

RES Workspace Manager 2012

# Getting Started with RES Workspace Manager 2012 SR3 Reporting Services



# Contents

<b>Chapter 1:</b>	<b>Introduction</b>	<b>1</b>
<hr/>		
<b>Chapter 2:</b>	<b>Installation</b>	<b>2</b>
2.1	Prerequisites .....	2
2.2	Installation .....	2
<hr/>		
<b>Chapter 3:</b>	<b>Integrating RES Reporting Services in Microsoft Visual Studio</b>	<b>3</b>
<hr/>		
<b>Chapter 4:</b>	<b>Reporting Services WebAPI</b>	<b>6</b>
4.1	Reporting Service Reference Tables .....	7
4.1.1	Reports for Usage Tracking Viewer tab: Details .....	7
4.1.2	Reports for Usage Tracking Viewer tab: Application .....	10
4.1.3	Reports for Usage Tracking Viewer tab: OU .....	11
4.1.4	Reports for Usage Tracking Viewer: User .....	12
4.1.5	Reports for Usage Tracking Viewer tab: Web Site .....	14
4.1.6	Reports for Usage Tracking Viewer: Computer .....	15
4.1.7	Reports for Usage Tracking Viewer tab: Sessions .....	17
4.1.8	Reports for Usage Tracking Viewer tab: Current Activity.....	18
4.1.9	Search Methods.....	20

## Chapter 1: Introduction

Reporting Services is an (open) Web service that allows the retrieval of Usage Tracking data from the Datastore. The Usage Tracking feature is available in the Delegation and Compliance module of RES Workspace Manager.

This data can be used to create reports tailored to your organization's needs, without clients needing to have direct access to the Datastore. Clients can be anything from users, other Web applications to other Web services.

The information to be examined can focus on the following areas:

- Usage Tracking details
- User metrics
- OU metrics
- Application metrics
- Web Site metrics
- Computer metrics
- Sessions metrics
- Current Activity metrics

To help specify search criteria, a number of search methods can be used:

- Search User
- Search OU
- Search Application
- Search Web Site
- Search Computer

RES Workspace Manager 2012 SR3 Reporting Services needs to be installed separately by someone with sufficient Administrative rights.

This document guides you through the installation of RES Workspace Manager 2012 SR3 Reporting Services. To show you how to address the Web service, an integration example is given for Microsoft Visual studio (with a sample code walk-through). The methods that can be used to call the data in the WebAPI (including filters, limiting results and sort order) and the results themselves are described in the Reporting Service Reference Tables.

## Chapter 2: Installation

This chapter covers the installation of RES Workspace Manager Reporting Services.

### 2.1 Prerequisites

- RES Workspace Manager 2012 SR3 or higher, with the Agent connected directly to the Datastore.
- A tool that can generate SOAP messages (for example, Visual Studio).
- Microsoft .NET Framework 4.
- Usage Tracking data must be available in the Datastore to get results from the Reporting Service.

### 2.2 Installation

1. Install the file `RES-WM-2012-Reporting-Services-SR3.msi` on a server that can connect to the RES Datastore.
2. Follow the installation prompts.
3. After installing RES Workspace Manager Reporting Services, you can change the configuration settings, such as the localhost port number, by editing the `reswmrs.exe.config` file. See the section [Integrating RES Web Services in Microsoft Visual Studio](#) (on page 3).




#### Note

- The installation adds the directory `C:\%Program files%\RES Workspace Manager Reporting Services\Web Services`.
- `Reswmrs.exe` will be started as a Microsoft Windows Service.

## Chapter 3: Integrating RES Reporting Services in Microsoft Visual Studio

Microsoft Visual Studio languages access the API through objects that serve as proxies for their server-side counterparts. Before using the API, you must first generate these objects from your server's WSDL (Web Services Description Language) file. The WSDL describes the public interface for the Web service.

Microsoft Visual Studio provides two approaches to import your WSDL file and to generate a Web service client: an IDE-based approach and a command line approach.


 **Notes**

- Before you begin, you first need to create a new application or open an existing application in Visual Studio.
- After the installation of RES Workspace Manager 2012 SR3 Reporting Services, check whether the **RES Workspace Manager Reporting Services** has been started correctly. To verify this, on the server running the Reporting Services, navigate to <http://localhost:8732/RES/WorkspaceManager/API?WSDL>.
- If the WSDL needs to be read from another location than `localhost`, this location needs to be changed manually in the `reswmrs.exe.config` file:
  - Change `localhost` to the correct host name in the `Add baseAddress` line.
  - Make sure you restart the service to ensure the new settings are applied (`net stop RESWMRS`, `net start RESWMRS` from the command line or use Computer Management to restart the **RES Workspace Manager Reporting Services** service).

A Web service client is any component or application that references and uses a Web service. This does not necessarily need to be a client-based application. In fact, in many cases, your Web service clients might be other Web applications, such as Web Forms or even other Web services. When accessing Web services in managed code, a proxy class and the .NET Framework handle all of the infrastructure coding.

### Adding a Service reference in Visual Studio 2010

1. On the **Project** menu, choose **Add Service Reference**.
2. In the **URL** box of the **Add Service Reference** dialog box, type the URL to obtain the service description of the Web service you want to access, for example:  
<http://localhost:8732/RES/WorkspaceManager/API?WSDL>
3. Click **Go** to retrieve information about the Web service.
4. In the **Namespace** box, rename the Namespace, for example, to `WMReporting`, which is the name you will use for this service reference.
5. Click **OK** to add a Service reference for the target Web service. For more information, see the topic **Adding and Removing Service References** in the Visual Studio documentation.
6. Visual studio retrieves the service description and generates a proxy class to interface between your application and the Reporting Service.

 **Note**

- You can change the value of the `maxBufferSize` in the configuration file of your application to prevent errors when receiving large amounts of data from the Reporting Service.

### Walk through the sample code

(example of how to integrate Reporting Services in Microsoft Visual Studio)

```
using System;

using System.Collections.Generic;

using System.Linq;

using System.Text;

using ConsoleApplication3.WMReporting;

namespace ConsoleApplication3

{

    class Program

    {

        static void Main(string[] args)

        {

// Initialize the client

            PFApiClient apiClient = new PFApiClient();

            // Initialize the settings

            DetailsSettings settings = new DetailsSettings();

            // Define the filters

            List<DetailsSettings.Filter> filters = new

List<DetailsSettings.Filter>();

            DetailsSettings.Filter filter = new DetailsSettings.Filter();

            filter.ColumnName = DetailMetricsFields.Application;

            filter.Operator = "like";

            filter.Value = "*a*";

            // Add the filter to the list

            filters.Add(filter);

            // Define the sort order

            List<DetailsSettings.Sort> sorts = new

List<DetailsSettings.Sort>();

            DetailsSettings.Sort sort = new DetailsSettings.Sort();
```

```
sort.ColumnName = DetailMetricsFields.CPUUsage;
sort.Order = DetailsSettings.SortOrder.Descending;
//Add the sort order to the list
sorts.Add(sort);
settings.Sorts = sorts.ToArray();
settings.Filters = filters.ToArray();

DetailsResponse response = apiClient.Details(settings);
if (response.Errors.Count() > 0)
{
    foreach (string error in response.Errors)
    {
        Console.WriteLine("ERROR: " + error);
    }
}
else
{
    foreach (DetailsResponseRow row in response.Results)
    {
        Console.WriteLine(string.Format("{0} | {1} | {2} | {3}",
row.Application, row.User, row.CPUUsage.ToString(), row.OU));
    }
}
Console.ReadLine();
}
}
}
```

## Chapter 4: Reporting Services WebAPI

This chapter describes the methods that can be used to access the RES Workspace Manager 2012 WebAPI. To use the WebAPI, you should have a basic familiarity with software development and SOAP Web services. The actions are very similar to the functionality of the **Usage Tracking Viewer** tabs in the Console.

### Common elements

Although each method returns specific combinations of Usage Tracking information, all methods use a common set of parameters to specify the:

- filter selection
- sort order
- returned data elements
- any limitation of the amount of data to be returned
- error codes

### Filter comparers

All methods have a filter that restricts the data to be returned. The filter criteria are specific for each method (specified in the [Reporting Service Reference Tables](#) (on page 7) below), but the comparers are the same throughout.

### Operands in text-based filters

Name	Description
=	Equal to
<>	Other than
LIKE	Like, for example: "John"

### Operands in numeric filters

Name	Description
=	Equal to
>	Greater than
<	Less than
<=	Less than or equal to
>=	Greater than or equal to
<>	Other than
LIKE	Like, for example: "John"

### Limit Results



Some methods have a parameter `LimitResults` that can be set at a valid number in order to limit the number of results returned.

### Sort order

Each method returns data (results) in columns (specified in the [Reporting Service Reference Tables](#) (on page 7)). In some methods, the returned data can be sorted by combining the results with a sorting parameter `Ascending` or `Descending`.

Name	Description
Descending	Sort column descending
Ascending	Sort column ascending

### Errors

Each method returns a number that indicates whether the request succeeded or failed.

Name	Description
0	Success
1	Failure
2	Invalid parameter

### Results

The results include a header with information about:

- Date and time of results generated
- How long it took to generate the results

## 4.1 Reporting Service Reference Tables

This chapter describes the areas from which you can retrieve Usage Tracking data (e.g. details, users, OUs, applications, Web sites, computers, etc.) followed by the methods that can be applied (e.g. all applications for OU, all Web sites for OU, etc.), the filters you can use and the results from the call. Five Search methods are explained to retrieve the data. These are very similar to the search functionality in the **Usage Tracking Viewer**.

### 4.1.1 Reports for Usage Tracking Viewer tab: Details

#### Usage Tracking Details

Data shown on the **Details** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following method:

Graph type/Report name	Method
Usage Tracking Details	Details

### Filter

Column name	Expected input	Expected input format
Application	Application name	Text
Application ID	Application ID	Number
StartDateTime	Date and time	yyyy-MM-dd HH:mm:ss
ComputerName	Domain/Computer	Text
TSclientIP	IP address	IP address format (numbers in segments with separators)
ClientName	Client name	Text
UserName	Domain/user name	Text
CanonicalOU	Canonical name	Text
EndDateTime	Date and time	yyyy-MM-dd HH:mm:ss
TimeOpen	Time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Time	HHH:mm:ss (HHH can exceed 24 hours)
Memory	Value in Kb	Number
CPUUsage	Time	HHH:mm:ss (HHH can exceed 24 hours)
CriticalCPUloadduration	Time	HHH:mm:ss
Sorts	DetailMetrics.Fields	-

## Results

Column name	Description	Format
Application	Application name	Text
Application ID	Application ID	Number
StartDateTime	Date and time	yyyy-MM-dd HH:mm:ss
ComputerName	Domain/Computer	Text
TSclientIP	IP address	IP address format (numbers in segments with separators)
ClientName	Client name	Text
UserName	Domain/user name	Text
CanonicalOU	Canonical name	Text
EndDateTime	Date and time	yyyy-MM-dd HH:mm:ss
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
Memory	Value in Kb	Number
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CriticalCPUloadduration	Length of time	HHH:mm:ss

## Usage Tracking DetailsWithExePath

Data shown on the **Details** tab from the **Usage Tracking Viewer** including the **Log path and executable in addition to application name** (if enabled in RES Workspace Manager at **Setup > Usage Tracking**) can be accessed through the Web service using the following method:

Graph type/Report name	Method
Usage Tracking Details	DetailsWithExePath

## Filter

Column name	Expected input	Expected input format
Application ID	Application name	Text
StartDateTime	Date and time	yyyy-MM-dd HH:mm:ss
ComputerName	Domain\Computer	Text
TSclientIP	IP address	IP address format (numbers in segments with separators)
ClientName	Client name	Text
UserName	Domain\user name	Text
CanonicalOU	Canonical name	Text
EndDateTime	Date and time	yyyy-MM-dd HH:mm:ss
TimeOpen	Time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Time	HHH:mm:ss (HHH can exceed 24 hours)
Memory	Value in Kb	Number
CPUUsage	Time	HHH:mm:ss (HHH can exceed 24 hours)
CriticalCPUloadduration	Time	HHH:mm:ss
Sorts	DetailMetrics.Fields	-

## Results

Column name	Description	Format
Application ID	Application name	Text
StartDateTime	Date and time	yyyy-MM-dd HH:mm:ss
ComputerName	Domain\Computer	Text
TSclientIP	IP address	IP address format (numbers in segments with separators)
ClientName	Client name	Text
UserName	Domain\user name	Text
CanonicalOU	Canonical name	Text
EndDateTime	Date and time	yyyy-MM-dd HH:mm:ss
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
Memory	Value in Kb	Number
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CriticalCPUloadduration	Length of time	HHH:mm:ss
ExecutablePath	Executable path	Text

### 4.1.2 Reports for Usage Tracking Viewer tab: Application

Data shown on the **Application** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following methods:

Graph type/Report name	Method
Organizational Units for application	OUsforApplication
Users for application	UsersforApplication
Daily maximum users for application	DailyMaxUsersforApplication

#### Organizational Units for application

##### Filter

Column name	Expected input	Expected input format
Application ID	Application ID	Number

##### Results

Column name	Description	Format
Application	Application name	Text
Application ID	Application ID	Number
CanonicalOU	Canonical name	Text
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)

#### Users for application

##### Filter

Column name	Expected input	Expected input format
Application ID	Application ID	Number

##### Results

Column name	Description	Format
Application	Application name	Text
Application ID	Application ID	Number
UserName	Domain/username	Text

TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)

### Daily maximum users for application

#### Filter

Column name	Expected input	Expected input format
Application ID	Application ID	Number
StartDate	Date	yyyy-MM-dd
EndDate	Date	yyyy-MM-dd
SuppressZeroDates	True or False	Boolean
Sorts	ApplicationMetrics.SortFields	-

#### Results

Column name	Description	Format
Application	Application name	Text
DateTime	Date	yyyy-MM-dd
Maximum	Number	Number
MaximumReached	Date	yyyy-MM-dd HHH:mm:ss

### 4.1.3 Reports for Usage Tracking Viewer tab: OU

Data shown on the **OU** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following methods:

Graph type/Report name	Method
All applications for Organizational Unit	ApplicationsforOU
All Web sites for OU	WebSitesforOU

### All applications for Organizational Unit

#### Filter

Column name	Expected input	Expected input format
CanoninalOU	Canonical name	Text
OUGUID	GUID	[text]
OUinheritance	True or False	Boolean

No Screensaver	True or False	Boolean
StartYearWeek	Year/Week	yyyy-ww
EndYearWeek	Year/Week	yyyy-ww
GetDetails	True or False	Boolean
Sorts	OUMetrics.SortFields	-

## Results

Column name	Description	Format
Application	Application name	Text
Application ID	Application ID	Number
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
UserCount*	Number	Number
DeviceCount*	Number	Number
TimesStarted*	Number	Number
LastStarted*	Date and time	yyyy-MM-dd HH:mm:ss

Columns marked with an asterisk (\*) are only retrieved when **Get Details** is included in the filter.

## All Web sites for OU

### Filter

Column name	Expected input	Expected input format
CanonicalOU	Canonical	Text
OUGUID	GUID	[text]
OUinheritance	True or False	Boolean
Sorts	OUMetrics.SortFields	-

## Results

Column name	Description	Format
Website	URL	Text
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)

### 4.1.4 Reports for Usage Tracking Viewer: User

Data shown on the **User** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following methods:

Graph type/Report name	Method
All applications for user	ApplicationsforUser
All Web sites for user	WebSitesforUser

### All applications for User

#### Filter

Column name	Expected input	Expected input format
UserName	Domain/Username	Text
No Screensaver	True or False	Boolean
Sorts	UserMetrics.SortFields	-

#### Results

Column name	Description	Format
Application	Application name	Text
Application ID	Application ID	Number
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
UserName	Domain/Username	Text

### All Web sites for user

#### Filter

Column name	Expected input	Expected input format
UserName	Domain/Username	Text
Sorts	UserMetrics.SortFields	-

#### Results

Column name	Description	Format
Website	URL	Text
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
UserName	Domain/Username	Text

### 4.1.5 Reports for Usage Tracking Viewer tab: Web Site

Data shown on the **Web Site** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following methods:

Graph type/Report name	Method
All Organizational Units for Web site	OUsforWebSite
All users for Web site	UsersforWebSite

#### All Organizational Units for Web site

##### Filter

Column name	Expected input	Expected input format
Website	URL	Text
Sorts	WebsiteMetrics.SortFields	-

##### Results

Column name	Description	Format
CanonicalOU	Canonical name	Text
Website	URL	Text
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)

#### All users for Web site

##### Filter

Column name	Expected input	Expected input format
Website	URL	Text
Sorts	WebsiteMetrics.SortFields	-

##### Results

Column name	Description	Format
Website	URL	Text
UserName	Domain/Username	Text
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)



### 4.1.6 Reports for Usage Tracking Viewer: Computer

Data shown on the **Computer** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following methods:

Graph type/Report name	Method
All applications for computer	ApplicationsforComputer
All Web sites for computer	WebsitesforComputer
Computing usage per computer	ResourceUsageforComputer
Daily maximum users for computer	DailyMaxUsersforComputer

#### All applications for computer

##### Filter

Column name	Expected input	Expected input format
ComputerName	Domain\Computer	Text
No Screensaver	True or False	Boolean
StartDate	Date	yyyy-MM-dd
EndDate	date	yyyy-MM-dd
Get Details	True or False	Boolean
Sorts	ComputerMetrics.SortFields	-

##### Results

Column name	Description	Format
Application	Application name	Text
Application ID	Application ID	Number
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
UserCount*	Number	Number
DeviceCount*	Number	Number
TimesStarted*	Number	Number
LastStarted*	Date and time	yyyy-MM-dd HH:mm:ss

Columns marked with an asterisk (\*) are only retrieved when **Get Details** is included in the filter.

## All Web sites for computer

## Filter

Column name	Expected input	Expected input format
ComputerName	Domain\Computer	Text
Sorts	ComputerMetrics.SortFields	-

## Results

Column name	Description	Format
Website	URL	Text
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
ComputerName	Domain\Computer	Text

## Computing usage per computer

## Filter

Column name	Expected input	Expected input format
ComputerName	Domain\Computer	Text
Sorts	ComputerMetrics.SortFields	-

## Results

Column name	Description	Format
ComputerName	Domain\Computer	Text
TimeOpen	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
TimeActive	Length of time	HHH:mm:ss (HHH can exceed 24 hours)
CPUUsage	Length of time	HHH:mm:ss (HHH can exceed 24 hours)

## Daily maximum users for computer

## Filter

Column name	Expected input	Expected input format
ComputerName	Domain\Computer	Text
StartDate	Date	yyyy-mm-dd
EndDate	Date	yyyy-mm-dd
SuppressZeroDates	True or False	Boolean
Sorts	ComputerMetrics.SortFields	-

## Results

Column name	Description	Format
ComputerName	Domain\Computer	Text
Date	Date	yyyy-mm-dd
Maximum	Number	Number
MaximumReached	Date	yyyy-mm-dd HHH:mm:ss

### 4.1.7 Reports for Usage Tracking Viewer tab: Sessions

Data shown on the **Sessions** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following methods:

Graph type/Report name	Method
Session details	SessionDetails

#### Session details

#### Filter

Column name	Expected input	Expected input format
LimitResults	Max number of records	Number
Filtertype	and/or	Text
Filters (collection)	SessionDetails.FilterFields	-
Sorts	SeesionDetails.SortFields	-

## Results

Column name	Description	Format
UserName	Domain\username	Text
StartDateTime	Date and time	yyyy-MM-dd HH:mm:ss
ComputerName	Domain\Computer	Text
TSclientIP	IP address	IP address format (numbers in segments with dot as separator)
ClientName	Client name	Text
Session state	Online = 1 Offline = 0	Online Offline
CanonicalOU	Canonical name	Text
EndDateTime	Date and time	yyyy-MM-dd HH:mm:ss
Duration	Length of time	HHHH:mm:ss

### 4.1.8 Reports for Usage Tracking Viewer tab: Current Activity

Data shown on the **Current Activity** tab from the **Usage Tracking Viewer** can be accessed through the Web service using the following methods:

Graph type/Report name	Method
Current Activity details	CurrentActivityDetails
Current Activity per client name	CurrentActivityforClient
Current Activity per user	CurrentActivityforUser
Current Activity per computer	CurrentactivityforComp
Current Activity per OU	CurrentActivityforOU

#### Current Activity details

##### Filter

Column name	Expected input	Expected input format
CanonicalOU	Canonical name	Text
StartDateTime	Date	yyyy-mm-dd hh-mm-ss
Sorts (collection)	CurrentActivity.SortFields	-

##### Results

Column name	Description	Format
UserName	Domain\Username	Text
Application ID	Application ID	Text
StartDateTime	Date time	yyyy-mm-dd hh:mm:ss
ComputerName	Domain\Computer	Text
TSclientIP	IP address	IP address format (numbers in segments with separators)
ClientName	Client name	Text
OUGUID	Number	Number

#### Current Activity per client name

##### Filter

Column name	Expected input	Expected input format
UserName	Domain\Username	Text

Application or Web site	Application name URL	Text
Application ID	Application ID	Number
ComputerName	Domain\Computer	Text
ClientName	Client name	Text
OUGUID	Number	Number
OU Inheritance	True or False	Boolean

## Results

Column name	Description	Format
ClientName	Client name	Text
TSclientIP	IP address	IP address format (numbers in segments with separators)
Active applications	Number	Number

## Current Activity per user

### Filter

Column name	Expected input	Expected input format
UserName	Domain\Username	Text
Application or Web site	Application name URL	Text
Application ID	Application ID	Number
ComputerName	Domain\Computer	Text
ClientName	Client name	Text
CanonicalOU	Canonical name	Text
OU Inheritance	True or False	Boolean

## Results

Column name	Description	Format
UserName	Domain\Username	Text
Active applications	Number	Number

## Current Activity per computer

### Filter

Column name	Expected input	Expected input format
UserName	Domain\Username	Text

Application or Web site	Application name URL	Text
Application ID	Application ID	Number
ComputerName	Domain\Computer	Text
ClientName	Client name	Text
CanonicalOU	Canonical name	Text
OU Inheritance	True or False	Boolean
Sorts	ComputerMetrics.SortFields	-

## Results

Column name	Description	Format
ComputerName	Domain\Computer	Text
Active applications	Number	Number

## Current Activity per OU

### Filter

Column name	Expected input	Expected input format
UserName	Domain\Username	Text
Application or Web site	Application name URL	Text
Application ID	Application ID	Number
ComputerName	Domain\Computer	Text
ClientName	Client name	Text
CanonicalOU	Canonical name	Text
OU Inheritance	True or False	Boolean
Sorts	OUMetrics.SortFields	-

## Results

Column name	Description	Format
CanonicalOU	Canonical name	Text
Active applications	Number	Number

### 4.1.9 Search Methods

Five search methods can be used to retrieve the data from Usage Tracking. These are very similar to the search functionality in the **Usage Tracking Viewer**.

## Search OU

### Filter

Column name	Expected input	Expected input format
OUnName	OU name	Text
Sorts	OUMetrics.Sortfields	-

### Results

Column name	Description	Format
CanonicalOU	Canonical name	Text
GUID	GUID	Number

## Search Application

### Filter

Column name	Expected input	Expected input format
ApplicationName	Application name	Text
Sorts	ApplicationMetrics.SortFields	-

### Results

Column name	Description	Format
Application ID	Application ID	Number
ApplicationName	Application	Text

## Search User

### Filter

Column name	Expected input	Expected input format
UserName	User name	Text
Sorts	UserMetrics.SortFields	-

### Results

Column name	Description	Format
UserName	Account name	Text: Domain\Username

### Search Web site

#### Filter

Column name	Expected input	Expected input format
Website	URL	Text
Sorts	WebsiteMetrics.SortFields	-

#### Results

Column name	Description	Format
Website	Web site URL	Text

### Search Computer

#### Filter

Column name	Expected input	Expected input format
ComputerName	Computer Domain name	Text
Sorts	ComputerMetrics.SortFields	-

#### Results

Column name	Description	Format
ComputerName	Computer Account name	Text: Domain\Computer



## **Disclaimer**

Whilst every care has been taken by RES Software to ensure that the information contained in this publication is correct and complete, it is possible that this is not the case. RES Software provides the publication "as is", without any warranty for its soundness, suitability for a different purpose or otherwise. RES Software is not liable for any damage which has occurred or may occur as a result of or in any respect related to the use of this publication. RES Software may change or terminate this publication at any time without further notice and shall not be responsible for any consequence(s) arising there from. Subject to this disclaimer, RES Software is not responsible for any contributions by third parties to this publication.

## **Copyright Notice**

Copyright © on software and all Materials 1998-2013 Real Enterprise Solutions Development BV, P.O. Box 33, 5201 AA 's-Hertogenbosch, The Netherlands. RES and the RES Software Logo are either registered trademarks or service marks of Real Enterprise Solutions Nederland B.V. in Europe, the United States and other countries. RES Automation Manager, RES Workspace Manager, Dynamic Desktop Studio, Virtual Desktop Extender and RES VDX are trade names of Real Enterprise Solutions Nederland B.V. in Europe, the United States and other countries. All other product and company names mentioned may be trademarks and/or service marks of their respective owners. Real Enterprise Solutions Development BV, The Netherlands has the following patents: U.S. Pat. "US 7,433,962", "US 7,565,652", "US 7,725,527", other patents pending or granted.