



This document holds basic data, links, and samples for the MatrixMaxx AMS/CRM's webservices API. The target audience for this document is internal Matrix staff as well as 3rd party system vendors, including both technical and non-technical team members.

Introduction

The basic MatrixMaxx® API allows access to data inside MatrixMaxx®. [REST \(Representational State Transfer\)](#) is followed as a principle. RESTful API services will return JSON formatted results. Searches are performed with MQL (minimal query language) encoded into the URL.

- We start with read-only access, and we generally want to confirm the data that will be pulled so that we can 'lock down' data access that is not required by any 3rd party
- If there is a desire for a 3rd party to **write** to the MatrixMaxx database, this must be discussed and scoped so that we know exactly what needs to be opened up.

Versions and Release Notes

- 21.3 - Sept 2021 - REST handler upgrades
- 21.1 - April 2021 - Initial components of the High Level (HL) API released into production

Getting Set Up

Authentication/Access Control

We typically grant access to our API by whitelisting your IPs (both DEV and LIVE IPs as needed) so that your servers may call our API. If this is not possible or practical with your setup, the MatrixMaxx AMS team will discuss alternate approaches/solutions

Rate Limits

We currently do not enforce a hard rate limit, as we are typically involved in reviewing/optimizing the calls coming into our system. To ensure optimal server performance, we recommend that you do not exceed 5 concurrent requests.

Endpoints

3rd parties will be given a URL pattern along the lines of <https://xxx.xxx.net/protectedRest/>

There will be different URLs for live/production and dev. Throughout this document, this is referred to as the 'baseURL'

Available Services

The following types of objects may be queried. To learn more about the attributes of an object, use the [Metadata](#) service.

- Individual
- Company
- User
- Committee
- Meeting
- Product
- CalendarItem
- Fee (meeting fees)
- Event
- Exhibit
- MailingList (this is poorly named. Some of the lists are just lists of individuals that are NOT connected to a mailing list).
- Tradeshow
- Session (meeting sessions)
- Attendee (meeting attendee)
- Comment (comments can be attached to individuals, companies, products, etc.)
- ObjectList (in theory, a group of any type of object, but so far only implemented as groups of companies)

Metadata

The resource at {baseURL}/Metadata/{objectType} will show information about each attribute belonging to that class. The name of the default WebSelect used for an attribute can be found in the webselectDefaultName field of the metadata for that attribute. For example the companyStatusId attribute of Company uses a WebSelect named CompanyStatus. NOTE: The examples provided here are just examples. Many of the attributes and associated classes (assocs) for an object are generic across the product, but each client's database also has its own custom attributes and assocs specific to it.

Example: examining 'Company' metadata

https://{baseURL}/protectedRest/Metadata/Company

```

"congressOfficerFor": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"howHeardAbout": {"fromClassName": "Company", "cardinality": "MM",
"toClassName": "HowHeardAbout", "type": "assoc"},
"exhibitions": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Exhibit", "type": "assoc"},
"diversityCertifications": {"fromClassName": "Company", "cardinality":
"1M", "toClassName": "DiversityCertification", "type": "assoc"},
"DateJoined": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": true},
"congressSenators": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"ParentCompanyNum": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"riskRating": {"webselectDefaultName": "RiskRating", "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"ParentCompanyName": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"RealActiveMembershipType": {"webselectDefaultName": null, "isDerived":
true, "type": "attribute", "attributeType": null, "isWebselect": true},
"aliases": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"isNonmember": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": true},
"congressRep": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"staffing": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": false},
"associations": {"fromClassName": "Company", "cardinality": "MM",
"toClassName": "Association", "type": "assoc"},
"activities": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Activity", "type": "assoc"},
"additionalRosterSlots": {"webselectDefaultName": null, "isDerived":
false, "type": "attribute", "attributeType": "int", "isWebselect": true},
"NameForSearch": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"hasSubsidiaries": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"congressSenator1": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"congressSenator2": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"ticker": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},

```

```
"primaryEmail": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": false},
"isPaid": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": false},
"name": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": false},
"countries": {"fromClassName": "Company", "cardinality": "MM",
"toClassName": "Country", "type": "assoc"},
"otherProductCategory": {"webselectDefaultName": null, "isDerived":
false, "type": "attribute", "attributeType": "string", "isWebselect":
true},
"matchString": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"hasWorkplaceFun": {"webselectDefaultName": "YesNo", "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"communications": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "MemberCommunication", "type": "assoc"},
"companyStatusId": {"webselectDefaultName": "CompanyStatus", "isDerived":
false, "type": "attribute", "attributeType": "id", "isWebselect": true},
"issuesOfImportance": {"fromClassName": "Company", "cardinality": "MM",
"toClassName": "IssueOfImportance", "type": "assoc"},
"PreferredTollFree": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"parentRelationships": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Relationship", "type": "assoc"},
"recordOrigin": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"logo": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": false},
"isMembershipInherited": {"webselectDefaultName": null, "isDerived":
true, "type": "attribute", "attributeType": null, "isWebselect": true},
"minorityOwned": {"webselectDefaultName": "YesNo", "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"comments": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Comment", "type": "assoc"},
"isPubliclyTraded": {"webselectDefaultName": "YesNo", "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"NameAndSubDiv": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"meetingHotels": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "MeetingHotel", "type": "assoc"},
"directoryContact": {"fromClassName": "Company", "cardinality": "M1",
"toClassName": "Individual", "type": "assoc"},
"article": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": false},
"Dues": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": false},
```

```
"RealMembershipStatus": {"webselectDefaultName": "MembershipStatus",
"isDerived": true, "type": "attribute", "attributeType": null,
"isWebselect": true},
"socialNetworking": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "SocialNetworking", "type": "assoc"},
"grossRevenue": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": false},
"prospectStatus": {"webselectDefaultName": "ProspectStatus", "isDerived":
false, "type": "attribute", "attributeType": "int", "isWebselect": true},
"womanOwned": {"webselectDefaultName": "YesNo", "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"taxId": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "int", "isWebselect": false},
"services": {"fromClassName": "Company", "cardinality": "MM",
"toClassName": "Service", "type": "assoc"},
"owner": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"PreferredFax": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": true},
"hasSocialCommittee": {"webselectDefaultName": "YesNo", "isDerived":
false, "type": "attribute", "attributeType": "string", "isWebselect":
true},
"silent": {"webselectDefaultName": "YesNo", "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"PreferredPhone": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"enteredBy": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": false},
"SubDivText": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": true},
"isActive": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": true},
"RealMembershipType": {"webselectDefaultName": "MembershipType",
"isDerived": true, "type": "attribute", "attributeType": null,
"isWebselect": true},
"website": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "WebAddress", "type": "assoc"},
"paidDues": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": false},
"hotel": {"fromClassName": "Company", "cardinality": "11", "toClassName":
"Hotel", "type": "assoc"},
"lastUpdated": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "datetime", "isWebselect": false},
"lastUpdatedBy": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": false},
"name2": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
```

```
"criticalNotes": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"totalEmployees": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"companyStruct": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"riskAssessment": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"numFilledSlots": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"MemberSegment": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"yearEstablished": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"congresspersonFor": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"keyContact": {"fromClassName": "Company", "cardinality": "M1",
"toClassName": "Individual", "type": "assoc"},
"annualSales": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"invoices": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Invoice", "type": "assoc"},
"states": {"fromClassName": "Company", "cardinality": "MM",
"toClassName": "StateProvince", "type": "assoc"},
"billingContact": {"fromClassName": "Company", "cardinality": "M1",
"toClassName": "Individual", "type": "assoc"},
"id": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "id", "isWebselect": false},
"MemberSince": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": true},
"alphaSort": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"isComplementary": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true},
"numFunEvents": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "int", "isWebselect": true},
"congressionalDistrict": {"webselectDefaultName": null, "isDerived":
false, "type": "attribute", "attributeType": "string", "isWebselect":
true},
"description": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": false},
"companyTypeId": {"webselectDefaultName": "CompanyType", "isDerived":
false, "type": "attribute", "attributeType": "int", "isWebselect": true},
"companyNum": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "int", "isWebselect": true},
"applications": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Application", "type": "assoc"},
```

```

"membership": {"fromClassName": "Company", "cardinality": "11",
"toClassName": "Membership", "type": "assoc"},
"childRelationships": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Relationship", "type": "assoc"},
"brands": {"webselectDefaultName": null, "isDerived": false, "type":
"attribute", "attributeType": "string", "isWebselect": true},
"highResImage": {"fromClassName": "Company", "cardinality": "11",
"toClassName": "File", "type": "assoc"},
"sizeCode": {"fromClassName": "Company", "cardinality": "M1",
"toClassName": "SizeCode", "type": "assoc"},
"branches": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Branch", "type": "assoc"},
"DateCanceled": {"webselectDefaultName": null, "isDerived": true, "type":
"attribute", "attributeType": null, "isWebselect": true},
"isActiveParent": {"webselectDefaultName": null, "isDerived": true,
"type": "attribute", "attributeType": null, "isWebselect": true},
"competitors": {"fromClassName": "Company", "cardinality": "1M",
"toClassName": "Competition", "type": "assoc"},
"alternateContacts": {"fromClassName": "Company", "cardinality": "MM",
"toClassName": "Individual", "type": "assoc"},
"howHeardAboutOther": {"webselectDefaultName": null, "isDerived": false,
"type": "attribute", "attributeType": "string", "isWebselect": true}}

```

WebSelects

In MatrixMaxx® a list of options like a lookup table is called WebSelect. A list of all webselects can be gotten at {baseUrl}/WebSelect/. Values for each WebSelect can be found at <https://{baseUrl}/protectedRest/WebSelect/{webSelectName}>

Example: the Month webselect

```
https://{baseUrl}/protectedRest/WebSelect/Month
```

```

[{"id": 1, "description": "January"},
{"id": 2, "description": "February"},
{"id": 3, "description": "March"},
{"id": 4, "description": "April"},
{"id": 5, "description": "May"},
{"id": 6, "description": "June"},
{"id": 7, "description": "July"},
{"id": 8, "description": "August"},
{"id": 9, "description": "September"},
{"id": 10, "description": "October"},

```

```
[{"id": 11, "description": "November"}, {"id": 12, "description": "December"}]
```

Example: the MembershipType webselect

<https://{baseUrl}/protectedRest/WebSelect/MembershipType>

```
[{"id": 1, "description": "Member"}, {"id": 2, "description": "Supplier"}, {"id": 7, "description": "NonMember"}, {"id": 8, "description": "Guest"}, {"id": 9, "description": "Individual Member"}]
```

Accessing a Single Resource: information about a particular object based on ID

Accessing a single resource will usually follow the pattern of <https://{baseUrl}/protectedRest/{objectType}/{id}>. The class will be something like User, and the id will be the MatrixMaxx id of the record, which is usually a hex string that can normally be found in the MatrixMaxx URL of the object.

For example, the info about an individual with MatrixMaxx ID 15E00090F65 would be available at <https://{baseUrl}/protectedRest/Individual/15E00090F65> For this Individual, their corresponding MatrixMaxx view page would be

<https://{baseUrl}/membership/IndividualFormPrivate/view?id=15E00090F65>

As you can see in the above example, the Individual's id is consistent across both URLs. An object id can be found in the URL on many MatrixMaxx pages, and you can use the corresponding id and object type with the API to access the data for any resource in the database.

Another example: You can access the data for Meeting 1A6FE000002D5 by sending a request to <https://{baseUrl}/protectedRest/Meeting/1A6FE000002D5>

The format of the results will not change, though the order of attributes in the resulting JSON are arbitrary and may change) Attributes may be added to a result without notice, but we generally don't delete attributes from the results unless we're sure they are not being used.

If an attribute contains a List of objects, that will normally be represented by a list of IDs and API URIs that link to those resources. If more details about those objects are needed they can be obtained with a separate request to that URI. Example: "events": [{"uri": "/protectedRest/Event/2B81B0000019A", "id": "2B81B0000019A"}, {"uri": "/protectedRest/Event/1A72D000001CA", "id": "1A72D000001CA"}]

If you want to get data from further nested resources in a single call without having to make an additional call, you can pass an argument "expand" to the resource request with the names of fields that you want to expand. Multiple fields can be passed through this field if they are comma separated. Example to get mainAddress and directoryContact from Company in a single request:

/protectedRest/Meeting/1A6FE000002D5?expand=mainAddress%2CdirectoryContact

Example: Meeting 1A6FE000002D5

https://{baseUrl}/protectedRest/Meeting/1A6FE000002D5

```
{
  "cancellationPolicy": "Cancellation Policy: 50% until week of the
meeting, then no refunds.",
  "calendarLinkType": 4,
  "staffNotificationRevisitedEmail": null,
  "groupRegistrationOn": "Yes",
  "hotelBlurb": null,
  "registrationEnds": null,
  "committee": null,
  "housingInfo": null,
  "sessionsCalendar": {
    "uri": "/protectedRest/Calendar/1A6FE000002ED",
    "id": "1A6FE000002ED"
  },
  "meetingHotels": null,
  "thankYouText": "Thank You for registering! Hope you have FUN at our
meeting!",
  "DateString": "Wednesday, October 1 (9am) - Sunday, October 5, 2014 (12am
US/Eastern)",
  "inSiteBodyTextColor": null,
  "location": {
    "uri": "/protectedRest/Address/1A6FE000002D8",
    "id": "1A6FE000002D8"
  },
  "tradeShow": null,
  "paymentInstructions": null,
  "staffCancellationEmail": null,
  "useBadges": "Yes",
  "reviewPageTopText": "Below are the registration details you have chosen.
Please review then, then proceed to the next step for payment.",
  "showPublicSchedule": null,
  "banner": {
    "uri": "/protectedRest/File/1A6FE000002D9",
    "id": "1A6FE000002D9"
  },
  "categories": null,
  "name": "2014 Annual Meeting",
}
```

```
"questionsPageTopText": "Use this page to answer registration questions
for your attendees.",
"committees": null,
"inSiteLinkTextColor": null,
"BudgetedRevenue": "0",
"NumUnapprovedRegistrations": 5,
"inSiteAccentColor": null,
"highlightedTabTextColor": "#ffffff",
"allowSpouseReg": null,
"inSiteTemplate": null,
"sidebarColor": null,
"calendarBlurb": null,
"IsAcceptingOnlineRegistrations": 1,
"DateRegistrationEnds": "2014/09/30 23:59:00 GMT-4",
"cmsImage": null,
"inSiteTabTitleColor": null,
"fontName": "sans",
"inSiteHighlightedTabTextColor": null,
"earlybirdCutoff": null,
"isWebinar": null,
"linkTextColor": "#160996",
"sponsorships": null,
"logo": null,
"calendarItem": {"uri": "/protectedRest/CalendarItem/1A6FE000002EC",
                  "id": "1A6FE000002EC"},
"feesPageTopText": "Use this page to select registration fees for the
attendees.",
"eventId": null,
"paymentPageTopText": null,
"comments": null,
"tabTitleColor": "#ffffff",
"HasFeeForNonmembers": null,
"inSiteRightNavTextColor": null,
"allowPlaceholders": "Yes",
"_budgetedRevenue": 0,
"attire": "Business Casual",
"DateRegistrationStarts": "2012/04/01",
"paymentAddress": null,
"reportingType": null,
"fontBodySize": "medium",
"previousMeeting": null,
"housingPageTopText": null,
"registrationQuestions": null,
"inSiteHighlightedTabColor": null,
"badgesPageTopText": "Use this page to fill in badge information for your
attendees. We ♥ badges!",
"inSiteFontTitleSize": null,
```

```
"audienceDescription": null,
"DateEarlyRegistrationDiscountEnds": "",
"highlightedTabColor": "#83cdd7",
"micrositePath": "2014-annual-meeting",
"continuingEdNumber": null,
"hostName": null,
"fontTitleSize": "medium",
"HasFeeForGuests": 0,
"micrositeTemplate": "Responsive",
"fees": [{"uri": "/protectedRest/Fee/1A6FE000002D6", "id":
"1A6FE000002D6"},
        {"uri": "/protectedRest/Fee/1A72F00000010", "id":
"1A72F00000010"},
        {"uri": "/protectedRest/Fee/1FC1100000399", "id":
"1FC1100000399"},
        {"uri": "/protectedRest/Fee/2023C00000001", "id":
"2023C00000001"}],
"navbarColor": "#0390a4",
"anticipatedRegistrations": 1000,
"bgcolor": "#dfeaeb",
"registrationStarts": null,
"HasFeeForSpouses": 0,
"type": 7,
"files": [{"uri": "/protectedRest/File/1A6FE000002DA", "id":
"1A6FE000002DA"},
        {"uri": "/protectedRest/File/1A6FE000002DB", "id":
"1A6FE000002DB"},
        {"uri": "/protectedRest/File/1A6FE000002DC", "id":
"1A6FE000002DC"},
        {"uri": "/protectedRest/File/1A6FE000002DD", "id":
"1A6FE000002DD"},
        {"uri": "/protectedRest/File/1A6FE000002DE", "id":
"1A6FE000002DE"}],
"selectSessionsOnRegForm": null,
"HasEventsForRegForm": null,
"attendeesPageTopText": null,
"showPublicSessionSearch": null,
"relatedUrls": null,
"feeDateRanges": [{"uri": "/protectedRest/FeeDateRange/1A6FE000002D7",
"id": "1A6FE000002D7"},
        {"uri": "/protectedRest/FeeDateRange/1A72F00000092",
"id": "1A72F00000092"}],
"rightNavTextColor": "#960909",
"inSiteHeadlineColor": null,
"clients": null,
"IsClosed": 0,
"bodyTextColor": "#333333",
```

```
"staffNotificationEmail": null,
"myMeeting": null,
"headlineColor": "#730996",
"inSiteFontBodySize": null,
"tabs": [{"uri": "/protectedRest/MicrositeTab/1A6FE000002DF", "id":
"1A6FE000002DF"},
        {"uri": "/protectedRest/MicrositeTab/1A6FE000002E2", "id":
"1A6FE000002E2"},
        {"uri": "/protectedRest/MicrositeTab/1A6FE000002E4", "id":
"1A6FE000002E4"},
        {"uri": "/protectedRest/MicrositeTab/1A6FE000002E6", "id":
"1A6FE000002E6"},
        {"uri": "/protectedRest/MicrositeTab/1A6FE000002E8", "id":
"1A6FE000002E8"},
        {"uri": "/protectedRest/MicrositeTab/1A6FE000002EA", "id":
"1A6FE000002EA"}],
"inSiteSidebarColor": null,
"retainedEarningsAcctCode": "1234",
"continuingEdUnits": null,
"id": "1A6FE000002D5",
"accountingCode": null,
"inSiteBanner": null,
"IsFull": false,
"accentColor": "#808080",
"contact": null,
"eventsPageTopText": "Use this page to sign up for events.",
"events": [{"uri": "/protectedRest/Event/2B81B0000019A", "id":
"2B81B0000019A"},
        {"uri": "/protectedRest/Event/1A72D000001CA", "id":
"1A72D000001CA"}],
"showFeesOnPublicView": 1,
"inSiteFontName": null,
"lowRoomTrigger": null,
"description": null,
"mediaAllowed": null,
"onlineRegistrationsEnabled": 1,
"includes": null,
"travelInfo": null,
"enableRSVP": 2,
"PublicUrl":
"www.nafc.maxx.matrixdev.net/forms/meeting/Microsite/2014-annual-meeting",
"enableHousing": 2,
"client": {"uri": "/protectedRest/Client/1DE00000009", "id":
"1DE00000009"},
"dateClosed": null,
"Year": 2014,
"maximumNumAttendees": 1000,
```

```
"inSiteNavbarColor": null,  
"cosponsors": null,  
"waitingListRecords": null}
```

Example: Individual 15E00090F65

<https://{baseUrl}/protectedRest/Individual/15E00090F65>

```
{  
  personalStatus: "Active Staff",  
  groupAffiliations: null,  
  CombinedMembershipStatus: 2,  
  isDirectoryContact: "N",  
  personalBackground: null,  
  prefix: null,  
  recordOrigin: "376108",  
  aliases: "Lara",  
  mainAddress: {  
    uri: "/protectedRest/Address/15E0001155D",  
    id: "15E0001155D"  
  },  
  title: "Director, Health & Safety",  
  HybridStatusDescription: "Inactive NonMember",  
  homeFax: null,  
  secondaryPhone: null,  
  publishInDirectory: "No",  
  pledges: null,  
  mobilePhone: null,  
  assistant: {  
    uri: "/protectedRest/Spouse/1DC300000020",  
    id: "1DC300000020"  
  },  
  congressParty: null,  
  pictureFile: null,  
  mxxCreated: "2002-11-19 00:00:00",  
  CombinedMembershipType: 7,  
  isPrimaryContact: "",  
  isPaid: "N",  
  communications: [  
    {  
      uri: "/protectedRest/MemberCommunication/178310000058C",  
      id: "178310000058C"  
    }  
  ]  
}
```

```
    }
  ],
  jobLevel: null,
  salutation: null,
  MembershipStatusText: "Inactive Special/Temporary Member",
  PreferredFax: "(202) 457-0486",
  comments: [
    {
      uri: "/protectedRest/Comment/1EF900000051",
      id: "1EF900000051"
    }
  ],
  email: "test@maxx.matrixgroup.net",
  individualType: null,
  maxxCreatedBy: "15B000000002",
  isActiveMember: "Inactive",
  congresspersonFor: null,
  dontSnailmail: "No",
  isOutreachLiaison: "N",
  billingContactFor: null,
  biographies: null,
  outreachLiaisonFor: null,
  currentCompany: {
    uri: "/protectedRest/Company/100000000C",
    id: "100000000C"
  },
  socialNetworking: null,
  isSomeContact: "Y",
  designations: null,
  directoryAddress: {
    uri: "/protectedRest/Address/15E0001155D",
    id: "15E0001155D"
  },
  childRelationships: null,
  prospectStatus: null,
  shippingAddress: {
    uri: "/protectedRest/Address/750C00000000C",
    id: "750C00000000C"
  },
  PreferredPhone: "(202) 457-0481",
  isGovernmentRelationsContact: "N",
  addresses: [
    {
      uri: "/protectedRest/Address/751000000000B",
      id: "751000000000B"
    }
  ],
]
```

```
degreeLevel: null,
preferredMethodOfComm: "E-Mail",
canSpams: null,
primaryContactFor: null,
source: null,
Assistant: null,
contentAreas: null,
sourceOther: null,
enteredBy: null,
parentRelationships: [
  {
    uri: "/protectedRest/IndividualRelationship/282800000026E",
    id: "282800000026E"
  }
],
FullNameNormal: "Lara Smith",
governmentRelationsContactFor: null,
keyContactFor: null,
companyRoles: [
  {
    uri: "/protectedRest/CompanyRole/FE65000000A4",
    id: "FE65000000A4"
  }
],
dontEmail: "No",
criticalNotes: null,
directoryContactFor: null,
visionsId: null,
firstName: "Lara",
CompanyName: "National Penguin & Ranchers Association (NPRA)",
middleInitial: null,
suffix: null,
lastName: "Smith",
FullName: "Smith, Lara",
canSpamSet: null,
alternateContactFor: null,
individualNum: 33131,
id: "15E00090F65",
ethnicity: null,
JobFunctionStr: "",
otherJobFunction: null,
specialInterests: [
  {
    uri: "/protectedRest/Category/2",
    id: "2"
  }
],
```

```

    isActiveGuest: false,
    membership: {
      uri: "/protectedRest/Membership/14CD500000757",
      id: "14CD500000757"
    },
    user: {
      uri: "/protectedRest/User/15E00090F71",
      id: "15E00090F71"
    },
    isAlternateContact: "N",
    highResImage: null,
    spouse: {
      uri: "/protectedRest/Spouse/1DC300000022",
      id: "1DC300000022"
    },
    pager: null,
    salary: null,
    committeeMemberships: [
      {
        uri: "/protectedRest/CommitteeMember/E6B000000095",
        id: "E6B000000095"
      },
      {
        uri: "/protectedRest/CommitteeMember/E6D00000003B",
        id: "E6D00000003B"
      }
    ],
    emailFromThirdPary: "No",
    alternateEmail: null,
    birthDay: null,
    homeAddress: {
      uri: "/protectedRest/Address/75100000000B",
      id: "75100000000B"
    },
    knowledges: null,
    isBillingContact: "N"
  }
}

```

Example: Company 16D00000007

curl <https://intranet.nafc.maxx.matrixdev.net/protectedRest/Company/16D00000007>

```
{
```



```
"congressofficeFor": null,
"alternateContacts": [
  {
    "uri": "/protectedRest/Individual/3E190000006F",
    "id": "3E190000006F"
  },
  {
    "uri": "/protectedRest/Individual/76A000003A2",
    "id": "76A000003A2"
  }
],
"exhibitions": [
  {
    "uri": "/protectedRest/Exhibit/2BA00000009",
    "id": "2BA00000009"
  },
  {
    "uri": "/protectedRest/Exhibit/D4500000002",
    "id": "D4500000002"
  },
  {
    "uri": "/protectedRest/Exhibit/19B300000082",
    "id": "19B300000082"
  },
  {
    "uri": "/protectedRest/Exhibit/21A700000020",
    "id": "21A700000020"
  },
  {
    "uri": "/protectedRest/Exhibit/343F00000060",
    "id": "343F00000060"
  },
  {
    "uri": "/protectedRest/Exhibit/36FF0000007F",
    "id": "36FF0000007F"
  },
  {
    "uri": "/protectedRest/Exhibit/40BA00000002",
    "id": "40BA00000002"
  },
  {
    "uri": "/protectedRest/Exhibit/40BC000000CD",
    "id": "40BC000000CD"
  },
  {
    "uri": "/protectedRest/Exhibit/56E200000004",
    "id": "56E200000004"
  }
]
```

```
    },
    {
      "uri": "/protectedRest/Exhibit/56E700000002",
      "id": "56E700000002"
    },
    {
      "uri": "/protectedRest/Exhibit/56E700000002F",
      "id": "56E700000002F"
    },
    {
      "uri": "/protectedRest/Exhibit/56F0000000020",
      "id": "56F0000000020"
    },
    {
      "uri": "/protectedRest/Exhibit/56F00000000BA",
      "id": "56F00000000BA"
    },
    {
      "uri": "/protectedRest/Exhibit/56F00000000E7",
      "id": "56F00000000E7"
    },
    {
      "uri": "/protectedRest/Exhibit/56F6000000021",
      "id": "56F6000000021"
    },
    {
      "uri": "/protectedRest/Exhibit/56F600000004F",
      "id": "56F600000004F"
    },
    {
      "uri": "/protectedRest/Exhibit/56F700000002F",
      "id": "56F700000002F"
    },
    {
      "uri": "/protectedRest/Exhibit/5726000000002",
      "id": "5726000000002"
    },
    {
      "uri": "/protectedRest/Exhibit/11F84000000001",
      "id": "11F84000000001"
    },
    {
      "uri": "/protectedRest/Exhibit/2ECBD000000004",
      "id": "2ECBD000000004"
    },
    {
      "uri": "/protectedRest/Exhibit/373F00000000043",
```

```

        "id": "373F000000043"
    }
],
    "diversityCertifications": null,
    "emailDomains": null,
    "DateJoined": "2010-07-02 00:00:00",
    "congressSenators": "Sen. Dianne Feinstein (D-CA), Sen. Barbara Boxer (D-CA)",
    "ParentCompanyNum": 0,
    "howHeardAbout": null,
    "ParentCompanyName": "",
    "RealActiveMembershipType": 1,
    "meetingHotels": null,
    "isNonmember": 0,
    "congressRep": "Rep. Adam B. Schiff (D-CA)",
    "approvedEmailDomains": null,
    "numFunEvents": 3,
    "staffing": null,
    "associations": null,
    "additionalRosterSlots": null,
    "NameForSearch": "",
    "hasSubsidiaries": null,
    "congressSenator1": "Sen. Dianne Feinstein (D-CA)",
    "congressSenator2": "Sen. Barbara Boxer (D-CA)",
    "mxxCreated": "2007-06-11 17:05:40",
    "ticker": "ACME",
    "isActive": true,
    "primaryEmail": "info@acme.com",
    "isPaid": "F",
    "name": "ACME & Company",
    "countries": [
        {
            "uri": "/protectedRest/Country/1",
            "id": "1"
        },
        {
            "uri": "/protectedRest/Country/22",
            "id": "22"
        }
    ],
    "otherProductCategory": null,
    "matchString": "ACME & Company|ACME and Company|ACME n Company|A Company that Makes Everything|ACME ABC ACME Toys|ACME",
    "hasWorkplaceFun": "Yes",
    "companyStatusId": null,
    "issuesOfImportance": null,
    "PreferredTollFree": "",

```

```
    "invoiceInformation": null,
    "recordOrigin": null,
    "logo":
"/home/matrix/maxx84_nafc/nafc_org/uploadFiles/16D00000007.logo.acme.gif",
    "isMembershipInherited": 0,
    "minorityOwned": "Yes",
    "comments": [
      {
        "uri": "/protectedRest/Comment/5A300000057",
        "id": "5A300000057"
      },
      {
        "uri": "/protectedRest/Comment/12FA00000021",
        "id": "12FA00000021"
      },
      {
        "uri": "/protectedRest/Comment/12FE00000006",
        "id": "12FE00000006"
      },
      {
        "uri": "/protectedRest/Comment/12FE0000000F",
        "id": "12FE0000000F"
      },
      {
        "uri": "/protectedRest/Comment/1CF00000001D",
        "id": "1CF00000001D"
      },
      {
        "uri": "/protectedRest/Comment/1CF000000027",
        "id": "1CF000000027"
      },
      {
        "uri": "/protectedRest/Comment/1CF500000004",
        "id": "1CF500000004"
      },
      {
        "uri": "/protectedRest/Comment/1CF50000001A",
        "id": "1CF50000001A"
      },
      {
        "uri": "/protectedRest/Comment/1CFB00000004",
        "id": "1CFB00000004"
      },
      {
        "uri": "/protectedRest/Comment/1CFB00000010",
        "id": "1CFB00000010"
      }
    ],
```

```
{
  "uri": "/protectedRest/Comment/1CFB0000001C",
  "id": "1CFB0000001C"
},
{
  "uri": "/protectedRest/Comment/1CFB00000029",
  "id": "1CFB00000029"
},
{
  "uri": "/protectedRest/Comment/1CFF00000004",
  "id": "1CFF00000004"
},
{
  "uri": "/protectedRest/Comment/1D0300000001",
  "id": "1D0300000001"
},
{
  "uri": "/protectedRest/Comment/344900000229",
  "id": "344900000229"
}
],
"isPubliclyTraded": "Yes",
"NameAndSubDiv": "ACME & Company<br />(Sub/Div)",
"aliases": null,
"DerivedMembershipId": "6AB00000135",
"directoryContact": {
  "uri": "/protectedRest/Individual/1121800000001",
  "id": "1121800000001"
},
"article": null,
"Dues": null,
"RealMembershipStatus": "Active",
"socialNetworking": [
  {
    "uri": "/protectedRest/SocialNetworking/C9BA00000048",
    "id": "C9BA00000048"
  },
  {
    "uri": "/protectedRest/SocialNetworking/D7A50000004B",
    "id": "D7A50000004B"
  },
  {
    "uri": "/protectedRest/SocialNetworking/D7AD00000002",
    "id": "D7AD00000002"
  },
  {
    "uri": "/protectedRest/SocialNetworking/D7D800000008",
```

```
        "id": "D7D800000008"
      },
      {
        "uri": "/protectedRest/SocialNetworking/D7D800000013",
        "id": "D7D800000013"
      },
      {
        "uri": "/protectedRest/SocialNetworking/3109D0000000A",
        "id": "3109D0000000A"
      },
      {
        "uri": "/protectedRest/SocialNetworking/3D89E00000016",
        "id": "3D89E00000016"
      }
    ],
    "grossRevenue": 123456.0,
    "hasActiveIndividualMember": true,
    "duesBudget": null,
    "prospectStatus": null,
    "womanOwned": "No",
    "taxId": null,
    "brands": "ACME, ABC, ACME Toys",
    "highResImage": {
      "uri": "/protectedRest/File/1B5C800000003",
      "id": "1B5C800000003"
    },
    "owner": null,
    "PreferredFax": "",
    "hasSocialCommittee": "No",
    "silent": null,
    "PreferredPhone": "D111|222|3333|",
    "enteredBy": null,
    "SubDivText": "",
    "parentRelationships": null,
    "RealMembershipType": "Member",
    "website": [
      {
        "uri": "/protectedRest/WebAddress/16E0000000B",
        "id": "16E0000000B"
      }
    ],
    "paidDues": "F",
    "PublicLogo":
"http://intranet.nafc.maxx.matrixdev.net/uploadFiles/16D00000007.logo.acme.gif",
    "hotel": null,
    "lastUpdated": "2018-09-28 11:01:05",
```

```
    "lastUpdatedBy": "maki",
    "name2": "A Company that Makes Everything",
    "criticalNotes": "ACME is NAFC's first member. They are very
special.",
    "totalEmployees": 100,
    "companyStruct": null,
    "riskAssessment": "Acme is solid as a rock.",
    "numFilledSlots": null,
    "MemberSegment": 0,
    "yearEstablished": 1922,
    "congresspersonFor": null,
    "annualSales": null,
    "states": [
      {
        "uri": "/protectedRest/StateProvince/3",
        "id": "3"
      },
      {
        "uri": "/protectedRest/StateProvince/5",
        "id": "5"
      },
      {
        "uri": "/protectedRest/StateProvince/8",
        "id": "8"
      }
    ],
    "billingContact": {
      "uri": "/protectedRest/Individual/112980000003F",
      "id": "112980000003F"
    },
    "DirectoryAddress": {
      "uri": "/protectedRest/Address/16D00000012",
      "id": "16D00000012"
    },
    "id": "16D00000007",
    "MemberSince": null,
    "alphaSort": "ACME",
    "isComplementary": null,
    "keyContact": {
      "uri": "/protectedRest/Individual/17500000014",
      "id": "17500000014"
    },
    "congressionalDistrict": "CA28",
    "description": "ACME Company: Largest cartoon company on the entire
PLANET !!",
    "riskRating": "Medium Risk",
    "companyTypeId": "Company",
```

```
"companyNum": 10,
"membership": {
  "uri": "/protectedRest/Membership/6AB00000135",
  "id": "6AB00000135"
},
"childRelationships": [
  {
    "uri": "/protectedRest/Relationship/16D00000023",
    "id": "16D00000023"
  },
  {
    "uri": "/protectedRest/Relationship/7A5A0000003A",
    "id": "7A5A0000003A"
  },
  {
    "uri": "/protectedRest/Relationship/7A5A0000008C",
    "id": "7A5A0000008C"
  },
  {
    "uri": "/protectedRest/Relationship/869F000000657",
    "id": "869F000000657"
  }
],
"services": [
  {
    "uri": "/protectedRest/Service/8",
    "id": "8"
  },
  {
    "uri": "/protectedRest/Service/6",
    "id": "6"
  }
],
"crmLeads": [
  {
    "uri": "/protectedRest/CRMLead/31D4700000090",
    "id": "31D4700000090"
  },
  {
    "uri": "/protectedRest/CRMLead/31E760000003C",
    "id": "31E760000003C"
  },
  {
    "uri": "/protectedRest/CRMLead/3232900000006",
    "id": "3232900000006"
  },
  {
```



```

        "uri": "/protectedRest/CRMLead/3233F000000001",
        "id": "3233F000000001"
    },
    {
        "uri": "/protectedRest/CRMLead/323BC000000001",
        "id": "323BC000000001"
    }
],
"createdOnWWW": "No",
"sizeCode": {
    "uri": "/protectedRest/SizeCode/310C6",
    "id": "310C6"
},
"branches": [
    {
        "uri": "/protectedRest/Branch/16D00000011",
        "id": "16D00000011"
    }
],
"DateCanceled": null,
"isActiveParent": true,
"competitors": [
    {
        "uri": "/protectedRest/Competition/1A230000002F",
        "id": "1A230000002F"
    },
    {
        "uri": "/protectedRest/Competition/1D6000000151",
        "id": "1D6000000151"
    },
    {
        "uri": "/protectedRest/Competition/1D60000001CD",
        "id": "1D60000001CD"
    },
    {
        "uri": "/protectedRest/Competition/972C00000027",
        "id": "972C00000027"
    }
],
"howHeardAboutOther": null
}

```

Example: Invoice 37B12000001F5

curl <https://intranet.nafc.maxx.matrixdev.net/protectedRest/Invoice/37B12000001F5>

```
{
  "exhibit": null,
  "dateOfInvoice": "2015-09-22 11:16:51",
  "pledge": null,
  "coupon": null,
  "InvoiceDescriptionPretty": "Membership Dues 2016",
  "adjustments": null,
  "mxxUpdated": null,
  "TotalAdjustments": "0.00",
  "sponsor": null,
  "registration": null,
  "retainedEarnings": null,
  "TotalRefundsIssued": "0.00",
  "invoiceNum": 1507,
  "id": "37B12000001F5",
  "DateOfLastPayment": null,
  "Status": "No Payments (Balance Due)",
  "preferredLineItems": null,
  "invoiceNumber": "37B12000001F6",
  "MembershipFee": null,
  "discounts": [],
  "RelatedObjectType": "Application",
  "Balance": "500.00",
  "IsCancelled": 2,
  "comments": null,
  "NetAmountNonDues": "500.00",
  "arItems": null,
  "application": {
    "uri": "/protectedRest/Application/37B12000001CD",
    "id": "37B12000001CD"
  },
  "SADateCancelled": null,
  "TotalPayments": "0.00",
  "mxxCreated": "2015-09-22 11:16:51",
  "_SABalance": 50000,
  "RefundMethod": null,
  "VerboseDescription": "Application",
  "AmountDue": "500.00",
  "NetAmount": "500.00",
  "AdditionalContact": null,
  "TotalDiscounts": "0.00",
  "TotalRegularPayments": "0.00",
  "mxxCreatedBy": "18E000000006",
  "_lineItems": [
    {
      "uri": "/protectedRest/LineItem/37B12000001F7",
```

```
        "id": "37B12000001F7"
    },
    ],
    "BillingCompanyName": "ACME & Company",
    "company": {
        "uri": "/protectedRest/Company/16D00000007",
        "id": "16D00000007"
    },
    "TotalPaymentsDues": "0.00",
    "TotalRetainedEarnings": "500.00",
    "mxxUpdatedBy": null,
    "HasProcessedPayments": false,
    "NetAmountIgnoreCancellation": "500.00",
    "InvoiceDescriptionForExport": "Membership Dues 2016",
    "TotalPaymentsNonDues": "0.00",
    "_SAAmountPaid": 0,
    "ViewLink":
"/applications/ApplicationFormPrivate/view?id=37B12000001CD",
    "purchase": null,
    "TotalPositivePayments": "0.00",
    "isPaid": false,
    "NetAmountDues": "500.00",
    "TotalCredits": null,
    "RefundDue": -0.0,
    "bill": null,
    "Description": "Application",
    "installmentSchedule": null,
    "amount": "500",
    "individuals": null,
    "payments": null,
    "HasPendingPayments": false,
    "DateCancelled": null,
    "breakdownOfInvoice": [
        {
            "itemDate": "2015-09-22 11:16:51",
            "isAdjustment": null,
            "itemDescription": "Member Renewal",
            "itemPrice": 500,
            "itemTotal": 500,
            "itemQuantity": 1,
            "itemAcctCode": "applicationCode"
        }
    ],
    "_SAAmountAR": 0,
    "_SAAmountAP": 0
}
```

Searching

Searching is accomplished by composing a URL that denotes the collection of resources you would like to retrieve. The arguments to the search will be listed after the url as querystring arguments (?arg1=value1&arg2=value2...). The left hand side of the arg-value pairs in its basic form is the attribute name for which you are searching. Additionally, it can contain an argument composed of an associated class, underscore, and the attribute name in the associated class.

The total number of records that match a search will be added to the HTTP header of the response as TotalNumberOfRows. This will be an integer value.

There is currently no way to control from the request the attributes placed into the result set. Each class has some defaults, but if you find it convenient to have other values placed into the result set let us know and we will try to accommodate that. Queries that cannot be described with the current query language can be coded or implemented by Matrix.

Query Language

Values in the arg=value clause can take several forms. The simplest is just the value itself, which will restrict the search to exact matches on the value. More complex searches involve operators for things like greater-than; 'like'; is-not-null.

- integer - exact match. Eg. numEmployees=50
- string - exact match. Eg. lastName=Smith
- in(valueList) - match any of the values in valueList. When using this with MaxxIds values need to be of the form 0x178310000058C with the 0x prefix or converted to the integer representation. If the values in valueList are strings, the string must be enclosed in (url-encoded) quotes. Eg. year=in(2008,2009), lastName=in(%27Smoot%27,%27Hawley%27)
- like(str) - wild card match as in [SQL](#). Eg. lastName=like(Mc%)
- neq(value) - not equals. Eg. firstName=neq(Natasha)
- neqnull(value)- not equals, ignoring nulls. Eg. language=neqnull(English)
- isnull() - attribute is null. Eg. price=isnull()
- isnotnull() - attribute is not null. Eg. description=isnotnull()
- gt(value) - greater than. Eg. startDate=gt(2014-05-01)
- gtn(value) - greater than or NULL. e.g. startDate=gtn(2014-05-01) will return all matches where startDate is NULL or startDate is greater than 2014-05-01.
- lt(value) - less than. Eg. endDate=lt(2015-01-01)
- ltn(value) - less than or NULL
- geq(value) - greater than or equals value. Eg. numEmployees=geq(51)
- geqn(value) - greater than or equals value, or is NULL.
- leq(value) - less than or equals value. Eg. name=leq(Smith)
- leqn(value) - less than or equals value, or is NULL.
- orConditions - all the other conditions should be combined with 'OR' instead of 'AND'.

Example: like

Find individual whose last name starts with "Mc" and whose username contains a "d":

```
https://{baseUrl}/protectedRest/Individual?lastName=like(Mc%)&user_username=like(%d%)
```

```
[
{
  city: "Arkansas City",
  firstName: "Randy",
  title: null,
  url: "/protectedRest/Individual/11AF400000057",
  lastName: "McGill",
  currentCompany: {
    uri: "/protectedRest/Company/10300000000B",
    id: "10300000000B"
  },
  id: "11AF400000057",
  orgName: "Carrie's Balloon Company",
  state: "KS",
  orgId: "10300000000B",
  email: "test@maxx.matrixgroup.net"
}
]
```

Example: in

Find organizations whose companyNumber is either 27413 or 20781.

```
https://{baseUrl}/protectedRest/Company?companyNum=in(27413,20781)
```

```
[
{
  name: "Massachusetts College of Art and Design",
  url: "/protectedRest/Company/19B0000079F",
  companyNum: 27413,
  websites: "http://www.massart.edu",
  ParentCompanyNum: 0,
  id: "19B0000079F",
  RealActiveMembershipType: 13
},
{

```

```
name: "New England Institute of Technology",
url: "/protectedRest/Company/19B0000039D",
companyNum: 20781,
websites:
"http://www.neit.edu/Programs/Associate-Degree-Programs/Architecture-and-I
nterior-Design",
ParentCompanyNum: 0,
id: "19B0000039D",
RealActiveMembershipType: 27
}
```

[https://{baseURL}/protectedRest/Product?year=in\(2008,2009\)](https://{baseURL}/protectedRest/Product?year=in(2008,2009))

```
[
{
status: 1,
miniImage:
"/home/matrix/maxx93_nafc/nafc_org/uploadFiles/140100000007.miniImage.mini
.gif",
description: "Great ideas about best practices from real companies. Get
some great information that you can implement at your company today!",
productNumber: "20409",
url: "/protectedRest/Product/140100000007",
type: "Booklet",
isFeature: "2",
linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=2040
9",
author: null,
year: 2008,
shortDescription: null,
id: "140100000007",
categories: [ ],
name: "Best Practices of Fun Companies"
},
{
status: 1,
miniImage:
"/home/matrix/maxx93_nafc/nafc_org/uploadFiles/FD9000000077.miniImage.party
Pack2.gif",
description: "This CD-ROM has all the tools you need to have fun, fun,
fun.",
productNumber: "DVD-party1234",
url: "/protectedRest/Product/FD9000000077",
```

```
type: "Booklet",
isFeature: "1",
linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=DVD-
party1234",
author: null,
year: 2009,
shortDescription: null,
id: "FD900000077",
categories: [ ],
name: "NAFC Party Pack"
}
]
```

Find Individuals with first name Rowan or Lucy. Don't forget the (encoded) quotes around strings in an 'in' clause.

```
https://{baseURL}/protectedRest/Individual?firstName=in(%27Rowan%27,%27Lucy%27)
```

```
[
{
city: "Washington",
firstName: "Lucy",
title: "Press Secretary",
url: "/protectedRest/Individual/2CBF500000460",
lastName: "Speed",
currentCompany: {
uri: "/protectedRest/Company/8DE900001F81",
id: "8DE900001F81"
},
id: "2CBF500000460",
orgName: "Office of Sen. Mark Pryor",
state: "DC",
orgId: "8DE900001F81",
email: null
}
]
```

Example: orConditions

Find organizations who meet any of the following conditions: name has "England" in it; or guideSummary has "England" in it; or guideUniversitySetting has "England" in it.

```
https://{baseURL}/protectedRest/Company?orConditions&name=LIKE(England)&guideSummary=LIKE(England)&guideUniversitySetting=LIKE(England)
```

```
[
{
name: "Massachusetts College of Art and Design",
url: "/protectedRest/Company/19B0000079F",
companyNum: 27413,
websites: "http://www.massart.edu",
ParentCompanyNum: 0,
id: "19B0000079F",
RealActiveMembershipType: 13
},
{
name: "New England Institute of Technology",
url: "/protectedRest/Company/19B0000039D",
companyNum: 20781,
websites:
"http://www.neit.edu/Programs/Associate-Degree-Programs/Architecture-and-Interior-Design",
ParentCompanyNum: 0,
id: "19B0000039D",
RealActiveMembershipType: 27
},
{
name: "Norwich University",
url: "/protectedRest/Company/19B00000190",
companyNum: 20502,
websites: "http://www.norwicharchart.org/",
ParentCompanyNum: 0,
id: "19B00000190",
RealActiveMembershipType: 22
},
{
name: "University of Massachusetts, Amherst",
url: "/protectedRest/Company/19B000005F4",
companyNum: 27036,
websites: "http://www.umass.edu/architecture",
ParentCompanyNum: 0,
id: "19B000005F4",
RealActiveMembershipType: 22
}
]
```


Getting results in batches and ordering results (e.g. for pagination)

If you expect a large set of search results, you will need to retrieve the results in batches. This is often used in the pagination of results on the front-end. There are several arguments available to all searches to enable this. By default, a maximum of 800 rows are returned; the `cursorSize` argument can be used to tailor this number. `TotalNumberOfRows` (in the http header) will always contain the total number of matches, not just the number in your batch. These arguments include:

- `cursorStart` - (integer) which record should be the first returned (default: 0)
- `cursorSize` - (integer) maximum number of results to return (default: 800)
- `orderBy` - clause specifying how to order the results. You can order by multiple attributes by separating `orderBy` clauses with a comma. The clauses should be formatted as such:
`<className>_<attributeName>[%20(ASC|DESC)]`

Example: Retrieving the first 2 featured published products, ordered first by author, descending, then by name, ascending

```
https://{baseUrl}/protectedRest/Product?isFeature=1&status=1&cursorStart=0
&cursorSize=2&orderBy=Product_author%20DESC,Product_name
```

```
[
{
  status: 1,
  miniImage: null,
  description: "Supermodern dadaism on paper. Glossy, medium
weight.Supermodern dadaism on paper. Glossy, medium weight.Supermodern
dadaism on paper. Glossy, medium weight.Supermodern dadaism on paper.
Glossy, medium weight.",
  productNumber: "33333",
  url: "/protectedRest/Product/157C5000002FD",
  type: "Poster",
  isFeature: "1",
  linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=3333
3",
  author: "Vacamann",
  year: 2010,
  shortDescription: "Supermodern dadaism on paper. Glossy, medium weight.",
  id: "157C5000002FD",
  categories: [
    "Employee Morale"
  ],
}
```

```

name: "Stuff and Nonsense - 12 Horses"
},
{
status: 1,
miniImage: null,
description: "Popular poster made digital by even more popular
demandPopular poster made digital by even more popular demandPopular
poster made digital by even more popular demandPopular poster made digital
by even more popular demandPopular poster made digital by even more
popular demand",
productNumber: "00-00-0011",
url: "/protectedRest/Product/157C4000004C6",
type: "Poster",
isFeature: "1",
linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=00-0
0-0011",
author: "Vacamann",
year: 2010,
shortDescription: "Popular poster made digital by even more popular
demand",
id: "157C4000004C6",
categories: [
"Employee Morale"
],
name: "Stuff and Nonsense [the digital sequel] - 12 Horses"
}
]

```

Companies (Organizations)

Currently, you make a call to the address object from each company. However, we are currently working on a more efficient way to do this as we have clients wanting to create custom directories rather than use our built-in templates.

Users

Accessing information about a User given username

To get a small set of information about an individual given their username, use the UserProfile service. In addition to name, organization name, email, membership status, etc, this service also returns a list of the committees the user is a member of, and a list of the users's privilege groups (which control access to various parts of the website). The following example retrieves information about user 'Grumpy':

Example: User 'Grumpy'

```
https://{baseUrl}/protectedRest/UserProfile/Grumpy
```

```
{
  "currentOrgName": "National Association of Fun Companies edit (NAFC)",
  "user_id": "1D4200000062",
  "individual": {
    "suffix": null,
    "personalStatus": "Active Staff",
    "isDirectoryContact": "N",
    "isKeyContact": "Y",
    "id": "1D4200000041",
    "primaryContactFor": null,
    "HybridStatusDescription": "Active Guest (Temporary Individual Membership)",
    "hasStaffPrivileges": true,
    "FullNameNormal": "Mr Grumpy Dwarf",
    "email": "test@matrixgrouptest.net",
    "individualType": null,
    "isActiveMember": "Active",
    "companyRoles": null,
    "isActiveGuest": true,
    "isPrimaryContact": "",
    "firstName": "Grumpy",
    "lastName": "Dwarf",
    "middleInitial": null,
    "otherJobFunction": null,
    "jobLevel": null,
    "FullName": "Dwarf, Mr Grumpy",
    "canSpamSet": "Y"},
  "committees": ["Board", "Executive Committee"],
  "groups": ["Staff", "Siteadmin", "Staffadmin", "TradeshowMaxx", "Intranet", "MeetingMaxx", "CommerceMaxx", "Membership: Committees", "Calendar", "Membership", "InfoMaxx", "MailingMaxx", "merge", "Member-Only", "Accounting", "Membership: Add-Edit", "MembershipView", "SponsorshipMaxx", "FundraisingMaxx", "ContentMaxx", "Tasks", "SurveyMaxx", "List: EXEC_Committee-list", "List: board-committee"]
}
```

Accessing information about a User given session id (from session cookie)

To get information about a use given their session id, use the InteractiveSession service.

- 'accesslevels_full' is a superset of accesslevels, and includes all inherited privilege groups.
- 'accesslevels' only includes the assigned privilege groups without all the inherited ones.
- 'impersonator' indicates this user was authenticated via impersonation (siteadmin logged in as the user)

Example: Session 1610076

`https://{baseUrl}/protectedRest/InteractiveSession/1610076`

```
{
  "username": "Grumpy",
  "impersonator": {
    "uri": "/protectedRest/Individual/1520000001C",
    "id": "1520000001C"
  },
  "name": "Grumpy Harry Dwarf",
  "accesslevels_full": [
    {
      "url": "/protectedRest/PrivilegeGroup/1520000001C",
      "name": "MembershipView",
      "id": "1520000001C"
    },
    {
      "url": "/protectedRest/PrivilegeGroup/6",
      "name": "Staffadmin",
      "id": "6"
    },
    {
      "url": "/protectedRest/PrivilegeGroup/1546B0000001E",
      "name": "MailingList: Awards_Committee",
      "id": "1546B0000001E"
    },
    {
      "url": "/protectedRest/PrivilegeGroup/C",
      "name": "Calendar",
      "id": "C"
    },
    {
      "url": "/protectedRest/PrivilegeGroup/16F00000000",
      "name": "SurveyMaxx",
      "id": "16F00000000"
    }
  ],
  "userid": "19B00042413",
  "individualid": "19B00042412",
  "accesslevels": [
    {
      "url": "/protectedRest/PrivilegeGroup/6",
      "name": "Staffadmin",
      "id": "6"
    },
    {
      "url": "/protectedRest/PrivilegeGroup/1546B00000008",
      "name": "MailingList: Awards_Committee",
      "id": "1546B00000008"
    }
  ],
  "individual": {
    "uri": "/protectedRest/Individual/19B00042412",
    "id": "19B00042412"
  }
}
```

Logging out a user given session id (from session cookie)

To log a user out, use the 'DELETE' http method instead of GET (or PUT or POST). This will remove the given session from the MatrixMaxx database, and clear any CAS tickets for that user (if CAS authentication is in use). You should still expire the browser session cookie yourself (we don't have access to the browser, so can't do that via webservices).

Pulling Specific Committees Given Individual ID

Given an Individual ID, this call will return all current committee memberships (meaning the user has active role, with a blank or future end date and not an expired or deleted role) on an active committee . (This call DOES include silent memberships)

`/protectedRest/CommitteeMember?individual_id=19D0005BCAD&roles_isDeleted=neqnull(1)&committee_status=1&committee_deleted=neqnull(Y)&roles_endDate=isNull()`

Example: Session 1475382

```
curl -X DELETE https://{baseUrl}/protectedRest/InteractiveSession/1475382
```

Accessing Committee Information

You may want to pull committee information. This section pulls info from the committee, not from the committee-generated list, which is an important distinction depending on what is being done.

`https://{baseUrl}/protectedRest/Committee`

would pull all the committees, but you probably only want active status committees, so

`https://{baseUrl}/protectedRest/Committee?committee_status=1`

This returns a dictionary with IDs. So from there you could pick specific committee ID(s) to work with.

You may also want to be working with certain roles on the committee, or excluding certain roles; to see all the roles

`https://{baseUrl}/protectedRest/WebSelect/CommitteeRoles`

So to pull all the 'current' individuals on a given committee, minus the silents, you would do something like this ...

```
https://{baseUrl}/protectedRest/Individual?committeeMemberships_committee_id=COMMITTEEID&committeeMemberships_roles_startDate=lt(TODAY)&committeeMemberships_roles_endDate=gtn(TODAY)&committeeMemberships_roles_role=neq(6)
```

For Example

```
https://{baseUrl}/protectedRest/Individual?committeeMemberships_committee_id=8908D0000000B&committeeMemberships_roles_startDate=lt(2018-07-31)&committeeMemberships_roles_endDate=gtn(2018-07-31)&committeeMemberships_roles_role=neq(6)
```

Accessing Upcoming Meetings

Use the UpcomingMeeting service to retrieve information about upcoming published meetings. There are several optional arguments available for filtering the results, including:

- number: (integer) maximum number of upcoming meetings to return of each meeting type.
- cursorSize: (integer) total maximum number of upcoming meetings to return, regardless of type. Does not work in conjunction with 'number' param above.
- includeNotEnded: (integer) if '1', include meetings currently in progress, else only return meetings that haven't started yet
- calendarName: (string) specify which calendar to pull events from (default depends on client, but usually is 'Public')
- type: (string) specify the meeting type you want. Options may vary by client, but default options are Seminar, Convention, Conference, Committee Meeting, Board Meeting, Other, Meeting, Conference Call. You can get your options with the WebSelect service, using webselect 'MeetingType'

Example: Pull Upcoming Meetings that are published to a particular calendar in MatrixMaxx

UpcomingMeetings is a special call that has some smarts built into it and is typically used to pull meetings onto a home page website calendar. You can pull all upcoming meetings published to a particular calendar type, or just a certain number of them.

This pulls all upcoming meetings published to a specific Calendar

```
https://{baseUrl}/protectedRest/UpcomingMeetings?calendarName=ACME%20Home%20Page%20Upcoming%20Events
```

This pulls the next 10 upcoming meetings published to a specific Calendar

```
https://{baseUrl}/protectedRest/UpcomingMeetings?number=10&calendarName=ACME%20Home%20Page%20Upcoming%20Events
```

Example: Next two meetings of each type, including those in progress

```
https://{baseURL}/protectedRest/UpcomingMeetings?number=2&includeNotEnded=1
```

```
[
  {
    "meeting_type": "Meeting",
    "meeting_categories": [],
    "meeting_id": "15726000002CA",
    "meeting_location_zip": null,
    "meeting_end_date_withzone": "2014/04/05 23:59:59 US/Eastern",
    "meeting_PublicUrl":
      "www.nafc.maxx.matrixdev.net/forms/meeting/Microsite/DiannesFuntastic",
    "meeting_location_country": "United States",
    "meeting_location_city": "Washington",
    "meeting_start_date": "2014-04-01 00:00:00",
    "meeting_start_date_withzone": "2014/04/01",
    "meeting_location_name": "Washington Convention Center",
    "meeting_name": "Dianne's Group Reg discount Funtastic Meeting 2013",
    "meeting_end_date": "2014-04-05 23:59:59",
    "meeting_cmsImageFilename": null,
    "meeting_location_street": ["Room 1000"],
    "meeting_location_region": "DC"},
  {
    "meeting_type": "Meeting",
    "meeting_categories": [],
    "meeting_id": "2D43100000001",
    "meeting_cmsImageFilename": null,
    "meeting_end_date_withzone": "2014/04/08",
    "meeting_PublicUrl":
      "www.nafc.maxx.matrixdev.net/forms/meeting/MeetingFormPublic/view?id=2D43100000001",
  }
]
```

```
"meeting_start_date": "2014-04-01 00:00:00",  
"meeting_start_date_withzone": "2014/04/01",  
"meeting_name": "WR 98265 Test Meeting",  
"meeting_end_date": "2014-04-08 00:00:00"}  
]
```

Example: Determine possible meeting types

<https://{baseUrl}/protectedRest/WebSelect/MeetingType>

```
[  
{  
  id: 7,  
  description: "Meeting"  
},  
{  
  id: 1,  
  description: "Seminar"  
},  
{  
  id: 2,  
  description: "Convention"  
},  
{  
  id: 3,  
  description: "Conference"  
},  
{  
  id: 4,  
  description: "Committee Meeting"  
},  
{  
  id: 8,  
  description: "Conference Call"  
},  
{  
  id: 5,  
  description: "Board Meeting"  
},  
{  
  id: 6,  
  description: "Other"  
}  
]
```


Example: Find upcoming conference calls

```
https://{baseUrl}/protectedRest/UpcomingMeetings?number=3&type=Conference%20Call
```

```
[
{
  meeting_type: "Conference Call",
  meeting_categories: [ ],
  meeting_id: "2D431000000001",
  meeting_cmsImageFilename: null,
  meeting_end_date_withzone: "2014/04/08",
  meeting_PublicUrl:
  "www.nafc.maxx.matrixdev.net/forms/meeting/MeetingFormPublic/view?id=2D431000000001",
  meeting_start_date: "2014-04-01 00:00:00",
  meeting_start_date_withzone: "2014/04/01",
  meeting_name: "WR 98265 Test Meeting",
  meeting_end_date: "2014-04-08 00:00:00"
}
]
```

Example: Find meetings that start after May 1, 2014

Regular search terms work too. Search object is CalendarItem, so you can get metadata on CalendarItem to learn what fields are searchable (<https://{baseUrl}/protectedRest/Metadata/CalendarItem>). Note that the UpcomingMeetings service will only return Calendar Items that are attached to meetings (the database can also contain non-meeting calendar items).

```
https://{baseUrl}/protectedRest/UpcomingMeetings?number=3&startDate=gt(2014-05-01)
```

```
[
{
  meeting_type: "Meeting",
  meeting_categories: [ ],
  meeting_id: "1A6FE000002D5",
  meeting_location_zip: "222024002",
  meeting_end_date_withzone: "2014/10/05",
  meeting_PublicUrl:
  "www.nafc.maxx.matrixdev.net/forms/meeting/Microsite/2014-annual-meeting",
  meeting_location_country: "United States",
}
```

```
meeting_location_city: "Arlington",
meeting_start_date: "2014-10-01 09:00:00",
meeting_start_date_withzone: "2014/10/01 09:00:00 US/Eastern",
meeting_location_name: "Hyatt Crystal City",
meeting_name: "2014 Annual Meeting",
meeting_end_date: "2014-10-05 00:00:00",
meeting_cmsImageFilename: null,
meeting_location_street: [
"2799 Jefferson Davis Hwy"
],
meeting_location_region: "VA"
},
{
meeting_type: "Meeting",
meeting_categories: [ ],
meeting_id: "1310600000014",
meeting_location_zip: "222024002",
meeting_end_date_withzone: "2015/10/14",
meeting_PublicUrl:
"www.nafc.maxx.matrixdev.net/forms/meeting/Microsite/2015_fall_meeting",
meeting_location_country: "United States",
meeting_location_city: "Arlington",
meeting_start_date: "2015-10-10 00:00:00",
meeting_start_date_withzone: "2015/10/10",
meeting_location_name: "Hyatt Crystal City",
meeting_name: "2015 Fall Meeting",
meeting_end_date: "2015-10-14 00:00:00",
meeting_cmsImageFilename: null,
meeting_location_street: [
"2799 Jefferson Davis Hwy"
],
meeting_location_region: "VA"
},
{
meeting_type: "Meeting",
meeting_categories: [ ],
meeting_id: "E537000000001",
meeting_location_zip: null,
meeting_end_date_withzone: "2020/01/07 15:00:00 US/Eastern",
meeting_PublicUrl:
"www.nafc.maxx.matrixdev.net/forms/meeting/MeetingFormPublic/view?id=E537000000001",
meeting_location_country: null,
meeting_location_city: null,
meeting_start_date: "2020-01-05 07:00:00",
meeting_start_date_withzone: "2020/01/05 07:00:00 US/Eastern",
meeting_location_name: "Disney's Coronado Springs Resort",
```

```
meeting_name: "Annual Funtastic Meeting 2020",
meeting_end_date: "2020-01-07 15:00:00",
meeting_cmsImageFilename: null
}
]
```

Example: Get the next upcoming meeting categorized as 'Sustainability'

To filter results by meeting category, use the `meeting_categories_description` argument.

```
https://{baseUrl}/protectedRest/UpcomingMeetings?meeting_categories_description=Sustainability
```

```
[
{
meeting_type: "Conference Call",
meeting_categories: [
"Customer Service",
"Sustainability"
],
meeting_id: "2D431000000001",
meeting_cmsImageFilename: null,
meeting_end_date_withzone: "2014/04/08",
meeting_PublicUrl:
"www.nafc.maxx.matrixdev.net/forms/meeting/MeetingFormPublic/view?id=2D431000000001",
meeting_start_date: "2014-04-01 00:00:00",
meeting_start_date_withzone: "2014/04/01",
meeting_name: "WR 98265 Test Meeting",
meeting_end_date: "2014-04-08 00:00:00"
}
]
```

Accessing Meeting Attendees via Meeting Registrations

A note about the MatrixMaxx Meeting structure: **Registrations hold Attendees**

- A registration may be holding just the main registered attendee, but may also be holding a partner/spouse and an unlimited number of guests (e.g., children).
 - This means that **registrationId and attendeeId are different numbers!**
 - e.g., Sarah Rey is an attendee, and her spouse Marvin Rey is also coming
 - The registrationId is 270E0400000176
 - Her attendeeId is 26EEB800000077
 - Her Spouse's attendeeId is 26EEB800000078
- Registrations hold information about these attendees, like their chosen badge names, events, etc. The API may be used to pull information about these attendees, but there should be a discussion with the MatrixMaxx team about the exact specs of the calls as the Meeting setup of the association client varies. (i.e., attributes such as whether the registration is submitted, cancelled, rejected, reviewed, or paid all interact with each other)
- It may also be necessary to call back to the individual's basic record to get info from it that is not in the attendee information (demographics, interests, etc.)

Given a Meeting, pull all the registered individual attendees

As a starting point: Get a list of all the Registered Attendees for a given Meeting ID whose registrations have been approved:

```
https://{baseUrl}/protectedRest/Attendee?registration_meeting_id={meetingId}&registration_dateApproved=isNotNull()
```

Here is the same call, refined to make sure it leaves out rejected and cancelled registrations, which is almost never desirable. (This call is being used by several Virtual Meeting Platform - VMP - systems that pull data from us.)

```
https://{baseUrl}/protectedRest/Attendee/?registration_meeting_id={meetingId}&registration_dateApproved=isNotNull()&registration_dateCancelled=isNull()&registration_dateRejected=isNull()
```

You may refine this call further to find only registrations updated/created since a certain date: e.g., adding *registration_mxxUpdated=gt(2020-010-06)* will find only registrations updated/created since that date

From that initial pull of all attendees, you can pull the attendee data for each individual by calling:

```
https://{baseUrl}/protectedRest/Attendee/{attendeeId}
```

And if necessary, you can pull individual data for that attendee at:

```
https://{baseUrl}/protectedRest/Individual/{individualId}
```

Given an individual, pull all of their meeting registrations

Here is example of how to get all of the (approved) meeting registration that an attendee is registered for, given their individual ID. (This call can also be modified, as per above, to further refine the statuses of the registrations being included.)

```
https://{baseUrl}/protectedRest/Meeting?registrations_attendee_individual_id=1CCA6F000000C9&registrations_dateApproved=isNotNull()
```

Accessing Commerce Product Information

Retrieving information on commerce products is done with the Product service:

```
https://{baseUrl}/protectedRest/Product
```

Product metadata can be examined with <https://{baseUrl}/protectedRest/Metadata/Product>

To retrieve a list of results that contains only published products (which is usually what you'll want), include 'status=1' in your search arguments like this:

```
https://{baseUrl}/protectedRest/Product?status=1
```

Example: retrieving published products tagged with 'Employee Morale' category

```
https://{baseUrl}/protectedRest/Product?status=1&categories_description=Employee%20Morale
```

```
[
{
  status: 1,
  miniImage: null,
```

```
description: "Get 2010's newest issues of 'Bottomline Fun', hot off the
presses! Our most popular publication, Bottomline Fun gives real-life,
proven examples of ways to make your organization more 'fun' while at the
same time improving your bottom line.",
productNumber: "BF2010",
url: "/protectedRest/Product/5ADB0000000F",
type: "Newsletter",
isFeature: "1",
linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=BF20
10",
author: null,
year: 2010,
shortDescription: "The 2010 Subscription to Bottomline Fun, giving you
real-life ways to make money and have more fun at the same time.",
id: "5ADB0000000F",
categories: [
"Customer Service",
"Employee Morale",
"Employee Recruitment",
"Employee Retention",
"Profitability",
"Sales and Marketing"
],
name: "Bottomline Fun 2010"
},
{
status: 1,
miniImage: null,
description: "Provides access to all previous years' issues of 'Bottomline
Fun' Our most popular publication, Bottomline Fun gives real-life, proven
examples of ways to make your organization more 'fun' while at the same
time improving your bottom line. ",
productNumber: "bfarc",
url: "/protectedRest/Product/5AFA00000006C",
type: "Newsletter",
isFeature: "2",
linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=bfar
c",
author: null,
year: null,
shortDescription: null,
id: "5AFA00000006C",
categories: [
"Customer Service",
"Employee Morale",
```

```

    "Employee Recruitment",
    "Employee Retention",
    "Profitability",
    "Sales and Marketing"
  ],
  name: "Bottomline Fun Archive"
},
{
  status: 2,
  miniImage: null,
  description: "Another exciting entry in the build your own carnival series.",
  productNumber: "951413",
  url: "/protectedRest/Product/55BD0000009D",
  type: "Manual",
  isFeature: "2",
  linkToStore:
    "/forms/store/ProductFormPublic/search?action=1&Product_productNumber=951413",
  author: "Dizzy Galespie",
  year: 2010,
  shortDescription: null,
  id: "55BD0000009D",
  categories: [
    "Employee Morale"
  ],
  name: "Building Your Own Merry-go-round"
}
]

```

Example: Get featured published products

`https://{baseUrl}/protectedRest/Product?isFeature=1&status=1`

```

[
  {
    status: 1,
    miniImage:
      "/home/matrix/maxx132_nafc/nafc_org/uploadFiles/13252000000D3.miniImage.quirrelbooklittle.jpg",

```

```
description: "Seminal work by renowned farmer and country-living guru
Michael Ramkin, detailing his lifelong passion and success with prey birds
of all kinds and quail in particular.",
productNumber: "852258",
url: "/protectedRest/Product/13252000000D3",
type: "Book",
isFeature: "1",
linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=8522
58",
author: "Michael Ramkin",
year: 2001,
shortDescription: "Country-living guru Michael Ramkin details his lifelong
passion and success with prey birds, particularly quail.",
id: "13252000000D3",
categories: [
"Profitability",
"Sales and Marketing"
],
name: "A Fowl Business: On Quail and Prey Bird Husbandry"
},
{
status: 1,
miniImage: null,
description: "Get 2010's newest issues of 'Bottomline Fun', hot off the
presses! Our most popular publication, Bottomline Fun gives real-life,
proven examples of ways to make your organization more 'fun' while at the
same time improving your bottom line.",
productNumber: "BF2010",
url: "/protectedRest/Product/5ADB0000000F",
type: "Newsletter",
isFeature: "1",
linkToStore:
"/forms/store/ProductFormPublic/search?action=1&Product_productNumber=BF20
10",
author: null,
year: 2010,
shortDescription: "The 2010 Subscription to Bottomline Fun, giving you
real-life ways to make money and have more fun at the same time.",
id: "5ADB0000000F",
categories: [
"Customer Service",
"Employee Morale",
"Employee Recruitment",
"Employee Retention",
"Profitability",
"Sales and Marketing"
```



```
],  
name: "Bottomline Fun 2010"  
},
```

Example: retrieving and display product images

The public image paths pulled from the API can be translated to the public website URL by removing everything before /uploadFiles/

So if the API returns: /data/nafc/current/nafc_org/uploadFiles/[filename].jpg then You can access this image at: /uploadFiles/[filename].jpg

Additional Resources

<https://wiki.matrixgroup.net/index.php/RESTfulWebServices> - Contains example of POST request

API Documentation for other Systems

These are links to other API documents that we have found useful or we have integrated with or pulled from at some point

- Sunlight Foundation (Congressional Data):
https://wiki.matrixgroup.net/index.php/Agile_And_Maxx

Examples of CMS-driven websites pulling MatrixMaxx data into their pages

- ACSA Public Meeting rosters (Wordpress CMS calling MatrixMaxx attendee data for a particular meeting):
<https://www.acsa-arch.org/conference/2019-administrators-conference/registration/>
- ACGIH home page pulling upcoming events and highlighted commerce publications (SiteFinity CMS calling MatrixMaxx meeting and commerce data): <https://www.acgih.org/>
- AWP home page pulling upcoming events (WordPress CMS calling MatrixMaxx meeting data): <https://www.awpa.org/>
- FMI custom directory, written in ColdFusion, <https://www.fmi.org/member-directory/> is a mix of the newer protectedRest (search dropdowns and social media results) and our older WDDX (for the actual search, which has been around for a while)
- IDFA (members-only) committee pull onto individual portal pages in WordPress CMS.