

Monarch Server Report Mining Edition

Version 15.0



[Developer Guide]



Datawatch Corporation makes no representation or warranties with respect to the contents of this manual or the associated software and especially disclaims any implied warranties of merchantability or fitness for any particular purpose. Further, Datawatch Corporation reserves the right to revise this publication and make changes from time to time to its contents without obligation to notify anyone of such revisions or changes.

Monarch Server software is offered and is to be used in accordance with a SOFTWARE LICENSE AND MAINTENANCE AGREEMENT. This agreement stipulates that this software be used only in the computer system designated in that agreement. The agreement further stipulates that the customer shall not copy or alter, or permit others to copy or alter, the software or related materials in whole or in part, in any media for any purpose, except to make an archive (back-up) copy or to make a copy as an essential step in the use of the software with the customer's computer.

Datawatch Corporation hereby grants the buyer the right to reprint this documentation for internal uses only. No part of this publication may be reproduced, stored in a retrieval system, or transmitted in any form or by any means, electronic, mechanical, photocopying, recording, or otherwise, for any other purposes, without the prior written permission of Datawatch Corporation.

Monarch Server 15.0 Report Mining Edition - Developer Guide

Copyright © 2018 by Datawatch Corporation

All rights reserved. Printed in the U.S.A.

Unpublished - Rights reserved under the copyright law of the United States.

Monarch Server is a trademark of Datawatch Corporation. Other products mentioned herein may be trademarks or registered trademarks of their respective owners in the United States or other countries.

For U.S. Government End Users, the software is a "Commercial Item(s)," as that term is defined at 48 C.F.R. Section 2.101, consisting of "Commercial Computer Software" and "Commercial Computer Software Documentation," as such terms are used in 48 C.F.R. Section 12.212 or 48 C.F.R. Section 227.7202, as applicable. Consistent with 48 C.F.R. Section 12.212 or 48 C.F.R. Sections 227.7202-1 through 227.7202-4, as applicable, the Commercial Computer Software and Commercial Computer Software Documentation are being licensed to U.S. Government end users (a) only as Commercial Items and (b) with only those rights as are granted to all other end users pursuant to the Datawatch Software License and Maintenance Agreement.

DATAWATCH CORPORATION

CORPORATE HEADQUARTERS

4 Crosby Drive
Bedford, MA 01730, USA
Tel.: +1 800.445.3311
Fax: +1 978.441.1114

NEW YORK

415 Madison Avenue, Suite 1421
New York, NY 10017
Tel.: +1 800.445.3311

GREENSBORO, NC

101 S Elm #225
Greensboro, NC 27401
Tel: +1 800.445.3311

EUROPE, MIDDLE EAST, AND AFRICA

EUROPEAN HEADQUARTERS

Siena Court, Broadway
Maidenhead, Berkshire SL6 1NJ
Tel: +44 203.868.0230

DATAWATCH AB

Eriksbergsgatan 10
Stockholm, Sweden SE-114 30
Tel: +46 853.480.483

ASIA PACIFIC

MANILA

U2011 20th Flr Jollibee Plaza Condominium
F. Ortigas Jr. Rd., Ortigas Center
Pasig City 1605 PH
Tel: +63 2.633.5583

Table of Contents

Introduction	1
Monarch Server Report Mining Edition Interactive Mining	1
Monarch Server Report Mining Edition Export View Request	2
Entry Point for Initial Request.....	2
Request Parameters Reference.....	3
Quick Reference Sheet	3
Passing Report Paths.....	7
Checksum Generation	7
Code example — hashing a message.....	8
Code example — hash encryption	9

Introduction

Monarch Server Report Mining Server (RMS) Edition can be used through an initiation request. The request is sent using the HTTP POST method. It can be generated in a client browser by sending an HTML form with the fields named according to the corresponding parameters of the initiation request.

The initiation request contains instructions to determine the type of presentation to be provided. The type of presentation can be either a full interactive mining application for the user to select views, or it can be a session-less request to publish a single export of data based on a predefined set of mining rules to a particular data format.

Monarch Server Report Mining Edition Interactive Mining

RMS recognizes the request and then queries the report, model, and template data to determine the user's available views.

The initiation request contains the following information:

- ❑ **Initiation Request Type.** A request to provide a full interactive mining application.
- ❑ **Report Retrieval information.** This is a complete resolvable path to a source report to be used by RMS.
- ❑ **Model Retrieval information.** A complete resolvable path(s) to models to be used by RMS. A folder path is allowed. The folder shall be scanned for .xmod files to obtain a list of all available models.
- ❑ **Template Retrieval information** (Excel templates and ES Style style sheets). Provides complete resolvable path(s) to templates to be used by RMS.
- ❑ **Model to Template associations.** The templates of types Excel and ES Style shall be associated with a particular model for correct data extraction and processing.
- ❑ **The List of Allowed Views.**
- ❑ **User ID.** User identifier.
- ❑ **Document Type ID.** Server Library Folder identifier.
- ❑ **Default View.** The initial view to be presented to the user. The user can navigate to other available views.
- ❑ **Dynamic filter.** The initial dynamic filter for Data, Summary, and ES Style Views.



Monarch Server Report Mining Edition

Export View Request

RMS recognizes the request and then publishes the data in the format defined by the initiation request.

The initiation request contains the following information:

- ❑ **Initiation Request Type.** A request to provide a single export of data.
- ❑ **Report Retrieval Information.** This is a complete resolvable path to a source report to be used by RMS. Multiple report segments can be specified for processing. They will be concatenated together and then processed.
- ❑ **Model Retrieval Information.** A complete resolvable path to a model to be used by RMS.
- ❑ **Template Retrieval Information** (Excel templates and ES Style style sheets). Provides complete resolvable path to a template to be used by RMS.
- ❑ **Model to Template Associations.** The templates of types Excel and ES Style shall be associated with a particular model for correct data extraction and processing.
- ❑ **Data View.** The format of the single export view to be provided to the user.
- ❑ **User ID.** User identifier.
- ❑ **Dynamic filter.** The initial dynamic filter for Data, Summary, and ES Style Views.

Entry Point for Initial Request

The initiation request is sent to the RMS entry point. The URL of entry point is

```
http://<host name>:<portnumber>/  
    <RMS virtual directory>/RequestTypeAnalyze/AnalyzeRequest
```

where

- ❑ <host name> is the name of the computer where the target Monarch Server RMS WebServer component is installed
- ❑ <port number> is the port number on which the RMS WebServer is listening
- ❑ <RMS virtual directory> is the virtual directory under which the application is installed (RMSClient by default).

The request is sent using the HTTP POST method. All parameters are passed as the fields of the POST request.

The request's response will be HTML, PDF, CSV, XLS(X), PRF content, or an HTTP error.



Request Parameters Reference

Quick Reference Sheet

PARAMETER NAME	TYPE		DESCRIPTION
	M=Mandatory O=Optional		
	INTERACTIVE	EXPORT	
REQUEST_TYPE		O	Type of request. Valid values are "I" for interactive or "E" for a single export view of mined data. If the type is not specified, the default type is set to interactive.
REPORT_PATH	M	M	Complete resolvable path to the source report(s). Paths to multiple reports are separated with a semicolon. Can be a local path (accompanied by a host name passed via REPORT_HOST) or a UNC path.
REPORT_HOST	O	O	Name of the host where the report is located. If the report host is not set, then the report path must be a UNC path to be equally accessible to any exporter.
MODEL_PATHS	M	M	<p>Complete resolvable path to the source models (non-template-bound) or a model definition ID. RMS supports the following formats for this parameter:</p> <ul style="list-style-type: none"> ❑ path to a single model ❑ list of paths to the models. This list should be in the following format: <code><model_1_path>;<model_2_path>;... <model_n_path></code> ❑ path to the folder which contains models ❑ model definition ID <p>This parameter should list only table/summary models and not template-bound models. The template-bound models should be specified in the TEMPLATES_MODELS parameter.</p>
TEMPLATE_PATHS	O	O	Complete resolvable path to the source templates or a template definition ID. RMS supports the following formats for

			<p>this parameter:</p> <ul style="list-style-type: none"> ❑ path to a single template ❑ list of paths to the templates. This list should be in the following format: <code><template_1_path>;<template_2_path>;...<template_n_path></code> ❑ path to the folder which contains templates ❑ template definition ID <p>If this parameter is absent, XFORM data views should not be available.</p>
TEMPLATES_MODELS	O	O	<p>The list of model-to-template associations. Paths or object definition IDs can be used. Parameter format: <code><template_1_path>;<model_1_path>;<template_2_path>;<model_2_path>...<template_n_path>;<model_n_path></code></p> <p>If this parameter is absent, XFORM data views should not be available. The template-bound models should be specified in this parameter only and not in MODEL_PATHS.</p>
CHECKSUM	M	M	Contains digital signature for initiation request's parameters.
USER_ID	M	M	User identifier. This identifier used for getting user preferences and for authentication.
USER_PASSWORD	O	O	User password. If a password is not provided, a new dynamic user is created in the RMS DB. This user is assigned to the user group that is specified in the web.config file of the RMS Client. If a password is provided, the user with the USER_ID and PASSWORD should already exist and be assigned with the RMS user privilege to login to RMS Client.
DOCTYPE_ID	O	N/A	Server Library Folder identifier. If a Server Library Folder identifier is provided, all the models and templates contained in the folder are made available to the user (if the user has the necessary access permissions). This identifier is also used when the user saves a model to the server. If DOCTYPE_ID is not provided, the model cannot be saved to the server. If DOCTYPE_ID is provided, then the model can be stored in RMS and can be



DYNAMIC_FILTER	O	O	<p>re-used in the next user sessions.</p> <p>The initial dynamic filter for Data, Summary, and ES Style Views. It consists of a value pair of column name and filter value:</p> <p><i>FILTERCOLUMN=[ColumnName];FILTERVALUE=[FilterValue];FILTEROPERATOR=[EQ.NEQ,GT,LT,GTE,LTE,CONTAINS,STARTSWITH,ENDSWITH]</i></p> <p>The date format for the filter value is YYYYMMDD.</p>
ALLOWED_VIEWS	O	N/A	<p>The list of data views that should be available for the user. Parameter format: <code><data_view_1 >;<data_view_2>;... <data_view_n></code>, where <code><data_view_n></code> is one of the following constants:</p> <ul style="list-style-type: none"> <input type="checkbox"/> HTML_REPORT <input type="checkbox"/> DYNAMIC <input type="checkbox"/> TABLE SUMMARY <input type="checkbox"/> XLS_TABLE XLS_SUMMARY PRF XFORM <input type="checkbox"/> REMOTE_PORTLETS
TIMESTAMP	M	M	<p>Timestamp of the request. This is the number of milliseconds that has elapsed since Jan 01, 1970 (Java-style date/time representation)</p>
JOINPASSWORD	O	O	<p>Join password for logging into an external data source.</p>
DEFAULT_VIEW	O	M	<p>Name of the view to show initially. Default is the Welcome page. The view name can be one of the values permitted for the ALLOWED_VIEWS parameter or the "WELCOME" value for the Welcome page. The parameter is mandatory for export type request to identify the requested view.</p>
EXPORTTO	N/A	M	<p>Defines the format of the exported view.</p> <p>Can be one of the following:</p> <p>For HTML_REPORT:</p> <ul style="list-style-type: none"> <input type="checkbox"/> PDF <input type="checkbox"/> HTML <p>For TABLE:</p> <ul style="list-style-type: none"> <input type="checkbox"/> PDF

			<input type="checkbox"/> HTML <input type="checkbox"/> CSV For SUMMARY: <input type="checkbox"/> PDF <input type="checkbox"/> HTML For XLS_TABLE: <input type="checkbox"/> XLS <input type="checkbox"/> CSV For XLS_SUMMARY: <input type="checkbox"/> XLS For PRF: <input type="checkbox"/> PRF For XFORM: <input type="checkbox"/> PDF <input type="checkbox"/> HTML
DELETE_REPORTS	0	0	Report removal flag. Set to "true," "1", or "on" to remove source report after user session ends.
DELETE_MODELS	0	0	Model removal flag. Set to "true," "1", or "on" to remove source models after user session ends.
DELETE_TEMPLATES	0	0	Template removal flag. Set to "true," "1", or "on" to remove source templates after user session ends.
ADMIN	0	N/A	Admin mode flag. Set to "true," "1", or "on" to allow uploading shared models and deleting models.
REPORT_ENCODING	0	0	<p>Encoding of incoming reports. Can be one of the following:</p> <input type="checkbox"/> ANSI <input type="checkbox"/> ASCII <input type="checkbox"/> UTF8 <input type="checkbox"/> UTF16LE <input type="checkbox"/> UTF16BE
LOCALE	0	0	<p>The locale to be used in the current session. Can be one of the following:</p> <input type="checkbox"/> en (English) <input type="checkbox"/> fr-FR (French-France)



- ❑ **de-DE** (German-Germany)
If the value is not defined or invalid, **en** (English) will be used.

Passing Report Paths

The REPORT_PATH should be a local path at the report storage machine, the host name of which is REPORT_HOST. The parameter may include a path to one report or to multiple reports, in which case the paths must be separated by semicolons. The RMS will access the report(s) directly in the following occasions:

- ❑ The Monarch Server Report Mining Edition that is performing the data processing is located at the machine described by the REPORT_HOST parameter.
- ❑ REPORT_PATH is a UNC path.

In all other cases, the RMS searches for the RmsContentServiceHost component to perform report data transfer.

Checksum Generation

The initial request is protected by a checksum — a type of digital signature.

The generation of a checksum includes computing a secure hash for significant attributes of the current request and then signing this hash with a secret key using an XOR operation.

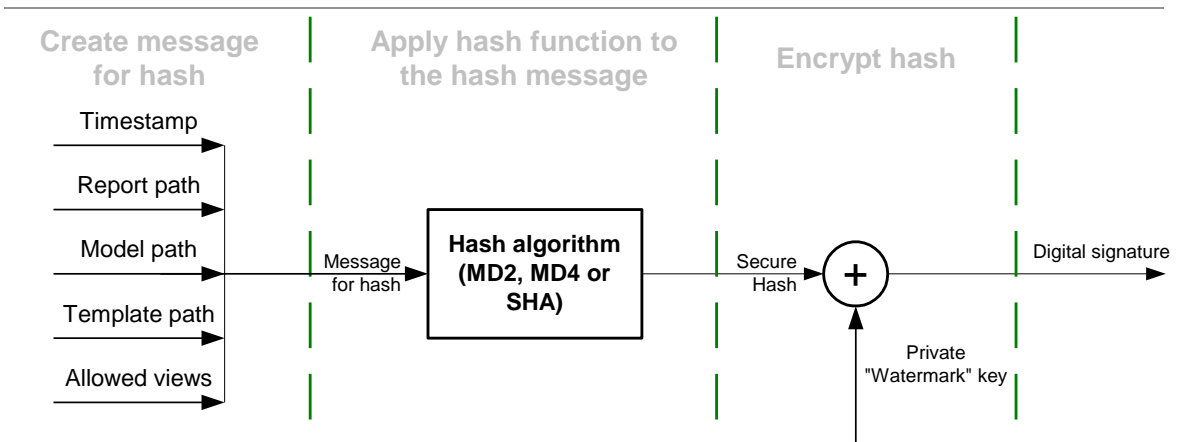


Figure 1. Computing a Checksum

The digital signature should be hex-encoded into symbolic representation before passing into RMS.

CODE EXAMPLE — HASHING A MESSAGE

To create a hash message, the request's parameters should be concatenated into one string in the following order: timestamp, report path, model path, template path, allowed views. The code below shows a simple C# function for generating a hash message:

```
// Create message for hash function
public static String GetHashMessage(
    long timestamp,
    String host,
    String reportPath,
    String modelPath,
    String templatePath,
    String allowedView)
{
    StringBuilder sb = new StringBuilder();
    // Concatenate all parameters into one String
    sb.Append(timestamp);
    sb.Append(host);
    sb.Append(reportPath);
    sb.Append(modelPath);
    sb.Append(templatePath);
    sb.Append(allowedView);
    // Return result
    return sb.ToString();
}
```

RMS supports all hash algorithms from the C# library. The algorithms are as follows.

The active algorithm can be set via RMS's configuration file.

Below is a simple C# function for hashing a message:

```
// Hashing
private static byte[] GetHash(String message, String
hashAlgorithmName)
{
    byte[] buf = Encoding.UTF8.GetBytes(message);

    // Create hash algorithm instance
    HashAlgorithm algorithm =
    HashAlgorithm.Create(hashAlgorithmName);
    // Hash message
    return algorithm.ComputeHash(buf);
}
```



CODE EXAMPLE — HASH ENCRYPTION

The hash shall be encrypted by an XOR operation with a secret key. The secret key's length is large enough to provide strong encryption. The key can be set differently for every instance of RMS to increase security. It is stored in the RMS configuration file and can be changed by the administrator.

Below is a simple C# function for hash encryption:

```
public static String Encrypt(String message, String
watermarkString, String hashAlgorithmName)
{
    byte[] hash = GetHash(message,
hashAlgorithmName);

    byte[] watermarkKey =
StreamHelper.HexStringToByteArray(watermarkString);
    // Check key Length
    if (hash.Length > watermarkKey.Length)
        throw new
RmsWebServerException("Incorrect \"watermark\" key.");
    // XOR
    byte[] encryptedHash = new byte[hash.Length];
    for (Int32 i = 0; i < hash.Length; i++)
    {
        encryptedHash[i] = (byte)(hash[i] ^
watermarkKey[i]);
    }
    // Return result
    return
StreamHelper.ByteArrayToHexString(encryptedHash);
} }
```