

# Developing with CaseMap Server

## User Guide

CaseMap® Server v2.0

- **Getting Started with CaseMap Development**
- **CaseMap Server Programming Reference**



## Developing with CaseMap Server

No part of this work may be reproduced or transmitted in any form or by any means, electronic or mechanical, including photocopying, recording, or by any information storage or retrieval system, without permission.

While the information contained herein is believed to be accurate, this work is provided "as is," without warranty of any kind. The information contained in this work does not constitute, and is not intended as, legal advice.

LexisNexis and the Knowledge Burst logo are registered trademarks of Reed Elsevier Inc., used under license. CaseMap is a registered trademark of LexisNexis, a division of Reed Elsevier Inc. Other products or services may be trademarks or registered trademarks of their respective companies.

© 2015 LexisNexis. All rights reserved.

CaseMap®  
CaseMap® Server  
TextMap®  
TimeMap®  
NoteMap®  
Sanction®  
Concordance®

Version: CaseMap® Server 2.0, CaseMap® Admin Console 2.0, CaseMap® SharePoint Web Parts 1.0  
Release: March 30, 2015

# Table of Contents

<b>Chapter 1 Developing with CaseMap Server</b>	<b>5</b>
<b>1 Getting Started with CaseMap Server Development</b>	<b>5</b>
About CaseMap Server Development	5
What you need to know	5
About the CaseMap Server programming documentation	6
About the CaseMap Server API	6
About authentication	6
Tutorial: What's the version, CaseMap?	7
<b>2 CaseMap Server Programming Reference</b>	<b>8</b>
<b>About URI Structure</b>	<b>9</b>
/ (Root)	9
<service>	10
<type>	11
<resourceID>	12
? <query>	13
<b>About Verbs</b>	<b>15</b>
GET	15
<b>About request headers</b>	<b>17</b>
<b>About response headers</b>	<b>18</b>
<b>CaseMap REST API</b>	<b>19</b>
CMServerAD and CMServer Service Reference	19
GET get-version-info	20
GET get-cases	23
GET get-case-view s	26
GET get-case-users	32
GET get-case-data	35
GET get-case-{spreadsheet-name}	41
GET get-case-issues-outline	46
<b>CaseMap Server Helper Methods</b>	<b>52</b>
<b>Index</b>	<b>0</b>

# Developing with CaseMap Server

User Guide

## Developing with CaseMap Server

Chapter



# Developing with CaseMap Server

## *Getting Started with CaseMap Server Development*

### About CaseMap Server Development

The CaseMap Server programming environment consists of six REST methods exposed using the Windows Communication Foundation (WCF). The methods allow a client application to access basic data retrieval capabilities, such as the ability to view CaseMap spreadsheet data and get a list of users in a case.

The following topics describe how to get started with CaseMap Server development:

Topic	Description
<a href="#">What you need to know</a>	What you likely need to know before reading any further into the documentation.
<a href="#">About the CaseMap Server programming documentation</a>	A summary of the main sections of the programming documentation, and what order they should be read in.
<a href="#">About the CaseMap Server API</a>	Summarizes the way the CaseMap Server API is organized.
<a href="#">About authentication</a>	Overview of the authentication security features built into the CaseMap Server REST API.
<a href="#">Tutorial: What's the version, CaseMap?</a>	Shows two different ways to retrieve information from CaseMap Server: either directly through a browser, or programmatically.

### What you need to know

In order to successfully develop an application using the CaseMap Server API, you must have knowledge in two areas: CaseMap Server itself, WCF. You may also find it useful to be familiar with C#.

- **CaseMap**

Currently, the CaseMap REST API allows you to retrieve CaseMap spreadsheet data. Therefore, you should have a basic understanding of how to use and administer CaseMap itself before you attempt to write an application that uses these features.

- **Windows Communication Foundation (WCF)**

WCF is a Microsoft technology that allows an application to expose a programming interface on a network. CaseMap Server uses WCF to expose the entire API. Therefore, you should have a basic understanding of WCF features, such as endpoints and how to make a basic client call. Knowledge of advanced WCF features may be useful, but is not necessary to interact with the API. For more information, see Microsoft's [WCF documentation](#).

Note that the CaseMap Server development documentation is written using C#. This includes all the code samples and procedural steps. However, you are not required to program in this language: WCF allows you to make calls to exposed resources in a variety of languages. However, it will likely be easier for you to understand the documentation if you are familiar with C#.

## About the CaseMap Server programming documentation

The documentation is divided up into two sections:

- [Getting Started](#)

A quick summary of the documentation (including this topic), as well as links to other parts of the documentation. If you have never written an application for CaseMap Server, you should start here.

- [Programming Reference](#)

Includes topics on URI structure, REST verbs, as well as a discussion of request and response headers. The REST API reference contains six methods, mostly addressing how to download and view spreadsheet data. Once you have read through the Getting Started sections, you should be able to understand any page in the reference. Note that although you can write an application that interacts with CaseMap Server in a variety of scripting languages, the reference is written in C#.

## About the CaseMap Server API


The current version of the CaseMap Server API uses WCF to expose two services: CMServerAD, and CMServerLA. These two services allow you to retrieve the various cases, users, views, and spreadsheet data currently available on the server. The two services differ only in their authentication protocol.

- **CMServerAD** - Exposes a REST endpoint that is authenticated using Active Directory.
- **CMServerLA** - Exposes a REST endpoint that uses Local Authentication.

Before you use these services however, you must confirm that you have the appropriate permissions to access the server. For more information, see [About the CaseMap Server programming documentation](#).

## About authentication

You can use either Active Directory or Local Authentication protocols to access CaseMap Server REST endpoints. CaseMap Server exposes these protocols through the CaseMapAD and CaseMapLA services.

 Regardless of which service you choose to use, the information you retrieve will be the

same. Also, CaseMap server does not use sessions; instead, your call is authenticated each time you make a call.

## Active Directory

To use Active Directory authentication, make all your calls to the `/CaseMapAD/CMServerAD/` resource. Note that your user account must have Active Directory permissions on the server. If you have the appropriate permissions, the server will automatically authenticate you.

**Example:** `http://localhost/CaseMapAD/CMServerAD/version`

## Local Authentication

To use Local Authentication, make all your calls to the `/CaseMapLA/CMServerLA/` resource. When using Local Authentication, the operating system will query you for the name and password of a local admin account.

**Example:** `http://localhost/CaseMapLA/CMServerLA/version`

## Tutorial: What's the version, CaseMap?

CaseMap Server currently exposes the CMServerAD service. This service exposes CaseMap Server version, case, fact, and issue information using a series of REST-based resources. You can query the service using either your browser, or programmatically. The following tutorials show how to retrieve the CaseMap Server version numbers using both paths.

### [To retrieve the version number of CaseMap Server using a browser](#)

Paste the following URL into your browser address window, replacing "localhost" with the location of your CaseMap server, and press Enter.

<http://localhost/CaseMapAD/CMServerAD.svc/version/>

### [To retrieve the version number of CaseMap Server programmatically](#)

1. In Visual Studio, add the following using statements to your code:

```
using System.IO;
using System.Net;
using System.Web;
using System.Xml;
using System.Xml.XPath;
```

2. Create the end point that describes the version resource.

```
string serverResource = "http://localhost";
string resourceName = "/CaseMapAD/CMServerAD.svc/version";
string endPoint = serverResource + resourceName;
```

If necessary, replace "localhost" with the URL of your CaseMap Server installation.

Note that this code samples uses the Active Directory authentication (CaseMapAD). You

could optionally use local authentication, by calling /CaseMapLA/CMServerLA.svc instead. Both services provide the same data: the difference is simply which authentication protocol is used. For more information, see [About authentication](#).

3. Create the call to the server.

```
HttpWebRequest request = WebRequest.Create(endPoint) as  
HttpWebRequest;  
  
request.Method = "GET";
```

4. send the response and display the returned content.

```
HttpWebResponse response = request.GetResponse() as HttpWebResponse;  
DisplayMessage(response);  
Console.ReadLine();
```

DisplayMessage is a helper function you can use to display REST responses to the console. For more information, see [CaseMap Server Helper Methods](#).

5. Close down the connection.

```
response.Close();
```

## **CaseMap Server Programming Reference**

The following topics contain reference information on CaseMap Server.

Topic	Description
<a href="#">About URI Structure</a>	The structure of the URI's used in a call to CaseMap Server.
<a href="#">About verbs</a>	The four verbs (GET, SET, PUT, DELETE) used by CaseMap Server.
<a href="#">About request headers</a>	The request headers a client application can send to CaseMap Server.
<a href="#">About response headers</a>	The response headers sent by CaseMap Server.
<a href="#">CaseMap REST API</a>	The API reference for CaseMap Server.



## About URI Structure

The CaseMap Server API is structured around making REST calls to resources stored on a server. The location of CaseMap Server resources are declared using a URI, which adheres to the following pattern:

### Resource Content:

`<ServerResource>/[<service>/[<type>/[<resource ID>]]][?<query>][#<fragment>]`

The following table describes the elements of a CaseMap Server URI:

Element	Description
<a href="#">/(Root)</a>	Lists available services; the root also contains the properties of the service host. <b>Example:</b> <code>http://localhost/CaseMapAD/</code>
<a href="#">&lt;service&gt;</a>	Uniquely identifies a service on a service host. <b>Example:</b> <code>http://localhost/CaseMapAD/CMServerAD.svc/</code>
<a href="#">&lt;type&gt;</a>	Uniquely identifies the target resource type within a service. <b>Example:</b> <code>http://localhost/CaseMapAD/CMServerAD.svc/cases/</code>
<a href="#">&lt;resourceID&gt;</a>	The GUID of the resource. <b>Example:</b> <code>http://localhost/CaseMapAD/CMServerAD.svc/cases/case-6/data</code>
<a href="#">? &lt;query&gt;</a>	Optional constraint used to filter or narrow the selection of a resource or a collection of resources. <b>Example:</b> <code>http://localhost/CaseMapAD/CMServerAD.svc/cases/case-6/data/facts/?recordCount=5&amp;startingRecordNumber=5</code>

### **/(Root)**

#### **Description**

Lists the services provided by Lambchop6

the service host.

### Syntax

```
<serverResource>
```

### Remarks

When accessed via a browser (and with the appropriate permissions), CaseMap Server returns the root folder of the website, which contains the following list of supported services, the licence, readme, and the Web.config file.

Service	Description
CMServerAD.svc	Active directory REST endpoint for CaseMap Server. CMServerAD and CMServerLA both support the REST calls described in this guide. For more information, see <a href="#">About authentication</a> .
CMServerAdmin.svc	SOAP endpoint for the the CaseMap Server Administration console. For more information, see <a href="#">Accessing the CaseMap Admin Console</a> .
CMServerClient.svc	SOAP endpoint for CaseMap Server client applications, such as CaseMap.
CMServerLA.svc	Local authentication REST endpoint for CaseMap Server. CMServerLA and CMServerAD both support the REST calls described in this guide. For more information, see <a href="#">About authentication</a> .

### Examples

The following example requests a list of provided services from the specified server.

```
GET http://localhost/CaseMapAD/
```

**<service>**

### Description

Currently not supported.

### Syntax

```
<ServerResource>/<service>
```

## Remarks

Some REST-based services support calling the service directly; in doing so they can return information about the service, such as the names and location of various resources. However, CaseMap Server does not currently support directly calling the CMServerAD or CMServerLA services themselves. However, the service does support a number of endpoints, such as CMServerAD.svc/version or CMServerLA.svc/cases.

## Supported Methods

This level of the URI is not directly supported by any method.

## Examples

If CaseMap Server did support a method at this level, the call would likely appear as the following:

```
GET http://localhost/CaseMapAD/CMServerAD.svc/
```

**<type>**

## Description

Describes one of the objects available as a location under a service.

## Syntax

```
<ServerResource>/<service>/<type>
```

## Remarks

Types roughly correspond to objects in database or object-oriented programming. For example, a case, view, or fact is considered a type.

A caller must be authenticated to make a successful GET request on a type. If the caller is authenticated, the list of resources returned to the caller will include only the resources the caller is permitted to access. If the caller is not authenticated, only the list of resources available to anonymous callers will be returned; if the caller must be authenticated, the relevant status code will be returned.

For CaseMap Server, authentication is provided by either local authentication (through CMServerLA.svc), or else through Active Directory (through CMServerAD.svc).

### Supported Methods

The following table describes the types currently available for each service.

Service	Available Type
CMServerAD or CMServerLA	<a href="#">Version Info</a> <a href="#">Cases</a> <a href="#">Views</a> <a href="#">Users</a> <a href="#">Data</a> <a href="#">{ Spreadsheet-Name }</a> <a href="#">Issues-Outline</a>

### Examples

The following example returns a list of all of the resources of the <cases> type, as described in [GET get-cases](#).

```
GET http://localhost/CaseMapAD/CMServerAD.svc/cases/
```

### <resourceID>

### Description

Uniquely identifies a resource within a service. The <resource ID> contains all the information that is needed by the service to discover the resource type and scoping, as well as the resource itself.

### Syntax

```
<ServerResource>/[<service>/[<type>/[<resource ID>]]]
```

## Remarks

The <resource ID> has the following structure:

```
{<type name>-<instance identifier>}
```

- The <type name> is the singular form of the resource type that is not qualified by the service name. For example, if the resource type is specified by "/CMServerAD/cases", the <type name> is "case".
- The <instance identifier> is the unique identifier of the resource within the resource scoping. The <instance identifier> is in general a UUID, but may be something else (such as a friendly name or integer) in the case of legacy data.

## Supported Methods and Examples

The following sections describe the <resource ID> naming convention for each type, as well as associated examples for each.

Universal Identifier (Uid)	Description
caseID	<p>Identifies a case.</p> <p><b>Syntax:</b> case-<code>{ integer }</code></p> <p><b>Comments:</b> CaseMap assigns each case a unique number upon creation. The integer is an increase from the previous created case. You can retrieve the caseID from <a href="#">GET get-cases</a>.</p> <p><b>Example:</b> <code>http://localhost/CaseMapAD/CMServerAD.svc/cases/case-6/views/?userId=3</code></p>
userID	<p>Identifies a user.</p> <p><b>Syntax:</b> userID-<code>{ integer }</code></p> <p><b>Comments:</b> CaseMap assigns each user a unique number upon creation. The integer is an increase from the previous created user. You can retrieve the userID values with <a href="#">GET get-case-users</a>.</p> <p><b>Example:</b> <code>http://localhost/CaseMapAD/CMServerAD.svc/cases/case-6/views/?userId=3</code></p>

## ? <query>

### Description

An option that constrains or filters to narrow the selection of a resource or a collection of

resources.

## Syntax

```
<ServerResource>/[<service>/[<type>/[<resource ID>]]][?<query>]
```

## Remarks

When applied to content, the <query> may specify constraints such as a search pattern or another selection criterion.

In addition to directly constraining the content or properties, the <query> may be used to constrain or filter factors that are external to the URI but are used by the method. For example, the <query> may constrain the effect of the method to a specific authenticated caller or callers. A <query> may be thought of as extra "parameters" to the method being invoked using the URI.

The query component contains non-hierarchical data that, along with data in the path component, serves to identify a resource within the scope of the URI's scheme and naming authority (if any). The query component is indicated by the first question mark ("?") character and terminated by a number sign ("#") character or by the end of the URI.

The characters slash ("/") and question mark ("?") may represent data within the query component.

For an example of a query syntax, see the Parameters section on the following API reference pages, as listed in the Supported Methods section, below.

## Supported Methods

The following table describes the methods that use queries in CaseMap Server.

Service	Available Type
CMServerAD	<a href="#">GET get-case-views</a>
CMServerLA	<a href="#">GET get-case-facts</a>
	<a href="#">GET get-case-issues</a>

## Examples

### Request Example

The following request retrieves a query result from case-6, as described in the example section of [GET get-case-facts](#).

```
GET http://localhost/CaseMapAD/CMServerAD/cases/case-6/data/facts/?recordCo
```

## About Verbs

Verbs are the basic commands that a client application sends to CaseMap Server as part of a request. In general, a verb is sent as part of the `request.Method` header. Currently, the API supports only the GET verb, as described in the following topic.

Verb	Description
<a href="#">GET</a>	Retrieves information from the specified resource.
PUT	Updates a resource, or if unavailable creates the new resource.
POST	Creates a new resource.
DELETE	Deletes the specified resource.

### **GET**

#### **Description**

Retrieves whatever information (in the form of an entity) is identified by the Request-URI.

#### **Remarks**

Generally speaking, a GET represents a request to retrieve information from the server. For example, calling GET on the Request-URI `http://ServerNameHere.com/CaseMapAD/CMServerAD.svc/version` returns the version information of the service.

If the Request-URI refers to a data-producing process, the server will return the produced data in the response, rather than the source text of the process; the exception would be if the text happens to be the output of the process.

The GET method should not contain entity-body in the request; any request entity-body will be discarded and not processed.

## Return Codes

The following table describes the return codes common to GET actions that return information about a single resource, such as [GET get-cases](#).

Status Code	Meaning	Example
200	The input is valid.	<code>/cases/case-6/data/</code>
401	The user is unauthorized.	User does not have the correct authorization on their account.
401	The authorization header contains a locked user ID.	<code>/cases/case-6/data/</code> , when user is locked.
401, WWW-authenticate	The user has no authorization.	The user does not have an Active Directory account on the server.
403	The user or policy has been denied.	Invalid user token.
404	The user input is invalid.	<code>/cases/case-\$\$/data/</code> <code>/cases/case-0/data/</code> <code>/cases/case-@/data/</code>
404	The URI is in an invalid format.	<code>/cases/case6-61/data/</code>
404	The specified Resource ID does not exist.	<code>/cases/case-9999/data/</code> , when case 9999 doesn't exist.
404	The call is missing necessary information.	<code>/cases/case-/data/</code>
404/410	The resource has been deleted.	<code>/cases/case-6/data/</code> , when case 6 has been deleted.
405	The method type does not match up with the call.	Giving DELETE/PUT/POST instead of GET

## Supported Methods

The following table describes the GET methods available for each service:



Service	GET methods
CMServerAD	<a href="#">GET get-version-info</a>
	<a href="#">GET get-cases</a>
	<a href="#">GET get-case-views</a>
	<a href="#">GET get-case_data</a>
	<a href="#">GET get-case-users</a>
	<a href="#">GET get-case-facts</a>
	<a href="#">GET get-case-issues</a>

### Example

The following is an example of a URI used to retrieve data about a case, as described in the example section of [GET get-case\\_data](#).

```
GET http://www.serverResourceName.com/CMServerAD/cases/case-6/data/
```

### About request headers

A Request header is a header contained in a request message that a client application sends to a server. These messages contain useful and necessary information about the request. For example, the most common request header you will use in CaseMap Server is the Accept header, which contains a number of parameters for many of the method calls. Other than <Accept>, all other request headers are either optional, auto-generated, or have a default value that is useable by most applications.

The following topics describe the headers you may need to place in a CaseMap Server request.

Header	Required	Description
Accept	No	Restricts what data the client will accept in the response. Used by <a href="#">GET get-case-facts</a> and <a href="#">GET get-case-issues</a> to limit the data returned.
Content-Length	Yes (autogenerated)	The length of content provided by PUT and POST verbs.
Host	Yes (autogenerated)	The host server. Used for logging.
user-agent	Yes	The user agent. Used for logging.

(autogenerated  
)

## About response headers

The following topics discuss the common headers you may find in a response message from CaseMap Server. Currently, all the headers are merely informational, as the current CaseMap Server returns the requested information in the body of the message, rather than in any header.

Header	Description
Cache-Control	The cache control options. For CaseMap Server, this usually indicates that all or part of the response message is intended for a single user and must not be cached by a shared cache. <b>Example:</b> Cache-Control: private
Connection	Options for the connection. For CaseMap Server, this indicates that the connection is not persistent, and that you must request a new connection with each request. <b>Example:</b> Connection: closed
Content-Type	The content MIME type; usually application/xml. <b>Example:</b> Content-Type: application/xml
Content-Length	The length of the body of the message. Always returned. <b>Example:</b> Content-Length: 253
Date	The date of the creation of the message. <b>Example:</b> Date: Mon, 27 Jun 2011 20:32:28 GMT
Server	The Server type and version the message came from. <b>Example:</b> Server: Microsoft-IIS/7.5
X-Asp-Net-Version	The version of ASP.NET used by CaseMap Server. <b>Example:</b> X-Asp-Net-Version: 2.0.502727

## CaseMap REST API

The following topics describe the API exposed by CaseMap Server.

Name	Description
<a href="#">CMServerAD and CMServer Service Reference</a>	Resources for retrieving CaseMap spreadsheet data, including views, data, users, facts, and issues.
<a href="#">CaseMap Server Helper Methods</a>	A series of helper methods used in the code samples to display message information.

### ***CMServerAD and CMServer Service Reference***

This topic contains API reference material for the CMServerAD and CMServerLA services. Both services return identical data; the difference is which authentication protocol the server uses. For more information, see [About authentication](#).

Name	Description
<a href="#">GET get-version-info</a>	Retrieves the version information of the web layer assembly.
<a href="#">GET get-cases</a>	Retrieves a list of registered cases on the server.
<a href="#">GET get-case-views</a>	Retrieves a collection of views associated with the specified case.
<a href="#">GET get-case_data</a>	Retrieves a collection of spreadsheets for the specified case.
<a href="#">GET get-case-users</a>	Retrieves a collection of users assigned to a specified case
<a href="#">GET get-case-facts</a>	Retrieves a list of facts for the specified spreadsheet.
<a href="#">GET get-case-issues</a>	Retrieves a list of issues for the specified case.

## ***GET get-version-info***

### **Description**

Retrieves the version information of the web layer assembly.

To use this action, send a GET request to the /version resource.

In response to a get-version-info request, CaseMap Server returns a <version> element that contains the major, minor, and build version numbers.

### **Requests**

#### **Syntax**

```
GET {ServerNameHere}/CaseMapAD/CMServerAD/version
```

or

```
GET {ServerNameHere}/CaseMapLA/CMServerLA/version
```

#### **Headers**

The request includes the GET, User-Agent, and Host headers, which are standard for all CaseMap Server requests. For more information, see [About request headers](#).

#### **Parameters**

The URL does not contain any parameters.

#### **Content**

The request does not contain any elements in the body.

### **Responses**

#### **Headers**

The response includes the headers common to all CaseMap Server responses. For more information, see [About response headers](#).

## Content

Name	Description
version	Parent element for the response. <b>Type:</b> complex <b>Children:</b> major, minor, build
major	Major version number. <b>Type:</b> int <b>Parent:</b> version <b>Example:</b> 1
minor	Minor version number <b>Type:</b> int <b>Parent:</b> version <b>Example:</b> 6
build	Build number. <b>Type:</b> int <b>Parent:</b> version <b>Example:</b> 71

## Status Codes

Returns 200 on a valid response. Otherwise, returns one of the common status codes. for more information, see [GET](#).

## Examples

### Request Example

The following requests the version information.

```
GET /CMServerAD/version HTTP/1.1
User-Agent: Fiddler
Host: localhost:3952
```

### **C# Example**

The following example retrieves the version number of CaseMap Server. The sample takes the server URL as a parameter. The sample also uses the `DisplayMessage` helper method to display the response message. For more information on the helper methods, see [CaseMap Server Helper Methods](#).

```
public static void getServerVersion(string serverResource)
{
    //define the resource and the endpoint
    string resourceName = "/CaseMapAD/CMServerAD.svc/version";
    string endPoint = serverResource + resourceName;

    //create the call
    HttpWebRequest request = WebRequest.Create(endPoint) as HttpWebRequest;
    request.Method = "GET";

    //send it off and get the response.
    HttpWebResponse response = request.GetResponse() as HttpWebResponse;

    //display the response
    DisplayMessage(response);
    Console.ReadLine();

    response.Close();
}
```

### **Response Example**

The following is a response to the request in the previous example.

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Tue, 09 Aug 2011 21:49:42 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/xml; charset=utf-8
Content-Length: 199
Connection: Close
```

```
<version xmlns="http://services.lexisnexis.com/xmlschema/litigation-services/
casemap/1" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <major>1</major>
  <minor>6</minor>
  <build>71</build>
</version>
```

## **GET get-cases**

### **Description**

Retrieves a list of registered cases on the server.

To use this action, send a GET request to the /cases/ resource.

In response to a get-cases request, CaseMap Server returns a <caseSummaries> element. This collection contains the client matter number, the description, the ID, and the case name for each case on the server.

### **Requests**

#### **Syntax**

```
GET {ServerNameHere}/CaseMapAD/CMServerAD/cases/
```

or

```
GET {ServerNameHere}/CaseMapLA/CMServerLA/cases/
```

#### **Headers**

The request includes the GET, User-Agent, and Host headers, which are standard for all CaseMap Server requests. For more information, see [About request headers](#).

#### **Parameters**

The URL does not contain any parameters.

#### **Content**

The request does not contain any elements in the body.

### **Responses**

#### **Headers**

The response includes the headers common to all CaseMap Server responses. For more information, see [About response headers](#).

## Content

Name	Description
caseSummaries	Parent element for the response; contains a collection of case summaries. <b>Type:</b> complex <b>Children:</b> caseSummary
caseSummary	Case summary for one case. <b>Type:</b> complex <b>Parent:</b> caseSummaries <b>Children:</b> clientMatterNumber, id, name
clientMatterNumber	The client matter number. <b>Type:</b> int <b>Parent:</b> caseSummary <b>Example:</b> 1234
id	ID number <b>Type:</b> int <b>Parent:</b> caseSummary <b>Example:</b> 5678
name	Friendly name of the case. <b>Type:</b> string <b>Parent:</b> caseSummary <b>Example:</b> CMCase

## Status Codes

Returns 200 on a valid response. Otherwise, returns one of the common status codes. for more information, see [GET](#).

## Examples



### Request Example

The following requests the version information.

```
GET /CMServerAD/cases/ HTTP/1.1
User-Agent: Fiddler
Host: localhost:3952
```

### C# Example

The following example retrieves a list of all cases on the specified server. The sample takes the server URL as a parameter. The sample also uses the `DisplayMessage` helper method to display the response. For more information on the helper methods, see [CaseMap Server Helper Methods](#).

```
public static void getServerCaseViews(string serverResource)
{
    //define the resource and the endpoint
    string resourceName = "/CaseMapAD/CMServerAD.svc/cases";
    string endPoint = serverResource + resourceName;

    //create the call and add in the authorization token
    HttpWebRequest request = WebRequest.Create(endPoint) as HttpWebRequest;
    request.Method = "GET";

    //send it off and get the returned list.
    HttpWebResponse response = request.GetResponse() as HttpWebResponse;

    //display the list
    DisplayMessage(response);
    Console.ReadLine();

    response.Close();
}
```

### Response Example

The following is a response to the request in the previous example.

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Tue, 09 Aug 2011 22:04:08 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/xml; charset=utf-8
Content-Length: 386
Connection: Close
```

```
<caseSummaries xmlns="http://services.lexisnexis.com/xmlschema/litigation-services/
casemap/1" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <caseSummary>
```

```
<clientMatterNumber>1234</clientMatterNumber>
<description>5678</description>
<id>1</id>
<name>CMCase</name>
</caseSummary>
<caseSummary>
  <clientMatterNumber/>
  <description/>
  <id>6</id>
  <name>Hawkins</name>
</caseSummary>
</caseSummaries>
```

## **GET get-case-views**

### **Description**

Retrieves a collection of views associated with the specified case.

To use this action, send a GET request to the `/cases/{caseID}/views/` resource.

In response to a get-case-views request, CaseMap Server returns a `<views>` element. This collection contains the data type ID, ID, and name of all the views associated with the case. You can optionally limit the response to a particular user or data type, using the optional query parameters.

A view is a set of fields and sorting information for a particular spreadsheet.

### **Requests**

#### **Syntax**

```
GET {ServerNameHere}/CaseMapAD/CMServerAD/cases/{caseID}/views/?
userId={userID}&dataTypeId={dataTypeID}
```

or

```
GET {ServerNameHere}/CaseMapLA/CMServerLA/cases/{caseID}/views/?
userId={userID}&dataTypeId={dataTypeID}
```

#### **Headers**

The request includes the GET, User-Agent, and Host headers, which are standard for all CaseMap Server requests. For more information, see [About request headers](#).

### Parameters

The URL contains the [caseID](#). In addition, the query string contains the following search values:

Value	Description
userId	(optional) ID number of the user to retrieve the spreadsheet views for. <b>Example:</b> 3
dataTypeId	(optional) Type of data to retrieve. This integer value is returned in the ID element by <a href="#">GET get-case_data</a> . <b>Example:</b> 1

### Content

The request does not contain any elements in the body.

### Responses

#### Headers

The response includes the headers common to all CaseMap Server responses. For more information, see [About response headers](#).

#### Content

Name	Description
views	Parent element of the response; contains a collection of view objects. <b>Type:</b> complex <b>Children:</b> view
view	Description of a view the user has. <b>Type:</b> complex <b>Parent:</b> views <b>Children:</b> dataTypeId, id, name

**dataTypeId** Data type ID. If the `dataTypeId` URL parameter was used to limit the search, this element will be the same for all views.

**Type:** int

**Parent:** view

**Example:** 3

<b>id</b>	The ID of the view <b>Type:</b> int <b>Parent:</b> view <b>Example:</b> 59228293
-----------	---

**name** The name of the view.

**Type:** string

**Parent:** view

**Example:** Document - Favorite

### Status Codes

Returns 200 on a valid response. Otherwise, returns one of the common status codes. For more information, see [GET](#).

### Examples

#### Request Example

The following requests the version information.

```
GET /CMServerAD/cases/case-6/views/?userId=3 HTTP/1.1
User-Agent: Fiddler
Host: local host: 3952
```

#### C# Example

The following example retrieves the case views from the specified CaseMap Server. The sample takes the server URL, case number and `userId` as a parameter. The sample also uses the `DisplayMessage` helper method to display the response message. For more information on the helper methods, see [CaseMap Server Helper Methods](#).

```
public static void getServerCaseViews(string serverResource, string caseID,
string userID)
{
```

```

//define the resource and the endpoint
string resourceName = "/CaseMapAD/CMServerAD.svc/cases/" + caseID + "/"
views/?userId=" + userID;
string endPoint = serverResource + resourceName;

//create the call and add in the authorization token
HttpRequest request = WebRequest.Create(endPoint) as HttpRequest;
request.Method = "GET";

//send it off and get the returned list.
HttpResponse response = request.GetResponse() as HttpResponse;

//display the list
DisplayMessage(response);
Console.ReadLine();

response.Close();
}

```

### Response Example

The following is a response to the request in the previous example.

```

HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Tue, 09 Aug 2011 22:20:34 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/xml; charset=utf-8
Content-Length: 2474
Connection: Close

```

```

<views xmlns="http://services.lexisnexis.com/xmlschema/litigation-services/casemap/1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <view>
    <dataTypeId>6</dataTypeId>
    <id>59228293</id>
    <name>Document - Favorite</name>
  </view>
  <view>
    <dataTypeId>18</dataTypeId>
    <id>164510641</id>
    <name>Favorite</name>
  </view>
  <view>
    <dataTypeId>8</dataTypeId>
    <id>206474897</id>
    <name>Event - Favorite</name>
  </view>
  <view>
    <dataTypeId>10</dataTypeId>
    <id>445172537</id>
    <name>Pleading - Favorite</name>
  </view>
  <view>

```

```
<dataTypeId>18</dataTypeId>
<id>484203523</id>
<name>Most Fields</name>
</view>
<view>
  <dataTypeId>12</dataTypeId>
  <id>506016688</id>
  <name>Other Discovery - Favorite</name>
</view>
<view>
  <dataTypeId>5</dataTypeId>
  <id>541528399</id>
  <name>Organization - Favorite</name>
</view>
<view>
  <dataTypeId>2</dataTypeId>
  <id>571339841</id>
  <name>Jury Instructions</name>
</view>
<view>
  <dataTypeId>11</dataTypeId>
  <id>720547300</id>
  <name>Proceeding - Favorite</name>
</view>
<view>
  <dataTypeId>1</dataTypeId>
  <id>866746305</id>
  <name>Favorite</name>
</view>
<view>
  <dataTypeId>15</dataTypeId>
  <id>868748177</id>
  <name>Assigned To</name>
</view>
<view>
  <dataTypeId>19</dataTypeId>
  <id>882473373</id>
  <name>Most Fields</name>
</view>
<view>
  <dataTypeId>2</dataTypeId>
  <id>1105810968</id>
  <name>Demonstrative Evidence Analysis</name>
</view>
<view>
  <dataTypeId>2</dataTypeId>
  <id>1200365318</id>
  <name>Favorite</name>
</view>
<view>
  <dataTypeId>4</dataTypeId>
  <id>1363887205</id>
  <name>Person - Favorite</name>
</view>
<view>
  <dataTypeId>13</dataTypeId>
  <id>1386622048</id>
```

```
<name>Demonstrative Evidence - Favorite</name>
</view>
<view>
  <dataTypeId>1</dataTypeId>
  <id>1474848196</id>
  <name>Date, Fact and Source</name>
</view>
<view>
  <dataTypeId>14</dataTypeId>
  <id>1574936314</id>
  <name>Other Object - Favorite</name>
</view>
<view>
  <dataTypeId>4</dataTypeId>
  <id>1617081878</id>
  <name>Person - Address/Phone</name>
</view>
<view>
  <dataTypeId>9</dataTypeId>
  <id>1674014038</id>
  <name>Place - Favorite</name>
</view>
<view>
  <dataTypeId>3</dataTypeId>
  <id>1704210468</id>
  <name>All Object Types - Favorite</name>
</view>
<view>
  <dataTypeId>15</dataTypeId>
  <id>1800626038</id>
  <name>Favorite</name>
</view>
<view>
  <dataTypeId>7</dataTypeId>
  <id>1884578000</id>
  <name>Other Physical Evidence - Favorite</name>
</view>
<view>
  <dataTypeId>19</dataTypeId>
  <id>1885741421</id>
  <name>Favorite</name>
</view>
<view>
  <dataTypeId>5</dataTypeId>
  <id>2040184350</id>
  <name>Organization - Address/Phone</name>
</view>
<view>
  <dataTypeId>6</dataTypeId>
  <id>2118863299</id>
  <name>Trial Exhibit List</name>
</view>
</views>
```

## ***GET get-case-users***

### **Description**

Retrieves a collection of users assigned to a specified case

To use this action, send a GET request to the `/cases/{caseID}/users` resource.

In response to a get-case-users request, CaseMap Server returns a `<userSummaries>` element. This collection contains the first name, ID, last name, and user name of each user assigned to the case.

### **Requests**

#### **Syntax**

```
GET {ServerNameHere}/CaseMapAD/CMServerAD/{caseID}/users/
```

or

```
GET {ServerNameHere}/CaseMapLA/CMServerLA/{caseID}/users/
```

#### **Headers**

The request includes the GET, User-Agent, and Host headers, which are standard for all CaseMap Server requests. For more information, see [About request headers](#).

#### **Parameters**

The URL contains the [caseID](#).

#### **Content**

The request does not contain any elements in the body.

### **Responses**

#### **Headers**

The response includes the headers common to all CaseMap Server responses. For more information, see [About response headers](#).



**Content**

Name	Description
userSummaries	Parent element for the response, and collection of user summary elements.  <b>Type:</b> complex <b>Children:</b> userSummary
userSummary	Summary of a single user  <b>Type:</b> complex <b>Parent:</b> userSummaries <b>Children:</b> emailAddress, firstName, id, lastName, userName
emailAddress	email address.  <b>Type:</b> string <b>Parent:</b> userSummary <b>Example:</b> <a href="mailto:testuser@lexisnexis.com">testuser@lexisnexis.com</a>
firstName	First name of the user.  <b>Type:</b> string <b>Parent:</b> userSummary <b>Example:</b> Testaccount
id	ID of the user.  <b>Type:</b> int <b>Parent:</b> userSummary <b>Example:</b> 6
lastName	Last name of the user.  <b>Type:</b> string <b>Parent:</b> userSummary <b>Example:</b> Userperson
userName	Account name of the user.  <b>Type:</b> string

**Parent:** userSummary

**Example:** test\_user

### Status Codes

Returns 200 on a valid response. Otherwise, returns one of the common status codes. For more information, see [GET](#).

## Examples

### Request Example

The following requests the version information.

```
GET /CMServerAD/case-6/users/ HTTP/1.1
User-Agent: Fiddler
Host: localhost:3952
```

### C# Example

The following example retrieves the users of a specified case from the specified CaseMap Server. The sample takes the server URL and case ID as parameters. The sample also uses the DisplayMessage helper method to display the response message. For more information on the helper methods, see [CaseMap Server Helper Methods](#).

```
public static void getServerCaseUsers(string serverResource, string caseID)
{
    //define the resource and the endpoint
    string resourceName = "/CaseMapAD/CMServerAD.svc/case/" + caseID + "/
users/";
    string endPoint = serverResource + resourceName;

    //create the call and add in the authorization token
    HttpWebRequest request = WebRequest.Create(endPoint) as HttpWebRequest;
    request.Method = "GET";

    //send it off and get the returned list.
    HttpWebResponse response = request.GetResponse() as HttpWebResponse;

    //display the list
    DisplayMessage(response);
    Console.ReadLine();

    response.Close();
}
```

## Response Example

The following is a response to the request in the previous example.

```

HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Tue, 09 Aug 2011 22:10:26 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/xml; charset=utf-8
Content-Length: 647
Connection: Close

```

```

<userSummaries xmlns="http://services.lexisnexis.com/xmlschema/litigation-services/casemap/1" xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <userSummary>
    <emailAddress/>
    <firstName>Test User 1</firstName>
    <id>3</id>
    <lastName>Prapin</lastName>
    <userName>pap_user_1</userName>
  </userSummary>
  <userSummary>
    <emailAddress/>
    <firstName>Test User 3</firstName>
    <id>6</id>
    <lastName>Prapin</lastName>
    <userName>pap_user_3</userName>
  </userSummary>
</userSummaries>

```

## GET get-case-data

### Description

Retrieves a collection of of spreadsheets for the specified case.

To use this action, send a GET request to the /cases/{caseID}/data resource.

In response to a get-case-data request, CaseMap Server returns a <data> element. This collection contains the location, display sequence, ID, as well as the plural and singular names of the spreadsheet.

Common spreadsheets include the Facts, Issues, and Persons spreadsheets.

### Requests

### Syntax

```
GET {ServerNameHere}/CaseMapAD/CMServerAD/cases/{caseID}/data  
or  
GET {ServerNameHere}/CaseMapLA/CMServerLA/cases/{caseID}/data
```

### Headers

The request includes the GET, User-Agent, and Host headers, which are standard for all CaseMap Server requests. For more information, see [About request headers](#).

### Parameters

The URL contains the [caseID](#).

### Content

The request does not contain any elements in the body.

## Responses

### Headers

The response includes the headers common to all CaseMap Server responses. For more information, see [About response headers](#).

### Content

Name	Description
data	Parent element for the response; contains a collection of spreadsheet descriptions. <b>Type:</b> complex <b>Children:</b> dataItem
dataItem	Description of a spreadsheet. <b>Type:</b> complex <b>Parent:</b> data <b>Children:</b> location, id, namePlural, nameSingular

location	Relative path of the spreadsheet. <b>Type:</b> string <b>Parent:</b> data <b>Example:</b> /cases/case-6/data/facts
displaySequence	Sequence the spreadsheet is in the display. <b>Type:</b> int <b>Parent:</b> data <b>Example:</b> 1
id	Data ID. This value can be used in calls to <a href="#">GET get-case-views</a> to limit the types of data returned. <b>Type:</b> int <b>Parent:</b> data <b>Example:</b> 1
namePlural	Plural name of the spreadsheet <b>Type:</b> string <b>Parent:</b> data <b>Example:</b> Issues
nameSingular	Singular name of the spreadsheet. <b>Type:</b> string <b>Parent:</b> data <b>Example:</b> Issue

**Status Codes**

Returns 200 on a valid response. Otherwise, returns one of the common status codes. for more information, see [GET](#).

**Examples****Request Example**

The following requests the version information.

```
GET /CMServerAD/cases/case-6/data/ HTTP/1.1
```

```
User-Agent: Fiddler
Host: localhost:3952
```

### C# Example

The following example retrieves the case data from the specified CaseMap Server. The sample takes the server URL and case ID as parameters. The sample also uses the `DisplayMessage` helper method to display the response message. For more information on the helper methods, see [CaseMap Server Helper Methods](#).

```
public static void getServerCaseData(string serverResource, string caseID)
{
    //define the resource and the endpoint
    string resourceName = "/CaseMapAD/CMServerAD.svc/case/" + caseID + "/
data/";
    string endPoint = serverResource + resourceName;

    //create the call and add in the authorization token
    HttpWebRequest request = WebRequest.Create(endPoint) as HttpWebRequest;
    request.Method = "GET";

    //send it off and get the returned list.
    HttpWebResponse response = request.GetResponse() as HttpWebResponse;

    //display the list
    DisplayMessage(response);
    Console.ReadLine();

    response.Close();
}
```

### Response Example

The following is a response to the request in the previous example.

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Tue, 09 Aug 2011 22:14:00 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/xml; charset=utf-8
Content-Length: 3721
Connection: Close
```

```
<data xmlns="http://services.lexisnexis.com/xmlschema/litigation-services/casemap/1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <dataItem>
    <location>/cases/case-6/data/facts</location>
    <displaySequence>1</displaySequence>
    <id>1</id>
    <namePlural>Facts</namePlural>
    <nameSingular>Fact</nameSingular>
  </dataItem>
</dataItem>
```

```

    <location>/cases/case-6/data/issues</location>
    <displaySequence>2</displaySequence>
    <id>2</id>
    <namePlural>Issues</namePlural>
    <nameSingular>Issue</nameSingular>
  </dataItem>
</dataItem>
  <location>/cases/case-6/data/all-objects</location>
  <displaySequence>3</displaySequence>
  <id>3</id>
  <namePlural>All Objects</namePlural>
  <nameSingular>All Objects</nameSingular>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/persons</location>
  <displaySequence>4</displaySequence>
  <id>4</id>
  <namePlural>Persons</namePlural>
  <nameSingular>Person</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/organizations</location>
  <displaySequence>5</displaySequence>
  <id>5</id>
  <namePlural>Organizations</namePlural>
  <nameSingular>Organization</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/documents</location>
  <displaySequence>6</displaySequence>
  <id>6</id>
  <namePlural>Documents</namePlural>
  <nameSingular>Document</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/other-physical-evidence</location>
  <displaySequence>7</displaySequence>
  <id>7</id>
  <namePlural>Other Physical Evidence</namePlural>
  <nameSingular>Other Physical Evidence</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/events</location>
  <displaySequence>8</displaySequence>
  <id>8</id>
  <namePlural>Events</namePlural>
  <nameSingular>Event</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/places</location>
  <displaySequence>9</displaySequence>
  <id>9</id>

```

```
<namePlural>Places</namePlural>
<nameSingular>Place</nameSingular>
<parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/pleadings</location>
  <displaySequence>10</displaySequence>
  <id>10</id>
  <namePlural>Pleadings</namePlural>
  <nameSingular>Pleading</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/proceedings</location>
  <displaySequence>11</displaySequence>
  <id>11</id>
  <namePlural>Proceedings</namePlural>
  <nameSingular>Proceeding</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/other-discovery</location>
  <displaySequence>12</displaySequence>
  <id>12</id>
  <namePlural>Other Discovery</namePlural>
  <nameSingular>Other Discovery</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/demonstrative-evidence</location>
  <displaySequence>13</displaySequence>
  <id>13</id>
  <namePlural>Demonstrative Evidence</namePlural>
  <nameSingular>Demonstrative Evidence</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/other-objects</location>
  <displaySequence>14</displaySequence>
  <id>14</id>
  <namePlural>Other Objects</namePlural>
  <nameSingular>Other Object</nameSingular>
  <parentID>3</parentID>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/questions</location>
  <displaySequence>15</displaySequence>
  <id>15</id>
  <namePlural>Questions</namePlural>
  <nameSingular>Question</nameSingular>
</dataItem>
<dataItem>
  <location>/cases/case-6/data/authorities</location>
  <displaySequence>16</displaySequence>
  <id>18</id>
  <namePlural>Authorities</namePlural>
  <nameSingular>Authority</nameSingular>
```



```

</dataItem>
<dataItem>
  <location>/cases/case-6/data/extracts-from-authorities</location>
  <displaySequence>17</displaySequence>
  <id>19</id>
  <namePlural>Extracts from Authorities</namePlural>
  <nameSingular>Authority Extract</nameSingular>
</dataItem>
</data>

```

## ***GET get-case-{spreadsheet-name}***

### **Description**

Retrieves data from the specified spreadsheet.

To use this action, send a GET request to the `/cases/{caseID}/data/{spreadsheet-name}/` resource. You identify both the view and name format options in the Accept header.

In response to a `get-case-{spreadsheet-name}` request, CaseMap Server returns a `<data>` element, which contains the starting record number, the record number, total record number, and an XML representation of a .NET table object. If Page and PageSize are not specified, all records are returned. Optionally, a view and format of object names may be specified via the Accept header. Object names may be Full or Short. If not specified, short names will be returned.

### **Requests**

#### **Syntax**

```
GET {ServerNameHere}/CaseMapAD/CMServerAD/cases/{caseID}/data/facts/?
recordCount={recordCount}&startingRecordNumber={recordNumber}
```

or

```
GET {ServerNameHere}/CaseMapLA/CMServerLA/cases/{caseID}/data/facts/?
recordCount={recordCount}&startingRecordNumber={recordNumber}
```

#### **Headers**

The request includes the GET, User-Agent, and Host headers, which are standard for all CaseMap Server requests. For more information, see [About request headers](#). In addition, the method uses the Accept header to define the limits on the search query, as described in the following table.

Name	Description						
Accept	Contains a number of limits on the search query. <b>Type:</b> string						
	<table border="1"> <thead> <tr> <th>Name</th> <th>Description</th> </tr> </thead> <tbody> <tr> <td>nameFormat</td> <td>(optional) format for the name elements. Must be one of the <ul style="list-style-type: none"> <li>• full</li> <li>• short</li> </ul> The default value is short.</td> </tr> <tr> <td>viewID</td> <td>(optional) ID of the spreadsheet view to retrieve. The view c both the fields included and the sort order of the records. Th is to return all non-virtual fields and sort the records by reco</td> </tr> </tbody> </table>	Name	Description	nameFormat	(optional) format for the name elements. Must be one of the <ul style="list-style-type: none"> <li>• full</li> <li>• short</li> </ul> The default value is short.	viewID	(optional) ID of the spreadsheet view to retrieve. The view c both the fields included and the sort order of the records. Th is to return all non-virtual fields and sort the records by reco
Name	Description						
nameFormat	(optional) format for the name elements. Must be one of the <ul style="list-style-type: none"> <li>• full</li> <li>• short</li> </ul> The default value is short.						
viewID	(optional) ID of the spreadsheet view to retrieve. The view c both the fields included and the sort order of the records. Th is to return all non-virtual fields and sort the records by reco						

**Example:** Accept: nameFormat=full ; viewID=1474848196

### Parameters

The URL contains the [caseID](#). In addition, the query string contains the following parameters:

Name	Description
recordCount	The record count. <b>Example:</b> 5
startingRecordNumber	Starting record Number <b>Example:</b> 3
PageSize&PageNum	(Optional) Paging support to return the number of records in each call <b>Example:</b> http://server:port/servicename.svc/cases/{case-[ID]}/data/facts?pagesize=[PGSIZE]&page=[PGNUM]#viewid=[ID]

### Content

The request does not contain any elements in the body.

## Responses

### Headers

The response includes the headers common to all CaseMap Server responses. For more information, see [About response headers](#).

### Content

Name	Description
data	parent element for the response <b>Type:</b> complex <b>Children:</b> startingRecordNumber, recordCount, totalRecordCount, tableData
startingRecordNumber	The starting record number. <b>Type:</b> int <b>Parent:</b> data <b>Example:</b> 5
recordCount	Number of records returned <b>Type:</b> int <b>Parent:</b> data <b>Example:</b> 5
totalRecordCount	Total number of records. <b>Type:</b> int <b>Parent:</b> data <b>Example:</b> 29
tableData	XML representation of a .NET table <b>Type:</b> complex For more information, see the .NET Framework documentation.

## Status Codes

Returns 200 on a valid response. Otherwise, returns one of the common status codes. for more information, see [GET](#).

## Examples

### Request Example

The following requests the version information.

```
GET /CMServerAD/cases/case-6/data/facts/?recordCount=5&startingRecordNumber=5 HTTP/1.1
User-Agent: Fiddler
Host: localhost:3952
Accept: nameFormat=full;viewId=1474848196
```

### C# Example

The following example retrieves the facts of a specified case from the specified CaseMap Server. The sample takes the server URL, case ID, recordCount, and starting record number as parameters. The sample also uses the DisplayMessage helper method to display the response message. For more information on the helper methods, see [CaseMap Server Helper Methods](#).

```
public static void getServerCaseFacts(string serverResource, string caseID,
string recordCount, string startingRecordNumber)
{
    //define the resource and the endpoint
    string resourceName = "/CaseMapAD/CMServerAD.svc/case/" + caseID + "/
data/facts/?recordCount="
        + recordCount + "&startingRecordNumber=" + startingRecordNumber;
    string endPoint = serverResource + resourceName;

    //create the call
    HttpWebRequest request = WebRequest.Create(endPoint) as HttpWebRequest;
    request.Method = "GET";

    //add in the additional header values
    request.Accept = "nameFormat=full;viewId=1474848196";

    //send it off and get the response
    HttpWebResponse response = request.GetResponse() as HttpWebResponse;

    //display the list
    DisplayMessage(response);
    Console.ReadLine();

    response.Close();
}
```

## Response Example

The following is a response to the request in the previous example.

```
HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Tue, 09 Aug 2011 22:40:10 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/xml; charset=utf-8
Content-Length: 6537
Connection: Close
```

```
<data xmlns="http://services.lexisnexis.com/xmlschema/litigation-services/casemap/1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <startingRecordNumber>5</startingRecordNumber>
  <recordCount>5</recordCount>
  <totalRecordCount>29</totalRecordCount>
  <tableData>
    <xs:schema id="NewDataSet" xmlns:xs="http://www.w3.org/2001/XMLSchema" xmlns=""
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
      <xs:element name="NewDataSet" msdata:IsDataSet="true"
msdata:MainDataTable="facts" msdata:UseCurrentLocale="true">
        <xs:complexType>
          <xs:choice minOccurs="0" maxOccurs="unbounded">
            <xs:element name="facts">
              <xs:complexType>
                <xs:sequence>
                  <xs:element name="F0" msdata:Caption="ID" type="xs:int"/>
                  <xs:element name="F1" msdata:Caption="Date & Time"
minOccurs="0">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:maxLength value="40"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                  <xs:element name="F2" msdata:Caption="Fact Text" minOccurs="0">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:maxLength value="2147483647"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                  <xs:element name="F3" msdata:Caption="Source(s)" minOccurs="0">
                    <xs:simpleType>
                      <xs:restriction base="xs:string">
                        <xs:maxLength value="2147483647"/>
                      </xs:restriction>
                    </xs:simpleType>
                  </xs:element>
                </xs:sequence>
              </xs:complexType>
            </xs:element>
          </xs:choice>
        </xs:complexType>
      </xs:element>
    </xs:schema>
    <diffgr:diffgram xmlns:diffgr="urn:schemas-microsoft-com:xml-diffgram-v1"
xmlns:msdata="urn:schemas-microsoft-com:xml-msdata">
```

```

<DocumentElement xmlns="">
  <facts diffgr:id="facts1" msdata:rowOrder="0">
    <F0>1038690095</F0>
    <F1>2003-12-01</F1>
    <F2>Philip Hawkins promoted to Anstar Biotech Industries VP of Sales.</F2>
    <F3>InterviewNotes</F3>
  </facts>
  <facts diffgr:id="facts2" msdata:rowOrder="1">
    <F0>774696773</F0>
    <F1>2004-01-09 to 2004-01-21</F1>
    <F2>Philip Hawkins negotiates draft Hawkins Employment Agreement with
William Lang.</F2>
    <F3>Hawkins Employment Agreement</F3>
  </facts>
  <facts diffgr:id="facts3" msdata:rowOrder="2">
    <F0>1461490321</F0>
    <F1>2004-02-??</F1>
    <F2>William Lang tells Philip Hawkins that he has changed his mind regarding
the Hawkins Employment Agreement. It is not in force as it was never signed and
changes were not finalized.</F2>
    <F3>Philip Hawkins, Deposition of William Lang, 11:3.</F3>
  </facts>
  <facts diffgr:id="facts4" msdata:rowOrder="3">
    <F0>383478900</F0>
    <F1>2004-03-??</F1>
    <F2>Susan Sheridan has dinner with Linda Collins and complains about Anstar
Biotech Industries management.</F2>
    <F3>Deposition of Linda Collins, 33:15.</F3>
  </facts>
  <facts diffgr:id="facts5" msdata:rowOrder="4">
    <F0>1902624760</F0>
    <F1>2005-05-11</F1>
    <F2>Philip Hawkins receives Hawkins Performance Review from William Lang.
Is rated a 1 "Outstanding Performer."</F2>
    <F3>Hawkins Performance Review</F3>
  </facts>
</DocumentElement>
</diffgr:diffgram>
</tableData>
</data>

```

## GET get-case-issues-outline

### Description

Retrieves a list of issues for the specified case.

To use this action, send a GET request to the `/cases/{caseID}/data/issues` resource. You specify the outline format option in the Accept header.

In response to a get-case-issues request, CaseMap Server returns an `<issues>` element. This collection contains a description of all the issues in the the case, including ID, name, outline number, short name, and any subissues.

## Requests

### Syntax

```
GET {ServerNameHere}/CaseMapAD/CMServerAD/cases/{caseID}/data/
```

or

```
GET {ServerNameHere}/CaseMapLA/CMServerLA/cases/{caseID}/data/
```

### Headers

The request includes the GET, User-Agent, and Host headers, which are standard for all CaseMap Server requests. For more information, see [About request headers](#). In addition, the Accept header can contain the following information:

Name	Description						
Accept	Contains the formatting and spreadsheet (viewID) information Type: string						
	<table border="1"> <thead> <tr> <th data-bbox="488 1125 639 1159">Name</th> <th data-bbox="639 1125 1466 1159">Description</th> </tr> </thead> <tbody> <tr> <td data-bbox="488 1159 639 1402">nameFormat</td> <td data-bbox="639 1159 1466 1402">                             (Optional) format for the name. Must be one of the following:                             <ul style="list-style-type: none"> <li>full</li> <li>short</li> </ul>                             The default value is full.                         </td> </tr> <tr> <td data-bbox="488 1402 639 1570">view</td> <td data-bbox="639 1402 1466 1570">                             (Optional) Type of view to retrieve. Must be one of the following:                             <ul style="list-style-type: none"> <li>outline</li> </ul>                             If this is omitted, the issue data is returned the same way as a spreadsheet.                         </td> </tr> </tbody> </table>	Name	Description	nameFormat	(Optional) format for the name. Must be one of the following: <ul style="list-style-type: none"> <li>full</li> <li>short</li> </ul> The default value is full.	view	(Optional) Type of view to retrieve. Must be one of the following: <ul style="list-style-type: none"> <li>outline</li> </ul> If this is omitted, the issue data is returned the same way as a spreadsheet.
Name	Description						
nameFormat	(Optional) format for the name. Must be one of the following: <ul style="list-style-type: none"> <li>full</li> <li>short</li> </ul> The default value is full.						
view	(Optional) Type of view to retrieve. Must be one of the following: <ul style="list-style-type: none"> <li>outline</li> </ul> If this is omitted, the issue data is returned the same way as a spreadsheet.						

**Example:** Accept: view=outline; nameFormat=short

### Parameters

The URL contains the [caseID](#).

## Content

The request does not contain any elements in the body.

## Responses

### Headers

The response includes the headers common to all CaseMap Server responses. For more information, see [About response headers](#).

### Content

Name	Description
issues	Parent element for the response; contains a collection of issue elements. <b>Type:</b> complex <b>Children:</b> issue
issue	Description of a single issue. <b>Type:</b> complex <b>Parent:</b> issues <b>Children:</b> id, name, outlineNumber, shortName, subIssues
id	ID of the issue. <b>Type:</b> int <b>Parent:</b> issue <b>Example:</b> 1769765926
name	Name of the issue <b>Type:</b> string <b>Parent:</b> issue <b>Example:</b> Wrongful Termination
outlineNumber	Outline number. <b>Type:</b> int



**Parent:** issue**Example:** 1

shortName	The short name of the issue <b>Type:</b> string <b>Parent:</b> issue <b>Example:</b> WrongfulTermination
subIssues	Additional issues organized under the current issue <b>Type:</b> complex <b>Parent:</b> issue <b>Children:</b> issue

### Status Codes

Returns 200 on a valid response. Otherwise, returns one of the common status codes. for more information, see [GET](#).

### Examples

#### Request Example

The following requests the version information.

```
GET /CMServerAD/cases/case-6/data/issues/ HTTP/1.1
User-Agent: Fiddler
Host: local host: 3952
Accept: view=outline
```

#### C# Example

The following example retrieves the issues of a specified case from the specified CaseMap Server. The sample takes the server URL and case ID as parameters. The sample also uses the DisplayMessage helper method to display the response message. For more information on the helper methods, see [CaseMap Server Helper Methods](#).

```
public static void getServerCaseIssues(string serverResource, string
caseID, string recordCount, string startingRecordNumber)
{
    //define the resource and the endpoint
    string resourceName = "/CaseMapAD/CMServerAD.svc/case/" + caseID + "/
data/issues/
    string endPoint = serverResource + resourceName;
```

```

//create the call
HttpRequest request = WebRequest.Create(endPoint) as HttpRequest;
request.Method = "GET";

//add in the additional header values
request.Accept = "view=outline;nameFormat=short";

//send it off and get the response
HttpResponse response = request.GetResponse() as HttpResponse;

//display the list
DisplayMessage(response);
Console.ReadLine();

response.Close();
}

```

### Response Example

The following is a response to the request in the previous example.

```

HTTP/1.1 200 OK
Server: ASP.NET Development Server/9.0.0.0
Date: Tue, 09 Aug 2011 23:26:18 GMT
X-AspNet-Version: 2.0.50727
Cache-Control: private
Content-Type: application/xml; charset=utf-8
Content-Length: 1895
Connection: Close

```

```

<issues xmlns="http://services.lexisnexis.com/xmlschema/litigation-services/casemap/1"
xmlns:i="http://www.w3.org/2001/XMLSchema-instance">
  <issue>
    <id>1769765926</id>
    <name>Wrongful Termination</name>
    <outlineNumber>1</outlineNumber>
    <shortName>WrongfulTermination</shortName>
    <subIssues/>
  </issue>
  <issue>
    <id>1554959045</id>
    <name>Age Discrimination</name>
    <outlineNumber>2</outlineNumber>
    <shortName>AgeDiscrimination</shortName>
    <subIssues>
      <issue>
        <id>1982336477</id>
        <name>Hawkins Specific</name>
        <outlineNumber>2.1</outlineNumber>
        <shortName>HawkinsSpecific</shortName>
        <subIssues/>
      </issue>
      <issue>
        <id>1491271512</id>
        <name>Pattern & Practice</name>
        <outlineNumber>2.2</outlineNumber>
        <shortName>Pattern&Practice</shortName>
        <subIssues/>
      </issue>
    </subIssues>
  </issue>

```

```

    </issue>
  </subIssues>
</issue>
<issue>
  <id>182509757</id>
  <name>Retaliation</name>
  <outlineNumber>3</outlineNumber>
  <shortName>Retaliation</shortName>
  <subIssues>
    <issue>
      <id>473045427</id>
      <name>Transfer</name>
      <outlineNumber>3.1</outlineNumber>
      <shortName>Transfer</shortName>
      <subIssues/>
    </issue>
    <issue>
      <id>1339773059</id>
      <name>Demotion</name>
      <outlineNumber>3.2</outlineNumber>
      <shortName>Demotion</shortName>
      <subIssues/>
    </issue>
  </subIssues>
</issue>
<issue>
  <id>1290855574</id>
  <name>Deserved Termination</name>
  <outlineNumber>4</outlineNumber>
  <shortName>DeservedTermination</shortName>
  <subIssues/>
</issue>
<issue>
  <id>323638003</id>
  <name>Damages</name>
  <outlineNumber>5</outlineNumber>
  <shortName>Damages</shortName>
  <subIssues>
    <issue>
      <id>1725246657</id>
      <name>Failure to Mitigate</name>
      <outlineNumber>5.1</outlineNumber>
      <shortName>FailureToMitigate</shortName>
      <subIssues/>
    </issue>
    <issue>
      <id>1666049993</id>
      <name>Lost Wages</name>
      <outlineNumber>5.2</outlineNumber>
      <shortName>LostWages</shortName>
      <subIssues/>
    </issue>
    <issue>
      <id>777908468</id>
      <name>Mental Anguish</name>
      <outlineNumber>5.3</outlineNumber>
      <shortName>MentalAnguish</shortName>

```

```
<subIssues/>
</issue>
</subIssues>
</issue>
</issues>
```

## CaseMap Server Helper Methods

The following methods are used in the code samples throughout this reference.

### DisplayMethod

This method displays the return codes and XML body of a message. DisplayMethod is used by all methods to display the returned messages.

```
using System.Xml;
using System.Xml.XPath;

public static void DisplayMessage(HttpWebResponse response)
{
    //Write out the status code number and name
    Console.WriteLine((int)response.StatusCode + " " + response.StatusCode);

    //stream the body of the message into an XML reader.
    XmlTextReader reader = new XmlTextReader(response.GetResponseStream());
    reader.WhitespaceHandling = WhitespaceHandling.None;

    //try to load the XML into an XML doc and display it.
    XmlDocument xd = new XmlDocument();
    try
    {
        xd.Load(reader);
        XmlNode xnodDE = xd.DocumentElement;

        ChildDisplay(xnodDE, 0);
        reader.Close();
    }
    //if anything goes wrong, return.
    catch
    {
        return;
    }
}

private static void ChildDisplay(XmlNode xnod, int level)
{
    XmlNode xnodWorking;
    String pad = new String(' ', level * 2);
```

```
    Console.WriteLine(pad + xnod.Name + "(" + xnod.NodeType.ToString() + ":"
" + xnod.Value + ")");

    if (xnod.NodeType == XmlNodeType.Element)
    {
        XmlNamedNodeMap mapAttributes = xnod.Attributes;
        for (int i = 0; i < mapAttributes.Count; i++)
        {
            Console.WriteLine(pad + " " + mapAttributes.Item(i).Name + " =
" + mapAttributes.Item(i).Value);
        }
    }

    if (xnod.HasChildNodes)
    {
        xnodWorking = xnod.FirstChild;
        while (xnodWorking != null)
        {
            ChildDisplay(xnodWorking, level + 1);
            xnodWorking = xnodWorking.NextSibling;
        }
    }
}
```