Metadata Configuration File
For RRD MA

Authors	Szymon Trocha
Date	29-11-2006
Current Version	1.4

Document Change Log

As SA3-WI15 Document			
Version number	**Date**	**Description of change**	**People**
1	20-11-06	First draft issued	Szymon Trocha
1.1	22-11-06	Numbering of levels according to Roman's comments	Szymon Trocha
1.2	23-11-06	Moved examples to the beginning and added new sections – Introduction and overview. Also added comments from reviewers and some additions based on reviews	Loukik Kudarimoti
1.3	27-11-06	Reviewers comments addressed	Szymon Trocha
1.4	29-11-06	Clarifying structure, final comments addressed	Loukik Kudarimoti, Szymon Trocha

General information

Service Name:
Service Type: MA
Version/release: 1.0
Service Description: Measurement Archive service using Round Robin Database
Contact person (s): Roman Łapacz
Contact Information: romradz@man.poznan.pl

Introduction

The Metadata configuration file describes measurement data stored in RRD files of the Measurement Archive. RRD Measurement Archive uses this configuration file while retrieving data as well as while registering to Lookup Service.

The configuration file can be made up of any number of 'chains'. Each chain describes the measurement data measured for a single interface, for one direction only. It also provides the location (the name of the RRD file, data source within the RRD file, etc) where the data can be fetched.

Each chain is made up of one metadata block and one data block linked together. The Metadata block describes the interface for which measurement data is stored. Each block has unique metadata and subject identifiers. For configuration simplicity, while working with your own file start with the metadata identifier (`id` attribute in the metadata element) with a value something like `meta1` and increment its value each time you add next metadata/data block for next interfaces monitored like `meta2`, `meta3`, etc.

The data block contains a key which is a pointer to the location of the RRD file along with the data source and the unit format in which data is stored. Units are used by clients to properly interpret retrieved values. The data block references to the metadata block using `metadataIdRef` attribute value. Make sure you CORRECTLy reference the proper metadata block in the data block i.e. `metadataIdRef` attribute value must be equal to `id` attribute value of the metadata block describing collected data.

The default installation of perfSONAR release provides a test metadata configuration file which together with test RRD file can be used for testing purposes. The content of this test metadata configuration file is just for demonstration and one must change it to correctly describe the content of one's own RRD archives.

The following section in this document contains an overview of the entire configuration file. Examples are available in the Example section. Each example is explained later on under the Schema elements section.

Overview of the configuration file

In order to get you familiarized with the structure of the configuration file, an example is provided in the next paragraph. This example lists all elements which are numbered indicating their nesting level inside the blocks. An explanation of such a nesting can be found below.

The metadata configuration file starts with the `store` element [1] which contains all namespaces used by the metadata configuration file. Then there is a chain of one or more pairs of `metadata` [1.1] and `data` [1.2] blocks. `Metadata` and `data` sections can be repeated as necessary incrementing the `id` attribute.

`Metadata` element [1.1], `subject` [1.1.1] as well as `interface` element [1.1.1.1] are container elements (building blocks of the structure of the configuration file). All elements placed under the interface block [1.1.1.1.1-1.1.1.1.7] describe interface for which measurement data is stored.

`Parameter` element with attribute `name` provides information about stored metric.

The `data` block pairs with the `metadata` block and provides information about the location of the data on the system. It uses `metadataIdRef` attribute to reference to the proper `metadata` block. Inside `data` block there are two elements: `key` [1.2.1] and `parameters` [1.2.1.1] which are container elements of the `data` block. Finally `parameter` elements [1.2.1.1.1-1.2.1.1.3] contain location of RRD file as well as data source name and measurement units.

Examples

This is an example of metadata configuration file for RRD Measurement Archive:

```
1 <nmwg:store
        xmlns:nmwgt="http://ggf.org/ns/nmwg/topology/2.0/"
        xmlns:nmtm="http://ggf.org/ns/nmwg/time/2.0/"
        xmlns:nmwg="http://ggf.org/ns/nmwg/base/2.0/"
        xmlns:perfsonar="http://ggf.org/ns/nmwg/tools/org/perfsonar/1.0/"
        xmlns:netutil="http://ggf.org/ns/nmwg/characteristic/utilization/2.0/"
        xmlns="http://ggf.org/ns/nmwg/base/2.0/">

1.1     <nmwg:metadata id="meta1">
1.1.1     <netutil:subject id="subj1">
1.1.1.1     <nmwgt:interface>
1.1.1.1.1     <nmwgt:hostName>mordor.middleearth.pl</nmwgt:hostName>
1.1.1.1.2     <nmwgt:ifAddress type="ipv4">10.1.2.3</nmwgt:ifAddress>
1.1.1.1.3     <nmwgt:ifName>test-0/1/0</nmwgt:ifName>
1.1.1.1.4     <nmwgt:ifDescription>Interface towards
                 Mordor</nmwgt:ifDescription>
1.1.1.1.5     <nmwgt:direction>in</nmwgt:direction>
1.1.1.1.6     <nmwgt:authRealm>rohan</nmwgt:authRealm>
1.1.1.1.7     <nmwgt:capacity>1000BaseT</nmwgt:capacity>
1.1.1.1     </nmwgt:interface>
1.1.1     </netutil:subject>
1.1.2     <nmwg:parameters>
1.1.2.1     <nmwg:parameter
                name="supportedEventType">utilization</nmwg:parameter>
1.1.2     </nmwg:parameters>
1.1     </nmwg:metadata>

1.2     <nmwg:data id="data1" metadataIdRef="meta1">
1.2.1     <nmwg:key>
1.2.1.1     <nmwg:parameters>
1.2.1.1.1     <nmwg:parameter
                name="file">/ps/data/rrd/test/test.rrd</nmwg:parameter>
1.2.1.1.2     <nmwg:parameter name="dataSource">bytes</nmwg:parameter>
1.2.1.1.3     <nmwg:parameter name="valueUnits">Bps</nmwg:parameter>
1.2.1.1     </nmwg:parameters>
1.2.1     </nmwg:key>
1.2     </nmwg:data>

1  </nmwg:store>
```

Namespaces

```
xmlns:nmwgt="http://ggf.org/ns/nmwg/topology/2.0/"
xmlns:nmtm="http://ggf.org/ns/nmwg/time/2.0/"
xmlns:nmwg="http://ggf.org/ns/nmwg/base/2.0/"
xmlns:perfsonar="http://ggf.org/ns/nmwg/tools/org/perfsonar/1.0/"
xmlns:netutil="http://ggf.org/ns/nmwg/characteristic/utilization/2.0/"
xmlns="http://ggf.org/ns/nmwg/base/2.0/"
```

Schema elements explained

1 Element-Name: `nmwg:store`

Mandatory: YES
Description: This element contains all namespaces used by the metadata configuration file.

1.1 Element-Name: `nmwg:metadata`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

Attribute: `id`

***Mandatory*:** YES
Values: Textual value
***Description*:** Identifier of metadata block. Each metadata element within the file must have unique identifier.

1.1.1 Element-Name: `netutil:subject`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

Attribute: `id`

***Mandatory*:** YES
Values: Textual value
***Description*:** Identifier of subject element. Each subject element within the file must have unique identifier.

1.1.1.1 Element-Name: `nmwgt:interface`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information. Interface block provides details for interface for which measurement data is stored.

1.1.1.1.1 Element-Name: `nmwgt:hostName`

Mandatory: NO
Description: Network element (router, switch) hostname. Should be a loopback interface host name.

1.1.1.1.2 Element-Name: `nmwgt:ifAddress`

Mandatory: NO
Description: Interface IP address.

1.1.1.1.3 Element-Name: `nmwgt:type`

Mandatory: NO
Description: Describes type of IP protocol used.
Possible values: ipv4, ipv6

1.1.1.1.4 Element-Name: `nmwgt:ifName`

Mandatory: NO
Description: Interface name for which measurement data is collected.
Constraints (formatting, etc) on values: The name must be given as specified in the network element.

1.1.1.1.5 Element-Name: `nmwgt:ifDescription`

Mandatory: NO
Description: Textual description of interface for which measurement data is collected.

1.1.1.1.6 Element-Name: `nmwgt:direction`

Mandatory: NO
Description: Indicates traffic direction for this measurement data.
Possible values: in, out

1.1.1.1.7 Element-Name: `nmwgt:authRealm`

Mandatory: NO
Description: Authentication realm to indicate who can access the data. Not in use currently. You may enter your Domain/NREN name here.

1.1.1.1.8 Element-Name: `nmwgt:capacity`

Mandatory: NO
Description: Textual description of link capacity. This parameter may be used by clients to estimate link load.

1.1.2 Element-Name: `nmwg:parameters`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

1.1.2.1 Element-Name: `nmwg:parameter`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.
Possible values: utilization

Attribute: `name`

***Mandatory*:** YES
Values: "`supportedEventType`"
***Description*:** Indicates metric supported by this archive.

1.2 Element-Name: `nmwg:data`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

Attribute: `id`

***Mandatory*:** YES
Values: Textual value
***Description*:** Identifier of data element. Each data element within the file must have unique identifier.

Attribute: `metadataIdRef`

***Mandatory*:** YES
Values: Textual value
***Description*:** Identifier of corresponding metadata element. Each data element within the file must have identifier which is a reference to the proper metadata block i.e. is equal to id attribute this block.

1.2.1 Element-Name: `nmwg:key`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information. Key is used by the service to identify the location of data. Then the client can use the key in order to request data.

1.2.1.1 Element-Name: `nmwg:parameters`

Mandatory: YES
Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

1.2.1.1.1 Element-Name: `nmwg:parameter`

Mandatory: YES

Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

Attribute: `name`

***Mandatory*:** YES

Values: "`file`"

***Description*:** Path to RRD file storing characteristic described in metadata section.

1.2.1.1.2 Element-Name: `nmwg:parameter`

Mandatory: YES

Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

Attribute: `name`

***Mandatory*:** YES

Values: "`dataSource`"

Description: Name of data source (DS) parameter as specified in RRD file and corresponding to the above traffic direction. One can extract it executing "rrdtool info filename.rrd" command on one's RRD measurement server.

Constraints (formatting, etc) on values: DS name must exactly correspond to RRD file configuration.

1.2.1.1.3 Element-Name: `nmwg:parameter`

Mandatory: YES

Description: This element is a container element. It acts as one of the building blocks of the metadata configuration file. It contains other elements which may provide actual information.

Attribute: `name`

***Mandatory*:** YES

Values: "`valueUnits`"

Description: Units of data stored in RRD file.

Other Information