damages, and losses related to that use.

U.S. Government

If you are acquiring the Products on behalf of any unit or agency of the U.S. Government, the following shall apply. Use, duplication, or disclosure of the Products is subject to the restrictions set forth in subparagraphs (c) (1) and (2) of the Commercial Computer Software - Restricted Rights clause at FAR 52.227-19 (JUNE 1987), if applicable, unless being provided to the Department of Defense. If being provided to the Department of Defense, use, duplication, or disclosure of the Products is subject to the restricted rights set forth in subparagraph (c) (1) (ii) of the Rights in Technical Data and Computer Software clause at DFARS 252.227-7013 (OCT 1988), if applicable. Products may or may not include a Restricted Rights notice, or other notice referring specifically to the terms and conditions of this Agreement. The terms and conditions of this Agreement shall each continue to apply, but only to the extent that such terms and conditions are not inconsistent with the rights provided to you under the aforementioned provisions of the FAR and DFARS, as applicable to the particular procuring agency and procurement transaction.

Assignment

Except as otherwise provided in this section, neither party may assign this Agreement, or any of its rights or obligations under this Agreement, without the prior written approval of the other party, which will not be unreasonably withheld. Any attempted assignment, delegation, or transfer without the necessary approval will be void. Notwithstanding the foregoing, for any Zebra acquisition, merger, consolidation, reorganization, or similar transaction, or any spin-off, divestiture, or other separation of a Zebra business, Zebra may, without the prior written consent of the other party: (i) assign its rights and obligations under this Agreement, in whole or in part, or (ii) split and assign its rights and obligations under this Agreement so as to retain the benefits of this Agreement for both Zebra and the assignee entity(ies) (and their respective Affiliates) following the split.

Governing Law

This Agreement shall be governed by the laws of the United States of America to the extent that they apply and otherwise by the laws of the State of New York without regard to its conflict of laws provisions or by the internal substantive laws of the country to which the Products is shipped if end-user customer is a sovereign governmental entity. The terms of the U.N. Convention on Contracts for the International Sale of Goods do not apply. In the event that the Uniform Computer information Transaction Act, any version of this Act, or a substantially similar law (collectively "UCITA") becomes applicable to a Party's performance under this Agreement, UCITA does not govern any aspect of this End User License Agreement or any license granted under this End-User License Agreement, or any of the parties' rights or obligations under this End User License Agreement. The governing law will be that in effect prior to the applicability of UCITA.

Compliance with Laws

Licensee will comply with all applicable laws and regulations, including export laws and regulations of the United States. Licensee will not, without the prior authorization of Zebra and the appropriate governmental authority of the United States, in any form export or re-export, sell or resell, ship or reship, or divert, through direct or indirect means, any item or technical data or direct or indirect products sold or otherwise furnished to any person within any territory for which the United States Government or any of its agencies at the time of the action, requires an export license or other governmental approval. Violation of this provision will be a material breach of this Agreement, permitting immediate termination by Zebra.

Third Party Software

The Products may contain one or more items of Third-Party Software. The terms of this Agreement govern your use of any Third-Party Software UNLESS A SEPARATE THIRD-PARTY SOFTWARE LICENSE IS INCLUDED, IN WHICH CASE YOUR USE OF THE THIRD-PARTY SOFTWARE WILL THEN BE GOVERNED BY THE SEPARATE THIRD-PARTY LICENSE.

Open Source Software

The Products may contain one or more items of Open Source Software. Open Source Software is software covered by a publicly available license governed solely under Copyright law, whereas the complete terms and obligations of such license attach to a licensee solely through the act of copying, using and/or distribution of the licensed software, such obligations often include one or more of attribution obligations, distribution obligations, copyleft obligations, and intellectual property encumbrances. The use of any Open Source Software is subject to the terms and conditions of this Agreement as well as the terms and conditions of the corresponding license of each Open Source Software package. If there is a conflict between the terms and conditions of this Agreement and the terms and conditions of the Open Source Software license, the applicable Open Source Software license will take precedence. Copies of the licenses for the included Open Source Software, if any, as well as their attributions, acknowledgements, and software information details, are provided in the electronic copy of this Agreement, which is available in the Legal Notices or README file associated with the Product. Zebra is required to reproduce the software licenses, acknowledgments and copyright notices as provided by the authors and owners, thus, all such information is provided in its native language form, without modification or translation. Depending on the license terms of the specific Open Source Software, source code may not be provided. Please reference and review the entire Open Source Software information to identify which Open Source Software packages have source code provided or available. For instructions on how to obtain a copy of any source code made publicly available by Zebra related to Open Source Software distributed by Zebra, you may send your request (including the Zebra Product name and version, along with the Open Source Software specifics) in writing to: Zebra Technologies Corporation, Open Source Software Director, Legal Department, 3 Overlook Point, Lincolnshire, IL 60069 USA.

©2015 ZIH Corp and/or its affiliates. All rights reserved. Zebra and the stylized Zebra head are trademarks of ZIH Corp., registered in many jurisdictions worldwide. All other trademarks are the property of their respective owners.

Obtaining Software Licenses

To obtain software licenses for *MPact Location & Analytics Server, Toolbox or Client Software Development Kit*, provide the following information:

- Identification
- Email address
- Payment

CHAPTER 1 MPACT SERVER API OVERVIEW

The MPact Server supports an API for retrieving and exporting product categories, category values, beacon positions, beacon configurations, hierarchal trees and beacon device ID data. The MPact Server *application programming interfaces* (API) specify how selected MPact interface components interact with the unique attributes set for it within MPact Server. A well defined API makes it more efficient to develop a program through the provision of requisite building blocks. A developer then assembles the data blocks for their specific deployment objective.

Developers can utilize a unique interface for MPact Server configuration object review and, when necessary, modification (in specific recommended areas). The MPact Server *Configuration Application Platform* provides examples of MPact user, site, floor, and beacon data objects and raw queries.

CHAPTER 2 MPACT SERVER API

2.1 MPact Server APIs

This section describes the name of the catalog, its parameters, example queries, example query results. It also provides MPact Server login instructions and both Type and View queries using an example developer tool. MPact Server RESTful APIs are implemented as JSON over HTTP using the request methods: `GET/POST/PUT/DELETE`.



NOTE: Developers can use any developer tool that allows them to interact with Web services and other Web resources for running queries and other HTTP request methods. Also, developers can run queries directly from a Web browser as long as the request is properly formed.

MPact Server APIs are organized into catalogs. For example, *Area*, *Campus*, *City*, and *Country* are all catalog types that define locations. Catalogs provide name/values contained for each type. Every catalog has fields marked by `id:true` and `q:true`. Catalogs are used to filter results of REST queries.

2.1.1 Query Filter (*q*)

Every RESTful query has a query filters known as `q` that can include an ID and a qualifier (should be TRUE). They must be URL encoded when passed to a RESTful request.

Use the following string to filter the results of a query by site identifier Site-1 and floor identifier Floor-1:

```
q=siID=Site-1,flID=Floor-1
```

The MPact Server supports the following query types:

- *Type Queries* - Allow users to browse types as independent entities.
- *View Queries* - Combine queries into a complex query.
- *Procedure queries* - Procedures are queries backed by scripting that internally invoke type or view queries.



NOTE: In this document, only Type and View queries are discussed.

2.2 Placeholder Types

Placeholders determine which values display depending on the functions used.

The MPact Server API includes placeholders for:

- **Position**
 - JSON name *phType*: "Position"

Position is associated with either Beacons or Access Points (APs).

- **Region**
 - JSON name *phType*: "Region"

Use *Region* as a placeholder type ID when a physical boundary, such a rectangle or polygon, is displayed in a query. Currently this is only applicable to a WI-FI locations.

2.3 Setting-up Hierarchies

The following are general guidelines for setting up MPact system hierarchies:

- Hierarchy Types include System, Country, CountryRegion, City, Campus, Site
- Hierarchy types can occur directly under system or under hierarchy type.
- System is always the root of the hierarchy, and has a constant "sys" as the ID.
- System type is added by default, and is created using the API.
- Hierarchy types have a catalog name called *pcid* that identifies the parent ID.
- IDs of other hierarchies can be set in the JSON.
- Floor always occurs under Site and Placeholder always occurs under Floor. They do not contain the *pcid* name.
- Refer to ["Save Site" on page 2-35](#) to add a Site directly under system.

2.4 Adding a Complex Hierarchy

The following example adds a complex hierarchy implementing JSON over HTTP using the POST request method:



NOTE: All hierarchy decisions are made during query time. During Save, add JSONs in any order. The only requirement is that *pcids* must be correct.


```

{
  "data" : [ {
    "Country" : {
      "coName" : "USA",
      "coID" : "US1391472944307",
      "pcid" : "sys"
    }
  },
  {
    "CountryRegion" : {
      "creName" : "East",
      "creID" : "East1391472944308",
      "pcid" : "US1391472944307"
    }
  },
  {
    "City" : {
      "pcid" : "East1391472944308",
      "ciName" : "New York",
      "ciID" : "New York1391472944308"
    }
  },
  {
    "Campus" : {
      "cmName" : "East Campus",
      "pcid" : "New York1391472944308",
      "cmID" : "East Campus1391472944308"
    }
  },
  {
    "Site" : {
      "siID" : "NY20",
      "siName" : "NY20",
      "pcid" : "East Campus1391472944308",
      "siLong" : "-87.15265274",
      "siLat" : "42.03794676"
    }
  },
  {
    "Floor" : {
      "siID" : "NY20",
      "flName" : "S2F1",
      "flGlobalNorth" : "0",
      "flImWidthM" : "400",
      "flPlanURL" : "/stats/services/rest/v1/bss/plan/floors/ExecutiveConferenceRoom.jpg",
      "flImHeightM" : "200",
      "flID" : "S2F1",
      "flImUnits" : "Meter",
      "flImWidthP" : "1200",
      "flImHeightP" : "779"
    }
  }
]
}

```

Posting the above JSON produced response number 200 indicating the hierarchy is created successfully. }

2 - 4 MPact Location & Analytics Server API Reference Guide

POST: /stats/services/rest/v1/feature/save/

SUCCESS: 200: Tue Sep 01 2015 16:37:55 GMT-0700 (Pacific Standard Time)

```
{
  "success" : "true",
  "ids" : [
    "US1391472944307",
    "East1391472944308",
    "New York1391472944308",
    "East Campus1391472944308",
    "NY20",
    "S2F1"
  ],
  "errors" : [
    "none",
    "none",
    "none",
    "none",
    "none",
    "none"
  ],
  "message" : "",
  "body" : {
    "queryResult" : {
      "isSuccessful" : "true"
    }
  }
}
```

2.5 Request Logging

MPact uses Tomcat as its container. Tomcat is capable of detailed request logging.

To enable request logging for the MPact Server:

1. From the CLI, change the directory to \$NUXI_BASE. (the base folder is where MPact Server is installed)

For example, `~/usr/nuxi`

2. Navigate to the `conf` folder `cd tomcat/conf_dist`
3. Use the following command: `vi server.xml`
4. Scroll down in the VI session (text editor) and edit the `AccessLogValve` section.

```
<!-- //remove these comment markers and edit the bolded entries
<Valve className="org.apache.catalina.valves.AccessLogValve" directory="/root/
usr/nuxi_logs/tomcat"

        prefix="localhost_access_log." suffix=".log"

        pattern="%{X-Forwarded-For}i | %a | %A | %l | %u | %t | %r | %s | %b
| %{User-Agent}i | %{Referer}i" />

..> //remove these comment markers
```

5. Save this file.
6. Exit VI.
7. Navigate to bin folder `cd/usr/nuxi/scripts/bin`
8. Stop the MPact Server: `./nxstats stop`
9. Start the MPact Server: `./nxstats start`

Logs will be in `$NUXI_LOGS/localhost_access_log*.log`

Logs will be in `$NUXI_LOGS/tomcat/localhost_access_log*.log`

After request logging is captured, disable logging by commenting out the valve section and restart the services.

Also, clear logs manually from the log directory to save space.

The following is a sample log statement:

```
services/rest/v1/proc/save/ss/updateClientStatus HTTP/1.1 | 401 | 63 | Java/
1.8.0 | -
- | 10.0.2.125 | 127.0.1.1 | - | - | [06/Oct/2015:09:26:59 -0700] | POST /
stats/services/rest/v1/proc/save/ss/updateClientStatus HTTP/1.1 | 401 | 63 |
Java/1.8.0 | -
{
```

2.6 Sample Type Query Using a Ruby Client

To login to the MPact Server and perform queries:

1. Using an SSH client, enter the IP address of the MPact Server in the Host Name field.



NOTE: Make sure the system has Ruby installed.

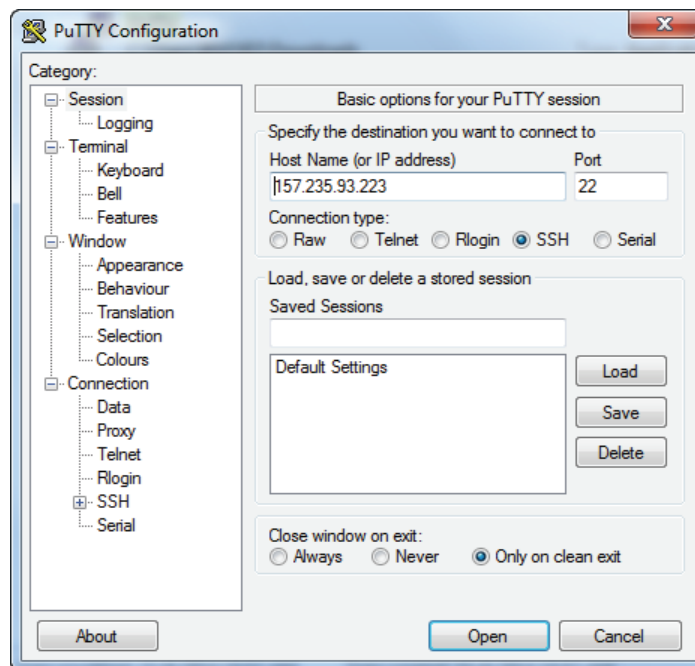


Figure 2-1 *Open Remote Session*

2. Click Open.
3. Login to the system as `root`.
4. Enter the system password.
5. In VI create a file and name it `mpact_test.rb` and paste the contents of the following Ruby code into the file:

```
require 'net/https'
URL = URL('http://127.0.0.1/stats/services/rest/v1/feature/list/
Floor?q=id%3D*')
Net::HTTP.start(URL.host, URL.port,
  :use_ssl => URL.scheme == 'https',
  :verify_mode => OpenSSL::SSL::VERIFY_NONE) do |http|
  request = Net::HTTP::Get.new URL.request_URL
  request.basic_auth 'superuser', 'motorola'
  response = http.request request # Net::HTTPResponse object
  puts response
  puts response.body
end
```

6. Run the query by entering the following commands at the prompt:

- .a vi mpact_test.rb.
- .b ruby mpact_test.rb.

The query results display in the shell.

2.7 Sample Type Query Using a Java Client

To login to the MPact Server and perform queries:

1. Create an Eclipse project and name it Mpact_RestfulClient.
2. Import the sample source file MpactRestClient.java

To run the query:

3. Build the project and run as a Java application.

The query response is displayed within Eclipse.



NOTE: Make sure the system has Java installed. To run the. Before running the Mpact_RestClient.java. sample code: 1. Download and install Eclipse from: <https://www.eclipse.org/downloads/>. 2. Download Jersey 2.13 from <https://jersey.java.net/> and follow the documentation to configure it on your system.

```
import java.io.IOException;
import javax.ws.rs.core.MediaType;
import javax.ws.rs.core.URLBuilder;
import org.apache.http.client.ClientProtocolException;
import com.sun.jersey.api.client.Client;
import com.sun.jersey.api.client.WebResource;
import com.sun.jersey.api.client.config.ClientConfig;
import com.sun.jersey.api.client.config.DefaultClientConfig;
public class MpactRestClient{
    public static void main(String[] args) throws ClientProtocolException,
    IOException {
        ClientConfig config = new DefaultClientConfig();
        Client client = Client.create(config);
        String username = "superuser";
        String password = "motorola";
        final HTTPBasicAuthFilter authFilter = new
HTTPBasicAuthFilter(username, password);
        client.addFilter(authFilter);
        client.addFilter(new LoggingFilter());
WebResource service = client.resource(URLBuilder.fromURL("http://127.0.0.1/
stats/services/rest/v1/feature/list/Floor?q=id%3D*")
.build());
System.out.println(
service.path('restPath').path('resourcePath').accept(MediaType.APPLICATION_
JSON)
.get(String.class)
);
    }
}
```

2.8 Sample Query Workflow

Refer to the following sample query work flows:

1. *List Sites* - Retrieves a list of sits (siIDs) and floors (fiIDs).
2. *List Beacons in Site* - Retrieve beacons for that siID using tagsInSite.
3. *List Clients in Site for the last hour* - Retrieve unique clients from the floor fiID) for the last one hour.

List Sites

URL: http://<MPact Server ip>/stats/services/rest/v1/feature/list/Site?q=id%3D*

Response:

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Site" : {
          "siID" : "Site11412036584350",
          "siName" : "Site1",
          "pcid" : "sys",
          "siLong" : "-54.6021",
          "siLat" : "-20.4682",
          "ts" : "1413489112378",
          "cid" : "Site11412036584350"
        }
      }, {
        "Site" : {
          "siID" : "Site21412036584352",
          "siName" : "Site2",
          "pcid" : "sys",
          "siLong" : "-121.7804924102764",
          "siLat" : "37.23527366052094",
          "ts" : "1414174154996",
          "cid" : "Site21412036584352"
        }
      }, {
        "Site" : {
          "siID" : "Site31412036584354",
          "siName" : "Site3",
          "pcid" : "sys",
          "ts" : "1412036584527",
          "cid" : "Site31412036584354"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "19",
        "totalResults" : "3"
      }
    }
  }
}
```



```

{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Site" : {
          "siID" : "Site11412036584350",
          "siName" : "Site1",
          "pcid" : "sys",
          "siLong" : "-54.6021",
          "siLat" : "-20.4682",
          "ts" : "1413489112378",
          "cid" : "Site11412036584350"
        }
      }, {
        "Site" : {
          "siID" : "Site21412036584352",
          "siName" : "Site2",
          "pcid" : "sys",
          "siLong" : "-121.7804924102764",
          "siLat" : "37.23527366052094",
          "ts" : "1414174154996",
          "cid" : "Site21412036584352"
        }
      }, {
        "Site" : {
          "siID" : "Site31412036584354",
          "siName" : "Site3",
          "pcid" : "sys",
          "ts" : "1412036584527",
          "cid" : "Site31412036584354"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "19",
        "totalResults" : "3"
      }
    }
  }
}

```

List Beacons in Site

URL: <http://<MPact Server ip>stats/services/rest/v1/feature/list/tagsInSite?q=siID%3DSite11412036584350>

Response:

```

{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Placeholder" : {
          "dvID" : "Tag_0424ae",
          "siID" : "Site11412036584350",
          "flGlobalNorth" : "0",
          "isMob" : "False",
          "flImHeightM" : "200",
          "inStatus" : "active",
          "phOff" : "1",
          "flID" : "Floor11412036584351",
          "flImUnits" : "Meter",
          "flImHeightP" : "442",
          "phType" : "Position",
          "flName" : "Floor1",
          "TagStatus_ts" : "1415660272617",
          "siLong" : "-54.6021",
          "phName" : "N-1",
          "phPos" : "(108.54,42.68)",
          "phID" : "Position_N-1_1412036791657",
          "siName" : "Site1",
          "clID" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94",
          "flPlanURL" : "/stats/services/rest/v1/bss/plan/floors/Floor-4 (1).jpg",
          "siLat" : "-20.4682",
          "taMac" : "0424ae",
          "phAntPat" : "Semi-Ellipse",
          "phDir" : "0",
          "tsBLDaysRemaining" : "336",
          "taName" : "0424ae",
          "pcid" : "sys",
          "flImWidthM" : "400",
          "Tag_ts" : "1412036863343",
          "phMajor" : "0",
          "phMinor" : "0",
          "flImWidthP" : "1106",
          "tsBL" : "51",
          "ts" : "1412036810800",
          "cid" : "Site11412036584350#Floor11412036584351#Position_N-1_1412036791657"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "9",
        "totalResults" : "1"
      }
    }
  }
}

```

List Clients in Site for the last hour

URL: http://<MPact Server ip>stats/services/rest/v1/feature/list/ClientStatus?q=id%3D*%2CbackWindow%3D3600%2Climit%3D10%2Cversions%3D1%2CflID%3DFloor11412036584351

Response:

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "ClientStatus" : {
          "siID" : "Site11412036584350",
          "phID" : "Position_N-1_1412036791657",
          "clID" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94",
          "flID" : "Floor11412036584351",
          "clPos" : "(108.54,42.68)",
          "clSource" : "BLE",
          "ts" : "1415660272617",
          "cid" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "2",
        "totalResults" : "2"
      }
    }
  }
}
```

2.9 Running Queries

This section lists Type and View queries run on the MPact Server.

2.9.1 Type Queries List

- *Ap - Retrieves a list of all Access Points* - AP stores information about access points in MPact.
- *Cat - Retrieves a list of all Categories* - This is an intersection entity between category and category value.
- *Client Status - Retrieves the Client status* - Object representing client status.
- *CVL - Retrieves a list of Category Values* - Object storing particular category values in MPact.
- *Floor - Retrieves a list of all floors* - Floor is a subset of area. A particular area may have “n” number of floors.
- *Placeholder - Retrieves a list of all the PlaceHolders* - Placeholder is related to floor. There can be different types of placeholders on the floor.
- *Site - Retrieves a list of all the Sites* - Site is a subset of a campus. A campus can have multiple sites.
- *Tag - Retrieves a list of all the Beacons* - Stores beacon information in MPact.
- *TagStatus - Retrieves the status of all Beacons* - Stores user information in MPact.



NOTE: Type queries with no filters will return a list of all results (examples are truncated). The number of “queryResult” data may not be correct.

2.9.2 View Queries List

- *cvlInCat - Retrieves Category Values in a Category*
- *floorsInSite - Retrieves Floors in the Site*
- *hierarchyInCountry - Retrieves Hierarchy for a Country*
- *tagsInSite - Retrieve Beacons in Site*

2.9.3 MPact APIs Using the Request Post Method

- *Create Site with Default Floor*
- *Install Beacon*

2.10 MPact Server API Examples

Type Queries

2.10.1 Ap - Retrieves a list of all Access Points

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/
Ap?q=id%3D*`

Parameters

Name	Description
dvId	The Id of the access point (primary key).
apName	The name of the access point.
apMAC	The factory encoded MAC address of the access point.
apIP	The IP address of the access point.
apType	The type of the access point.
apModel	The model of the access point.
radioMode	The radio mode used for communication.
siID	The site Id of the site in which this access point is installed.
phID	The placeHolder Id of the access point.

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Ap" : {
          "dvID" : "Ap_MAC2",
          "phID" : "Position_ap_p1",
          "apName" : "MAC2",
          "ts" : "1415660946068",
          "cid" : "Position_ap_p1"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "3",
        "totalResults" : "1"
      }
    }
  }
}
```

2.10.2 Cat - Retrieves a list of all Categories

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/Cat?q=id%3D*`

Parameters

Name	Description
caID	The identifier for the category (primary key).
caName	The category name.
caDesc	The category description.

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Cat" : {
          "caName" : "Cereals",
          "caID" : "Cereals",
          "ts" : "1408492242693",
          "cid" : "Cereals"
        }
      }, {
        "Cat" : {
          "caName" : "Clothes",
          "caID" : "Clothes",
          "ts" : "1408492129989",
          "cid" : "Clothes"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "2",
        "totalResults" : "13"
      }
    }
  }
}
```

2.10.3 Client Status - Retrieves the Client status

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/ClientStatus?q=id%3D*%2CbackWindow%3D3600%2Climit%3D10%2Cversions%3D1`

Parameters

Name	Description
clID	The client Id.
phID	The placeholder Id.
clPos	The position of the client.
clSource	The source of the client.
flID	The identifier for the floor (primary key).
siID	A reference to the site Id.

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "ClientStatus" : {
          "siID" : "Site11412036584350",
          "phID" : "Position_N-1_1412036791657",
          "clID" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94",
          "flID" : "Floor11412036584351",
          "clPos" : "(108.54,42.68)",
          "clSource" : "BLE",
          "ts" : "1415660272617",
          "cid" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94"
        }
      }, {
        "ClientStatus" : {
          "siID" : "Site21412036584352",
          "phID" : "Position_V2_1412037812236",
          "clID" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94",
          "flID" : "Floor21412036584353",
          "clPos" : "(41.57,5.43)",
          "clSource" : "BLE",
          "ts" : "1415660764620",
          "cid" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94"
        }
      }
    ]
  }
}
```

```

    } ]
  },
  "queryResult" : {
    "isSuccessful" : "true",
    "success" : {
      "missRatio" : "0.0",
      "timeElapsedMsec" : "2",
      "totalResults" : "2"
    }
  }
}
}
}
}

```

2.10.4 CVL - Retrieves a list of Category Values

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/Cv1?q=id%3D*`

Parameters

Name	Description
cvid	The identifier of the CategoryValue (primary key).
cvName	The name of the CategoryValue.
cvDesc	The description of the CategoryValue.

Http Response

```

{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Cvl" : {
          "cvID" : "Accessories",
          "cvName" : "Accessories",
          "ts" : "1408492386348",
          "cid" : "Accessories"
        }
      }, {
        "Cvl" : {
          "cvID" : "Beverages",
          "cvDesc" : "Soda and juice",
          "cvName" : "Beverages",
          "ts" : "1408579975823",
          "cid" : "Beverages"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "2",
        "totalResults" : "26"
      }
    }
  }
}

```

2.10.5 Floor - Retrieves a list of all floors

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/
Floor?q=id%3D*`

Parameters

Name	Description
flID	The identifier for the floor (primary key).
flName	The name of the floor.
flDesc	The description of the floor.
flGlobalNorth	The global north direction of the floor.
flLMWidthP	The floor width in pixels.
flLHeightP	The floor height in pixels.
flLWidthM	The floor width in meters.
flLHeightM	The floor height in meters.
flLMUnits	The floor measurement units.
flLSize	The floor size.
flPlanUrl	The URI to the floor plan.
siId	A reference to the site Id.
ArID	A reference to the area Id.

Http Response

```

{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Floor" : {
          "siID" : "1st Floor1406650194555",
          "flName" : "Demo Area",
          "flGlobalNorth" : "0",
          "flImWidthM" : "400",
          "flPlanURL" : "/stats/services/rest/v1/bss/plan/floors/moto_store.png",
          "flImHeightM" : "400",
          "flID" : "1st Floor1406650194556",
          "flImUnits" : "Meter",
          "flImWidthP" : "708",
          "flImHeightP" : "378",
          "ts" : "1413836326230",
          "cid" : "1st Floor1406650194556"
          "cid" : "Beverages"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "1",
        "totalResults" : "2"
      }
    }
  }
}

```

2.10.6 Placeholder - Retrieves a list of all the PlaceHolders

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/Placeholder?q=id%3D**`

Parameters

Name	Description
phID	The identifier of the PlaceholderCat (primary key).
cvid	A reference to the category Value Id.

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Placeholder" : {
          "siID" : "1st Floor1406650194555",
          "phPos" : "(165.62,150.85)",
          "phID" : "Position_1406661431860_1406661431860",
          "isMob" : "False",
          "phOff" : "1",
          "flID" : "2nd Floor1406650194557",
          "phAntPat" : "Semi-Ellipse",
          "phDir" : "0",
          "phDesc" : "Book and author promotion",
          "phType" : "Position",
          "phMajor" : "0",
          "phMinor" : "0",
          "phName" : "1406661431860",
          "ts" : "1406664523908",
          "cid" : "Position_1406661431860_1406661431860"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "4",
        "totalResults" : "39"
      }
    }
  }
}
```

2.10.7 Site - Retrieves a list of all the Sites

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/
Site?q=id%3D*`

Parameters

Name	Description
siID	The site Id in which this access point is installed.
siName	The name of the site.
siDesc	The description of the site.
siLoc	The location of the site.
siLat	The latitude of the site.
siLong	The longitude of the site.
coID	A reference to the country Id.
reID	A reference to the region Id.
ciID	A reference to the city Id.
pcID	This is the immediate parent Id. In this case, it refers to the Site within the Campus.

Http Response

```

{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Site" : {
          "siID" : "Site11412036584350",
          "siName" : "Site1",
          "pcid" : "sys",
          "siLong" : "-54.6021",
          "siLat" : "-20.4682",
          "ts" : "1413489112378",
          "cid" : "Site11412036584350"
        }
      }, {
        "Site" : {
          "siID" : "Site21412036584352",
          "siName" : "Site2",
          "pcid" : "sys",
          "siLong" : "-121.7804924102764",
          "siLat" : "37.23527366052094",
          "ts" : "1414174154996",
          "cid" : "Site21412036584352"
        }
      }, {
        "Site" : {
          "siID" : "Site31412036584354",
          "siName" : "Site3",
          "pcid" : "sys",
          "ts" : "1412036584527",
          "cid" : "Site31412036584354"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "19",

```

```
        "totalResults" : "3"  
    }  
  }  
}
```

2.10.8 Tag- Retrieves a list of all the Beacons

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/Tag?q=id%3D*`

Parameters

Name	Description
dvID	The beacon identifier (primary key).
taName	The beacon name.
taDesc	The description of the beacon.
inStatus	The beacon install status, either active or inactive.
phId	Reference to the placeHolder Id.

Http Response

```

{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Tag" : {
          "dvID" : "Tag_4862c0",
          "taName" : "4862c0",
          "phID" : "Position_1408480973216_1408480973216",
          "inStatus" : "active",
          "taMac" : "4862c0",
          "ts" : "1414431511100",
          "cid" : "Position_1408480973216_1408480973216"
        }
      }, {
        "Tag" : {
          "dvID" : "Tag_485f7d",
          "taName" : "485f7d",
          "phID" : "Position_1408480973224_1408480973225_1",
          "inStatus" : "active",
          "taMac" : "485f7d",
          "ts" : "1414431576150",
          "cid" : "Position_1408480973224_1408480973225_1"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "4",
        "totalResults" : "39"
      }
    }
  }
}

```


2.10.9 TagStatus- Retrieves the status of all Beacons

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/TagStatus?q=id%3D**`

Parameters

Name	Description
dvID	The beacon identifier (primary key).
tsBl	The beacon battery life.
cLID	The client Id.

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "TagStatus" : {
          "tsBLDaysRemaining" : "7",
          "dvID" : "Tag_02c1b6",
          "clID" : "superuser",
          "tsBL" : "9",
          "ts" : "1413848220845",
          "cid" : "Tag_02c1b6"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "5",
        "totalResults" : "52"
      }
    }
  }
}
```

View Queries

2.10.10 cvlInCat - Retrieves Category Values in a Category

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/cvlInCat?q=caID%3DCat1`

Parameters

Name	Description
caid	The area identifier (primary key).

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Cv1" : {
          "caID" : "Cat1",
          "caName" : "Cat1-name",
          "caDesc" : "Cat1-desc",
          "cvDesc" : "Cv11-desc",
          "cvID" : "Cv11",
          "cvName" : "Cv11-name",
          "ts" : "1389420484297",
          "cid" : "Cat1#Cv11"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "9",
        "totalResults" : "1"
      }
    }
  }
}
```

2.10.11 floorsInSite - Retrieves Floors in the Site

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/floorsInSite?q=siID%3DSite-1`

Parameters

Name	Description
siID	The Id of site where the access point resides.

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Floor" : {
          "siID" : "Site-1",
          "siLat" : "37.7750 N",
          "siName" : "Site-1",
          "pcid" : "Campus-1",
          "siLong" : "122.4183 W",
          "siDesc" : "Site-1",
          "arID" : "Area-1",
          "arName" : "Area-1",
          "arDesc" : "Area-1",
          "flImHeightM" : "9",
          "flImHeightP" : "995",
          "flGlobalNorth" : "320",
          "flImWidthM" : "18",
          "flName" : "Floor-1",
          "flImWidthP" : "1858",
          "flDesc" : "Floor-1",
          "flID" : "Floor-1",
          "flPlanURL" : "/stats/services/rest/v1/bss/Floor-1.jpg",
          "flImUnits" : "pixels",
          "ts" : "1389420484285",
          "cid" : "Site-1#Area-1#Floor-1"
        }
      } ]
    },
    "queryResult" : {
      "isSuccessful" : "true",
      "success" : {
```

```

    "missRatio" : "0.0",
    "timeElapsedMsec" : "18",
    "totalResults" : "1"
  }
}
}
}

```

2.10.12 hierarchyInCountry - Retrieves Hierarchy for a Country

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/list/hierarchyInCountry?q=coID%3DCountry-1`

Parameters

Name	Description
coID	The Country Id.
creID	The Country region Id.
ciID	The City Id.
cmID	The Campus Id.
siID	The Id of site where the access point resides.
flID	The Floor Id.

Http Response

```

{
  "body" : {
    "queryData" : {
      "children" : [ {
        "pcid" : "sys",
        "coID" : "Country-1",
        "desc" : "Country-1",
        "name" : "Country-1",
        "type" : "Country",
        "children" : [ {
          "desc" : "CountryRegion-1",
          "pcid" : "Country-1",
          "creID" : "CountryRegion-1",
          "name" : "CountryRegion-1",
          "type" : "CountryRegion",
          "children" : [ {
            "name" : "City-1",
            "desc" : "City-1",

```

```

"pcid" : "CountryRegion-1",
"ciID" : "City-1",
"type" : "City",
"children" : [ {
  "cmID" : "Campus-1",
  "desc" : "Campus-1",
  "pcid" : "City-1",
  "name" : "Campus-1",
  "type" : "Campus",
  "children" : [ {
    "siID" : "Site-1",
    "siLat" : "37.7750 N",
    "name" : "Site-1",
    "pcid" : "Campus-1",
    "siLong" : "122.4183 W",
    "desc" : "Site-1",
    "type" : "Site",
    "children" : [ {
      "siID" : "Site-1",
      "arID" : "Area-1",
      "name" : "Area-1",
      "desc" : "Area-1",
      "type" : "Area",
      "children" : [ {
        "arID" : "Area-1",
        "siID" : "Site-1",
        "flImHeightM" : "9",
        "flImHeightP" : "995",
        "flGlobalNorth" : "320",
        "flImWidthM" : "18",
        "name" : "Floor-1",
        "flImWidthP" : "1858",
        "desc" : "Floor-1",
        "flID" : "Floor-1",
        "flPlanURL" : "/stats/services/rest/v1/bss/Floor-1.jpg",
        "flImUnits" : "pixels",
        "type" : "Floor",
        "ts" : "1389420484285",
        "cid" : "Floor-1"
      } ],
      "ts" : "1389420484285",
      "cid" : "Area-1"
    } ],
    "ts" : "1389420484285",
    "cid" : "Area-1"
  } ],
  "ts" : "1389420484285",
  "cid" : "Area-1"
} ],
"ts" : "1389420484285",
"cid" : "Area-1"

```

```

        } ],
        "ts" : "1389420484283",
        "cid" : "Site-1"
    } ],
    "ts" : "1389420484282",
    "cid" : "Campus-1"
} ],
"ts" : "1389420484281",
"cid" : "City-1"
} ],
"ts" : "1389420484280",
"cid" : "CountryRegion-1"
} ],
"ts" : "1389420484278",
"cid" : "Country-1"
} ]
},
"queryResult" : {
    "isSuccessful" : "true",
    "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "52",
        "totalResults" : "1"
    }
}
}
}
}

```

2.10.13 tagsInSite - Retrieve Beacons in Site

Example URL: `http://<MPact Server ip>stats/services/rest/v1/feature/list/tagsInSite?q=siID%3DSite11412036584350`

Parameters

Name	Description
arID	The area identifier (primary key)
siID	The Id of site where an access point resides.
flID	The floor Id.
tagID	The beacon Id.

Http Response

```
{
  "body" : {
    "queryData" : {
      "data" : [ {
        "Placeholder" : {
          "dvID" : "Tag_0424ae",
          "siID" : "Site11412036584350",
          "flGlobalNorth" : "0",
          "isMob" : "False",
          "flImHeightM" : "200",
          "inStatus" : "active",
          "phOff" : "1",
          "flID" : "Floor11412036584351",
          "flImUnits" : "Meter",
          "flImHeightP" : "442",
          "phType" : "Position",
          "flName" : "Floor1",
          "TagStatus_ts" : "1415660272617",
          "siLong" : "-54.6021",
          "phName" : "N-1",
          "phPos" : "(108.54,42.68)",
          "phID" : "Position_N-1_1412036791657",
          "siName" : "Site1",
          "clID" : "big ipadfggytrff-A9C6CFEA-0EC3-47E5-80BD-4F7D32081A94",
          "flPlanURL" : "/stats/services/rest/v1/bss/plan/floors/Floor-4 (1).jpg",
          "siLat" : "-20.4682",
          "taMac" : "0424ae",
          "phAntPat" : "Semi-Ellipse",
```

```

        "phDir" : "0",
        "tsBLDaysRemaining" : "336",
        "taName" : "0424ae",
        "pcid" : "sys",
        "flImWidthM" : "400",
        "Tag_ts" : "1412036863343",
        "phMajor" : "0",
        "phMinor" : "0",
        "flImWidthP" : "1106",
        "tsBL" : "51",
        "ts" : "1412036810800",
        "cid" : "Site11412036584350#Floor11412036584351#Position_N-1_1412036791657"
    }
} ]
},
"queryResult" : {
    "isSuccessful" : "true",
    "success" : {
        "missRatio" : "0.0",
        "timeElapsedMsec" : "9",
        "totalResults" : "1"
    }
}
}
}
}

```


2.11 MPact APIs Using the Request Post Method

2.11.1 Create Site with Default Floor

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/feature/save/`

Use the following string to create a site with a default floor plan:

Save Site

```
{
  "data" : [ {
    "Site" : {
      "siDesc" : "CAUS1_10_",
      "siID" : "CAUS1_10_",
      "siName" : "CAUS1_10_",
      "pcid" : "sys",
      "siLat": "10.6701", "siLong": "-6.8225"
    }
  }
]
```

Save Floor

```
{
  "data" : [ {
    "Floor" : {
      "siID" : " CAUS1_10_",
      "flName" : "Floor 1",
      "flDesc" : "Floor 1",
      "flImWidthM" : "-3.4028234663852886E38",
      "flPlanURL" : "/stats/services/rest/v1/bss/plan/floors/default.png",
      "flImHeightM" : "-3.4028234663852886E38",
      "flID" : " CAUS1_10_Floor 1_1",
      "flImUnits" : "Meter",
      "flImHeightP" : "630",
      "flImWidthP" : "1183"
    }
  }
]
```

Save Placeholder

```

{
  "data" : [ {
    "Placeholder" : {
      "phDesc" : "ACCESS_POINT",
      "phType" : "Position",
      "siID" : "CAUS1_10_",
      "phPos" : "(35.3085106382979,2.6780794021930774)",
      "plannedDvMac" : "00:23:68:41:dc:50",
      "phID" : "Position_00:23:68:41:dc:50",
      "phName" : "00-23-68-41-dc-50",
      "flID" : " CAUS1_10_Floor 1_1",
      "ts" : "1435178098988"
    }
  }
]
}

```

2.11.2 Install Beacon

Example URL: `http://<MPact Server ip>/stats/services/rest/v1/proc/save/installTag`

Use the following string to install a beacon:

Install Beacon

```

{
  "taName": "41dc50", //last 3 bytes of MAC
  "taMac": "41dc50", //last 3 bytes of MAC
  "dvModel": "ATLS-T1B",
  "dvSkus": "MPACT-T1B10-000-WR",
  "inStatus": "installed",
  "phID": "Position_00:23:68:41:dc:50"
}

```

2.11.3 Update Status for Neighboring Beacons

Example URL: `http://<MPact Server ip> /stats/services/rest/v1/proc/save/updateStatusForNeighboringTags`

Use the following string to update the status of neighboring beacons:

```
{
  "clID": "32fab6",
  "Tags": [
    {
      "taName": "<3 bytes mac>",
      "tsBL": "55" //battery life percentage
    },
    {
      "taName": "<3 bytes mac>",
      "tsBL": "74"
    },
    {
      "taName": "<3 bytes mac>",
      "tsBL": "100"
    }
  ]
}
```


APPENDIX CUSTOMER SUPPORT

If you have a problem with your equipment, contact Support for your region. Support and issue resolution is provided for products under warranty or that are covered by a services agreement. Contact information and Web self-service is available by visiting www.zebra.com/support.

When contacting Support, please provide the following information:

- *MAC ID of the unit*
- *Model number or product name*
- *Software type and version number*

Support responds to calls by email or telephone within the time limits set forth in support agreements. If you purchased your product from a business partner, contact that business partner for support.

Customer Support Web Site

The support site, located at www.zebra.com/support provides information and online assistance including developer tools, software downloads, product manuals, support contact information and online repair requests.

Manuals

To see manuals, go to: www.zebra.com/support.



ZEBRA

MN002605A01 Revision A February, 2016