Resource Tracker

A Purposed Remote System Monitoring Methodology

SandeepKaur
School of Computing Science&Engg
GalgotiasUniversity
GreaterNoida,Indialine 3-City, Country
sandeep.kaur@galgotiasuniversity.edu.in

MamtaNarwaria
School of Computing Science&Engg
GalgotiasUniversity
GreaterNoida,Indialine 3-City, Country
mamta.narwaria@galgotiasuniversity.edu.in

Abstract—**Hardware management and resource tracking is an important activity in maintaining labs at institute and offices. The resources are spread across various labs; some of the systems are at different location of the office. The monitoring of these systems is itself a tedious task. In this paper a 'Resource Tracker' methodology is proposed that provides a method to track remote resources by identifying parameters. With this approach we can create custom report organized in the way we want to show details about the system parameters like basic hardware details, memory, disk drivers, network status of the machine etc. The aim is improve the data accuracy and overcome limitation of traditional methods to track the information of resources. The motivation for development of "Resource Tracker "came from an inherent drawback of the traditional methods. The methodology also provides holistic view of the data by combining the parameters and hence gives a collective view and resource analysis. The methodology also provides a way to find out a system with particular hardware and software configurations. The proposed system is based on three-tierarchitecture.**

Keywords—WMI(Window management Instrumentation); Remote System;Resources.

I. INTRODUCTION

The methodology aims to provide a central repository with information about physical systems. So in basic terms this application is repository of information. This can be viewed as a result of the natural evolution of information technology. With the increasing size of people in the team and increasing sizes and number of hardware resources, it becomes very important task to be able to manage all these resources. The methodology aims to provide with all the important information readily available. Hardware management and resource tracking is an important activity in maintaining labs at institute and offices. There are various teams in organization like "Engineering team" or a "Sale and Marketing team" and various labs in institute where a number of computer systems are in regular use. These are frequently modified or updated both from hardware and software perspective. It always helps to know how many resources (RAM, Hard disk etc.) are allocated to the various systems, whether a systems is responding or not. Having information readily available at hand saves tremendous amount of time. Resource Tracker methodology provides a definite improvement in management of the systems. The instruments of study are based on the relevant literature and the data generatedby WMI. Windows Management Instrumentation (WMI) is the infrastructure for management data and operations on Windows-based operating systems. It can be used in scripts or applications to automate administrative tasks on remote computers. It is a set of extensions to the Windows Driver Model that provides an operating system interface through which instrumented components provide information and notification. WMI is Micros Management (WBEM) and Common Information Model (CIM) standards from the Distributed Management Task Force (DMTF). There are close to 100 WMI providers made available by Microsoft. But these information are very lengthy and hence difficult to analyze on a whole, mainly due to the nature of the data. Traditionally such information may be maintained in excel sheets or mails. But the problem in traditional method is that it's not easy to track the information and since the information is manually updated and maintained, chances for errors are higher. With increases number of systems it become impractical to track system through traditional approach. There are some commercial tools available in market but these may be complex to setup (like Nagios) or involves significant cost. The motivation for development of "Resource Tracker "came from an inherent drawback of the traditional reports. Resource Tracker methodology provides reports to track remote resources by identifying parameters. With this approach we can create custom reports organized in the way we want to show details about the system parameters like basic hardware details, Memory, disk drivers, Network status of the machine etc. This way the methodology also provides holistic view of the data, by combining the parameters and hence aim to provide a collective view and resource analysis. The methodology also provides a way to find out a system with particular hardware and software configurations. This methodwouldprovidefollowingadvantages:

1. Better management of the physicalsystems.
2. By providing notification to users whenever a machine goes offline or is not reachable, user will get alerted to take necessary steps and prevent the scheduledtasks**.**
3. Easier for the administrator to monitor the usage of thesystems.

ISBN:978-1-5090-1666-2/16/$31.00 ©2016 IEEE

II. RESEARCHMETHODOLOGY

In this paper based on the requirement we consider of these following parameters list track:-

1. User should be able to add and update the machine to track in thesystem.
2. User should get details about the following parametersofthesystem
   a. Basichardwaredetailslikethemachine manufacturer,model etc.
   b. BasicOSdetailslikeOSnameetc.
   c. Disk drives.
   d. Memory.
   e. Processesrunningonthesystems.
   f. The list of products installed for the user(particulartowindows)
   g. List of the drives shared by user (particularto windows)
   h. Network Status of the machine i.e. whether its connected to network or not
3. Usercanalsogroupthemachinestogetherunder a specificname.
4. Basic performance requirements also hold for the application specifically to loadingtime.

a. Analysis for the major requirements

Analyzing the requirements gives the initial design for the system. Following are the major point for the analysis of the system

1. The parameters for the system can be separated out into three different categories:-
   a. **Category-1**: These are the very rare or non-changing parameters for the systems like the manufacturer, model number of the system. The Network adaptorsonthesystem,OSinstalled.

   b. **Category-2**: There are parameters that change more frequently than category-1 parameters but very rarely as compared to the category-3 parameters. Someof the parameters in this category can be disk drives on the system, RAM on thesystemsetc.

   c. **Category-3**: These are the most frequently changing parameters for the system like the list of the processes running on the system, currentfreememoryorspace.

Based on the three categories the data displayed can be grouped as:-

Sr.	Property	Category
1.	Manufacturer and Model, processor	1
2.	OS installed	1
3.	Physical Drives	2
4.	Partitions/Volumes	2
5.	Shares on the system (win	2
6.	Physical Memory	2
7.	Processes running	3
8.	Product installed	2
9.	Current memory status	3

Table1: Category of the parameters

2. The category of the property will also define how the data for the property will be stored at the application end i.ewhether it will be stored as form of xml, whether fetched at runtimeorwhetherindatabase.
3. Since the application needs to interact with different type of machines like windowssystems, MAC based system or a UNIX based systems, so it is necessary to have a standard format of the data exchange. XML is generic format which can be used to store and transfer theinformation.
4. Since user needs to have functionality to group machines, this also implies that one machine can be part of multiple groups and the default group for each user will be his username.
5. The status of the machine (i.e. whether they are on network or not) is the most basic requirement for the system, so this data should be readily available or be the first screen for theuser on login.

III. KEY CONCEPTS ANDTECHNOLOGIES

WMI – The silver bullet for windows systems: Windows Management Instrumentation (WMI) is the infrastructure for management data and operations on Windows-based operating systems. It can be used in scripts or applications to automate administrative tasks on remote computers. It is a set of extensions to the Windows Driver Model that provides an operating system interface through which instrumented components provide information and notification. WMI is Microsoft's implementation of the Web-Based Enterprise Management (WBEM) and Common Information Model (CIM) standards from the Distributed Management Task Force (DMTF). There are close to 100 WMI providers made available by Microsoft. There are lots of classes available in WMI to query the system data. Below is the list of the major classes used for theproject:-

- Win32_ComputerSystem:- This class represents the computer system running windows. The Computer systemparameters fetched using this class like Manufacturer, Model, System Type and total Physical Memory (in Bytes).These parameters comes under Category 1.
- Win32_OperatingSystem:-The Win32_ Operating System WMI class represents a Windows-based operating system installed on a computer. Any operating system that can be installed on a computer that can run a Windows-based operating system is a descendent or member of this class. The properties fetched using this class are: Name, CSD Version, Current Time Zone, Last Boot upTime and these are all category 1 parameters. The other properties fetched using this class are Total Visible Memory Size, Total Virtual Memory Size, Free Physical Memory and Free Virtual Memory. These are all category 3 parameters.
- Win32_Processor:-The Win32_Processor WMI class represents a device that can interpret a sequence of instructions on a computer running on a Windows operating system. On a multiprocessor computer, one instance of the Win32_Processor class exists for each processor.
- Win32_DiskDrive:- The Win32_DiskDrive WMI class represents a physical disk drive as seen by a computer running the Windows operating system. Any interface to a Windows physical disk drive is a descendent (or member) of this class. The Disk Driver parameters properties fetched using this class arecaption,Name,Size,partitions.
- Win32_LogicalDisk:- The Win32_LogicalDisk WMI class represents a data source that resolves to an actual local storage device on a computer system running windows. The Local disk parameters fetched using this class are Caption, Description, Free Space andSize
- Win32_PhysicalMemory:-The Win32_ Physical Memory WMI class represents a physical memory device located on a computer system and available to the operating system. Using this class we can get the details of RAM modules installed on the system. The Physical memory parameters fetched using this class are Capacity, Description andDevice Locator.
- Win32_Process:- The Win32_Process WMI class represents a process on an operating system. So using this class we fetch the details of the processes running on the system. The process parameters fetched using this class are Process ID, Name, ExecutablePath Thread Count, Handle (Process Identifier), Virtual Size and Peak Virtual Size. These are all Category 3 parameters and need to be fetchedatruntime.
- Win32_Product:- The Win32_Product W M I class represents products as they are installed by Windows Installer. A product generally correlates to one installation package. The product parameters fetched using this class are Name, Version, Vendor, Install Date and Install Location. These are also Category 2 parameters.

There are vast set of classes provided by Microsoft which can be used to increase the dataset to be fetched. Currently the classes list is keeping in mind the key parametersrequired.

IV. RESOURCETRACKERSYSTEM

Resource tracker methodology will be based on Three Tier Architecture. This is a client-server architecture in which the user interface, functional process logic, computer data storage and data access are developed and maintained as independent modules, most often on separate platforms. The three-tier model is considered to be a software architecture and a software design pattern. Apart from the usual advantages of modular software with well-defined interfaces, the three-tier architecture allows any of the three tiers to be upgraded or replaced independently as requirements or technology change. For example, a change of operating system in the presentation tier would only affect the user interface code. Typically, the user interface runs on a desktop PC or workstation and uses a standard graphical user interface, functional process logic may consist of one or more separate modules running on a workstation or application server, and an RDBMS on a database server.

Figure 1 represent the three tier architecture for the methodology. The various tiers are represented in this along with the interaction between the various tiers.

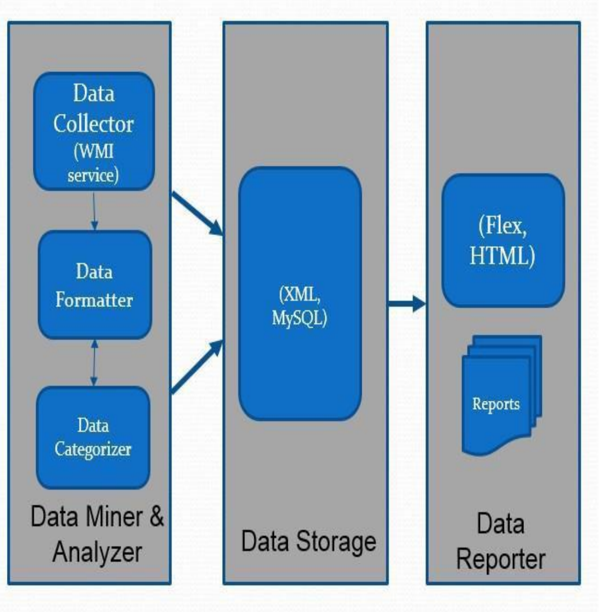

Figure 1: Resource Tracker System

The Resource Tracker System architecture has the following three tiers:

a.) Data Reporter

The presentation tier for Resource Tracker is 'Data Reporter'. This is the topmost level of the methodology. The presentation tier displays information related to such services as operating system details, Physical memory etc. The data reporter main functionality is to gather the data from the storage and display it depending on the system or the group of the systems selected by the user. It communicates with other tiers by outputting results to the browser/client tier. This can be developed using tools like Flex, HTML etc.

b.) Data Miner & Analyzer

Application Tier (Business Logic/Logic Tier) is called Data Miner & Analyzer. The logic tier is pulled out from the presentation tier and, as its own layer, it controls an application's functionality by performing detailed processing. The logic tier will consist of set of scripts and set of web services. The request from the user will be received by the web services and processed by appropriate scripts. The scripts, based on the category of the data requested fetch the data and return it in the suitable format to the requesting. .This tier is divided into 3 phases

1. Data Collector: - In this phase data is collected using WMI. The ability to obtain management data from remote computers is what makes WMI useful. Following are the basic steps to be followedtoobtainsystemdatausingWMI:-

   i.) Decide the programming language to use for the WMIscripting.

   ii.) Ensure that your connections to remote computers work.

   iii.) After connecting to WMI, you can obtain data through queries andenumerations.

2. Data Categorizer: - Once the data is fetched from the remote machine, the data is categorized according to category as defined previously i.e. whether the details fetched for selected system belongsto:-

   i.) Category-1 parameters – This includes hardware manufacturer, model and basic OS details like OS, Patch, last boot time are displayed.

   ii.) Category-2 parameters - This include the details of physical drives on the system, the logical drives and shares created on the system and memory details like the RAM modules installed on the system and the physical and virtual memory status.

   iii.) Category-3 parameters– This includes list of the processes running on the system. E.g. Process ID, name etc. Those details that are fetched at run-time from the system.

3. Data Formatter: - In this phase data is formatted tomake it more readable as the data returned from the WMI API will be in raw format. For example the disk data (size, free size) are converted to Giga Bytes and stored in the database.

c.) Data Storage

This tier consists of Database Servers. Here information is stored and retrieved. This tier keeps data neutral and independent from application servers or business logic. Giving data its own tier also improves scalability and performance.

In resource tracker system the data can be stored in two type of formats:-

1. SQL database – The SQL database will store some of the basic data like the user details, machine details, and some Category-1 parameters data. The data can be fetched from SQL database in XML format, or in other words can be converted to XML format to display inReporter.

2. XML files – Run time data particularly the Category 3 and few of Category 2 type parameters are fetched at the runtime i.e. when user requests/selects a particular type of report like current running processes on the system. Since this data is regularly changing so it's not stored inSQL database and will be stored as suitably formatted XMLswheneveruserrequestsforit.

The using of two different types of storage mechanism does not add overhead because the logic tier suitable formats the data, in both the cases, to XML format. The advantage is that it provides flexibility of data storage. The data stored in the XML can be shifted to the database or the other way round without affecting the user interface for the application. In this case the scripts just read the data fromtheXMLfilesandsenditbacktothefrontend.

d.) Algorithm & Sample Script

Below is sample structure for the script implemented in Perl language:-

```
<header section> use Win32::OLE;

use DBI::Mysqlsimple; use XML::Simple;

<variables section>

/*Here we define the parameter and the properties to use and also the array structure to hold the values*/

$WMI_Key= "Capacity,Caption,Description,DeviceLocator";

$Win32_Class = "PhysicalMemory"; my @dumpxml = ();

/*Note that the array elements will be hash references. These hash references will contain set of parameter and values. Data in the array is populated as*/
```

```
$dumpxml[<count of the set>]{$object->{Name}} =

$object->{Value};

/*The XML conversion is done using the XML::Simple module as*/

$xml = new XML::Simple (NoAttr=>1, RootName=>'data');

$xml->XMLout(\@dumpxml);
```

There can be a separate script for each of the different set of parameters. But the structure is same and these can becombined easily into a single script or exe. Internally the script can use an array of hashes to store the entire set of parameters and values. Finally the array of hashes is converted to XML.

Algorithm for the Script:-

1. Set all the local variable i.e. the property, parameters,arraysetc.
2. Get the details for the machine to fetch thedetails.
3. Check the type of the property i.e. Category-1, Category-2orCategory-3
4. If Category-1 gotostep7.
5. If Category-2 gotostep8.
6. If Category-3 gotostep9.
7. Getthedetails for the parameters from the database and gotostep12.
8. Get the path to the XML file storing the details forthesystemandgotostep12
9. Connect to theremotesystem.
10. Gettheparametersandvaluesfromthesystem.
11. Loop through the set of parameters until all the parameters have been processed i.e. added to the hash and the hash appended to array. Gotostep 12.
12. CovertthedatatoXMLformat.
13. Return/PrinttheXML.

The XML is then received by the User Interface.

V. CONCLUSIONS

In various big software companies, lot of efttforts are spend by the engineers in tracking their systems details because of the fact that the information is not readily available.Also the resource administrator needs to keep the track of all the resources (like RAM, disk drives) being used in his team so as to effectively manage them. Again in this case need arises to have the correct information readily available. Most of the times we overlook this activity, undermining its importance. This paper is an effort to improve productivity and make the useful information readily available, which helps to save a lot of time and effort. Moving forward this methodology can be enhanced further to improve and expand its functionality. We can use various techniques to send instant notification whenever a machine goes of the network. User can also set threshold levels like for disk free, number of processes, RAM usage and whenever any threshold is exceeded user will be send an alert like SMS or mail. To enable analysis and reporting feature can be another extension to the system. The system can be enhanced to maintain historical datamaybeasxmlfiles.

REFERENCES

[1] Wenxian Zeng and Yue Wang, “Design and Implementation of Server Monitoring System Based on SNMP”, IEEE International Joint Conference on Artificial Intelligence, pp. 680-682, 2009.

[2] http://www.perl.org/

[3] http://www.nagios.org/

[4] A. Bohra, I.Neantiu, P.Gallard,F.Sultan, and L. Iftode, “Remote repair of operating system state using Backdoor”, IEEE proceeding of the international conference on Autonomic computing, May 2004.

[5] http://www.w3schools.com/xml/