Reconfigurable Framework for Remote Monitoring and Management of Computer Systems

Gülşah Yalçın
Computer Engineering, Middle East Technical University
Ankara, Turkey
Email: gulsah.yalcin@ceng.metu.edu.tr

Halit Oğuztüzün
Computer Engineering, Middle East Technical University
Ankara, Turkey
Email: oguztuzn@ceng.metu.edu.tr

***Abstract*—Remote Monitoring and Management systems are information technology software tools to organize and manage client workstations. They are used by many companies that are willing to minimize their labor cost, collect and measure the data of a variety of clients, administrate them from a single point, in a reliable and secure way. Dynamic profile deployment, dynamic reconfiguration of monitors in response to changes in clients' profiles and creating notifications or running a procedure on the fly are the main features of remote monitoring systems which can be fully expressed by Dynamic Software Product Line Approach. Reconfigurable Framework for Remote Monitoring and Management of Computer Systems aims to provide IT service providers with a dynamically reconfigurable, reusable and easy to define smart monitoring and measurement mechanism. It applies dynamic software product line concepts which are well defined in academic studies but not extensively implemented in business realm. It enables information technology management systems with reusable and autonomously monitoring software assets.**

***Index Terms*—Remote Monitoring and Management, Dynamic Software Product Line, Reconfigurability, Runtime Adaptation**

1. Introduction

1.1. Problem Statement

Increasing demand for complex multidisciplinary systems, fast developing technology trends and need for adaptation to a fast evolving world require large scaled, multi employee companies in almost every field. To ensure the sustainability of the work flow, increase productivity by keeping employees workstations healthy and consistent, keep servers, network areas or other software systems of company always up and running, companies hire Information Technology (IT) staff and managers. Another key point for companies having computer systems is enhancing the quality of service and reducing the risks, therefore they budget for Management Service Providers (MSP). To reduce the cost to employ IT management staff but more importantly diminish the human errors, automate the management of computer systems and administrate them from a single center, Remote Monitoring and Management (RMM) software tools are being used by organizations. RMM software generally has two components, one of which is an admin portal used by an IT personnel and the other is agent deployed to managed client computers. With the help of the agents, RMM system identifies newly joined clients and deploys some default and predefined set of monitoring mechanisms and instructions. There is also need for RMM systems to take actions after deployment and show dynamic behaviors during runtime. At this point, the application of Dynamic Software Product Line (DSPL) Engineering methodology seems well suited for RMM systems. Although there is extensive academic research on DSPL concepts and techniques, the application areas and best practices in business are still preliminary. How DSPL mechanisms are put into practice in RMM product families and what are the effectiveness, benefits and gains in DSPL approach applied to RMM systems are the driving questions for this paper.

1.2. Motivation

A Remote Monitoring and Management System is used to manage clients workstation remotely by aiming automatic, reliable and maintainable management monitors. The features of RMM are varying according to the custom needs. Nevertheless, in all of them, RMM can detect the new clients joining the network and then configure and manage them automatically by the rules prescribed by IT service providers. For example, after new client identification is completed, a default profile defined by IT manager or a specific one to this client is deployed to the workstation and controls the end user achine remotely. In a security concerned profile,the manager can define such a rule that Malware Detection monitor starts to watch the computer and whenever it detects a virus or spy, it removes the malware automatically, or creates a notification to user or Management Service Providers. As another rule concerned with, whenever a CPU Monitor detects that CPU usage is exceeding the 70 percentage, then Event Log Monitor also is set up. Considering these features, remote monitoring and management software is very suitable to model it as Dynamic Software Product Line. With this motivation, Re-

configurable Framework for RMM Systems emerged which facilitates the construction and configuration of remote monitors by applying the dynamic reconfiguration and asset reuse approaches guided by a variability model.

1.3. Contributions

Reconfigurable Framework for RMM Systems contributes the software engineering domain by exploring the applicability of DSPL methodology in the Remote Monitoring and Management domain. Although the framework addresses computer systems, it can also be a guide for monitoring systems in other application fields.

1.4. Organization

In the rest of the paper, initially the background for RMM and DSPL concepts, then commonality and variation points analysis of RMM systems is given, which is an important phase for dynamic software product line development. Then the details of the framework in terms of architecture, self awareness and dynamic binding mechanism and reusable assets are explained.

2. Background

2.1. Remote Monitoring and Management

Remote Monitoring and Management is software tool used by IT/Management service providers to manage and organize their client systems. A RMM system enables Managed Service Providers to setup monitors over client computers/systems to gather information and data pertaining to how the system is performing and to supervise the security and performance related issues. With the help of monitoring mechanism, MSPs or technical support team can execute management tasks autonomously [1]. Remote Monitors are, in general, shaped with three main concerns: Security, Performance and Communication. Under these categories; TCP, Network Bandwidth, CPU, RAM, Folder Size, File Size, Malware Detection and many other monitors are set to clients and their workstations are managed by service providers remotely. An autonomous and self-adaptive remote monitoring and management software is preferred by many company because it helps reduce labor cost and replace the error prone manual management systems. Instead of wasting hours for employed IT support teams, RMM software with a single IT manager can organize the client machines automatically by minimizing human factor and solve the issues without interrupting the client.

2.2. Dynamic Software Product Lines

To react to dynamic needs originating from humans, system or environmental changes and new feature demands in software intensive products, new approaches and solutions have appeared in the literature. Software Product Line (SPL) development is one of leading approaches for large-scale product families in which many of them share the same components and have variable and configurable options and requirements. Software product line approach aims to create a product portfolio with lower cost and high quality by adopting reusable software components. However the more important need; adaptation to variable systems and environment, sensing the changes in context conditions have led to a newer approach: Dynamic Software Product Lines [2]. In todays large and heterogeneous systems, determining the requirements and features before deployment phase is not enough, instead there is an emerging challenge to handle variability during runtime. DSPL approach supports to define the variability points of a system for both pre and post deployment stages and then suggests feature activation and deactivation mechanisms to handle autonomous decision making and then dynamic reconfiguration of software assets during execution. [3]

Dynamic software product lines provide following properties:

- Adaptation to dynamically varying system and environment needs.
- Dynamic reconfiguration of product by applying autonomous decision making at runtime.
- Support for wide range of product families with the help of reusable software components.
- Management of variability driven by an explicit variability model.

2.3. Variability Modeling with Feature Models in DSPL

A distinguishing aspect of a DSPL from manual-managing systems in general is that a DSPL must possess a variability model explicitly, variability management and dynamic adaptation mechanisms must be using this model. In a product family, the most important thing to maximize the reuse of product components is determination of commonalities and variabilities between them. Every single aspect that is common or distinct can be accepted as a variant for this product family. It is crucial to identify the goals for a system, ask domain questions and give explicit answers to these questions, and in return they are also another variability points/variants [4] On the other hand, a product also can have some internal states and vary by the environmental changes during execution cycles. To adopt product to the resulting conditions, some features are activated or deactivated and the system needs to evolve and achieve self adaptation. All possible features that can be active for a product or system are also variation points.

> *"Commonalities and variabilities are modeled from the perspective of product features, âĂIJstakeholder visible characteristics of productsâĂİ in a product line that are of stakeholders concern.* [4]

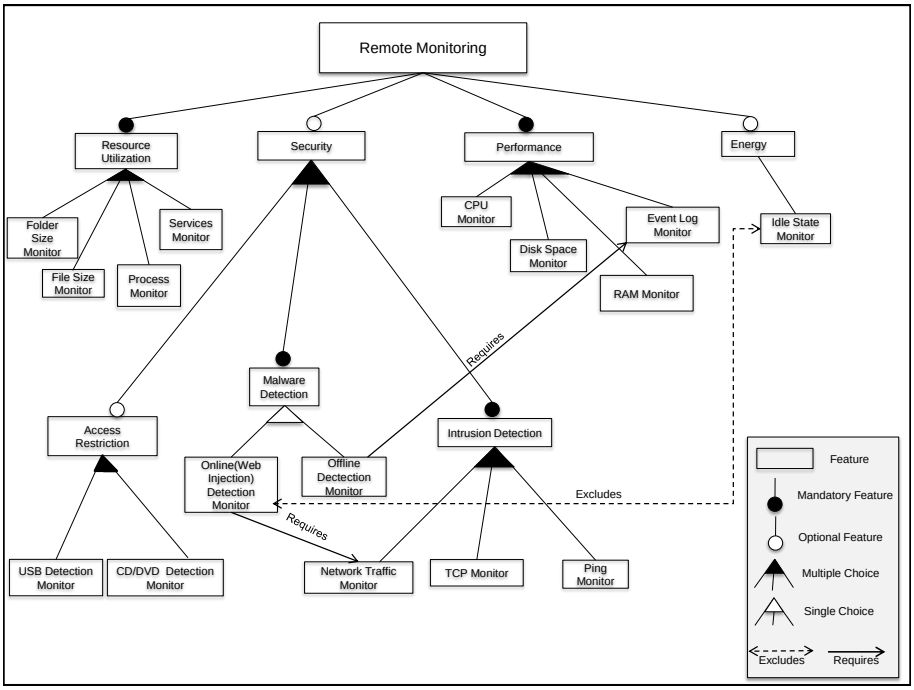

Figure 1: Remote Monitoring System Feature Model

3. Feature Model of RMM Systems

In the RMM systems, management involves monitoring of workstations and collecting the monitor outputs. Therefore determination of monitors according to the profile of clients is the initial step of system development. For the Commonality and Variability management(C&V) of Remote Monitoring (RM), which is the building block of DSPLs, we adopt the aspect and viewpoint oriented approach introduced by Kyo C. Kang and Hyesun Lee in [4]. C&V of a product line can be modeled in many different ways based on different viewpoints adhering to the principle of separation of concerns. In the problem space, user goals and objectives, required quality attributes, and product usage contexts are typically modeled in product line engineering. In the solution space, C&V is modeled for the functional dimension (i.e., capabilities, services), the operating environmental dimension (e.g., operating systems, middleware), and the design dimension (e.g., domain technologies).

3.1. Problem Space Dimension

Some variability points are derived from the goals that are addressed by different assets of a software product line. These goals shape the product features and also the variability model of product line in return. From this point of view, it is useful to identify following concepts for C&V of Remote Monitoring systems:

- *The goal:* Enable IT staff with a remote monitoring and management system for the clients.
- *Usage Context:* Remote Monitoring and Management will be available for various operating systems and it will be working as a background task. It should enable IT technician to configure RMM for different type of user groups as well as individual users.
- *Performance Attribute:* RM must be functioning for up to 2000 online clients at a time.

3.2. Solution Space Dimension

- *Capabilities:* Clients can have different profiles in accordance with the several categories such as Resource Utilization, Security, Performance and Energy. According to profile applied to a client workstation, different monitors are dynamically loaded or unloaded at run time.
- *Operating Environment:* Windows (XP and later)

Figure 1 shows the constructed Remote Monitoring System Feature Model by using the FORM approach [5]. The model reflects only functional dimension of common and variable points belonging to RMMs, and other dimensions are out of the scope of this paper. Every monitor depicted in the model represents a feature of the RMM system. The activation and deactivation of these features determine the current configuration of the system. For some product lines there may exist potentially conflicting or dependent features, RMM system modeled in this framework also has these kind of features. For example, when Malware Detection Monitor or its sub feature Online (Web Injection) Detection is chosen by IT manager for this client, Network Traffic Monitor must also be activated to control the injections. On the other hand, because Online Malware Detection means

TABLE 1: Feature Dictionary of RMM System

Feature	Definition
Resource Utilization	This feature is the first of four main and abstract features of the RMM system. It has four child feature watching the different sources of computer systems: Folder Size, File Size, Process and Service Monitors. Resource Utilization feature is mandatory and the system must have at least one child feature of it.
Folder Size Monitor	This feature continuously watches the size of predefined folder, if the size of it becomes higher than the specified value, this value is reported to the system immediately.
File Size Monitor	This feature continuously watches the size of predefined file, if the size of it becomes higher than the specified value, this value is reported to the system immediately.
Process Monitor	This feature continuously watches the predefined process or set of processes, if the running state of process changes, status is reported to the system immediately.
Service Monitor	This feature continuously watches the predefined service or set of services, if the running state of processes changes, status is reported to the system immediately.
Security	This feature is the second of four main and abstract features of the RMM system. This optional feature aims to detect security vulnerabilities on computer systems. It has 3 different child feature : Access Restriction, Malware Detection, Intrusion Detection.
Access Restriction	This abstract feature watches the drivers of computer systems with restricted access to drivers. USB Detection and CD/DVD Detection Monitors are the concrete child of this feature. When Access Restriction is activated, one of these features must also be activated.
USB Detection Monitor	This feature continuously monitors the USB drive and reports the existence of drive to the system when it is detected
CD/DVD Detection Monitor	This feature continuously monitors the CD/DVD drive and reports the existence of drive to the system when it is detected
Malware Detection	This is an abstract feature to monitor the malware attacks to computer systems. When Security feature is activated, this feature must also be activated. Two concrete child features; Online and Offline Detection Monitors can be alternatively active in a system.
Online Detection Monitor	This concrete feature monitors the attacks that can take place over the Internet. When this feature is activated, Network Traffic Monitor feature must also be activated. Idle State Monitor cannot be active when this feature is active.
Offline Detection Monitor	This concrete feature monitors the changes in operating system caused by the applications of computer systems. It reports the security vulnerability to the system. Event Log Monitor must also be activated when this feature is activated.
Intrusion Detection	This feature is an abstract feature to detect the unauthorized entry to the system. Intrusion Detection Monitor must be activated when Security feature is activated. At least one of three concrete child of it; Network Traffic, TCP and Ping Monitors must also be activated.
Network Traffic Monitor	This feature checks the network traffic of computer system at the specified frequency and when the percentage of usage exceeds the specified threshold, it reports the percentage to the system.
TCP Monitor	This feature monitors the specified TCP Port at the specified frequency. It reports the state changes and the list of connected clients to the system.
Ping Monitor	This feature pings the specified URL at the specified frequency and reports the packet loss percentage when the it exceeds the specified threshold.
Performance	This mandatory feature is the third of four main and abstract features of the RMM system. It has four child feature monitoring the components that can be a measure of performance: CPU, Disk Space, RAM and Event Log Monitors. At least one of these coexisting features must be active in the system.
CPU Monitor	This feature watches the percentage of CPU usage and reports to the system together with the current value when the specified percentage value is exceeded.
Disk Space Monitor	This feature keeps track of the usage of disk space, reports to the system together with the current value when the specified percentage is exceeded.
RAM Monitor	This feature keeps track of RAM usage, reports the system together with the current value when the specified percentage is exceeded.
Event Log Monitor	This feature monitors the Event Log component of Windows Operating System, reports error log to the system together with application name. This feature can also be configured to track error logs for a specific application.
Energy	This optional feature is the last of four main and abstract features of the RMM system. It has one child feature: Idle State Monitor.
Idle State Monitor	It is a concrete feature that monitors the idle state of the system. It continuously checks the indicators of idle state; whether the screen protector is active, a user input is present or the mouse and the keyboard are not used for a certain period of time. It reports the idle state changes to the system.

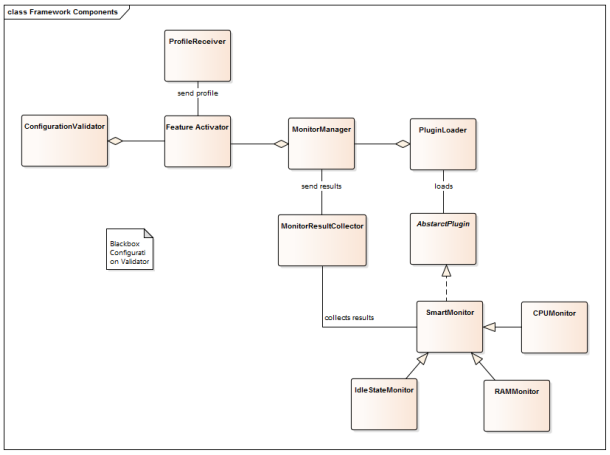

Figure 2: Remote Monitoring System Software Components

that possibly workstation user is online and there is an active network traffic over computer, RMM system excludes the Idle State Monitor.

4. The Framework as a DSPL

A RMM system consists of two components; the first component which is the portal assists IT staff/manager to create and manage monitors and the second one which is the agent manages client workstations according to the profiles assigned them. The framework provides an infrastructure to implement the second part of Remote Monitoring and Management systems.

4.1. Architecture

Reconfigurable Framework for RMM Systems adopts the component based architectural style. After the latest configuration is determined according to the profiles of a client, the monitor components related with the activated features are set up. A change in a profile means that there is a need for new configuration and new active features in accordance with these configuration. The features correspond to monitors, which are dynamically reconfigurable components on RMM systems.

Framework defines a generic and abstract monitor named SmartMonitor and enable the product line engineer to define specialized monitors according to the different features of product. All monitors in the product line of RMM are derived from this class and the specification of monitors differ in accordance with the different behavior of remote monitoring mechanisms (See Figure 2).

SmartMonitor implements the AbstractPlugin interface which enables monitors to run as reconfigurable plug-ins. MonitorManager component of the framework facilitate plug-in loading or unloading with the help of PluginLoader class. SmartMonitors can be dynamically created at runtime and they watch the client system, apply a procedure if threshold exceeds or the specific condition is met. For

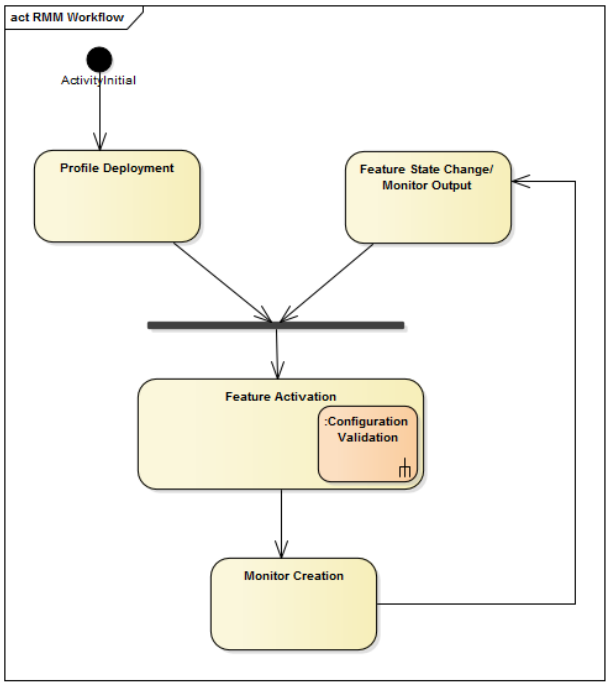

Figure 3: Remote Monitoring System Activity Diagram

example, a RAMMonitor can be implemented in a way that it observes the RAM usage periodically and creates a notification to IT manager if the usage exceeds 70 percentage. To handle notifications and collect the outputs generated by monitors, MonitorResultCollector class is presented by the framework. The results of monitors are interpreted as new states of features by MonitorResultCollector and resulting configuration is sent to FeatureActivator through MonitorManager. FeatureActivator interprets the output and examines if any new activation or deactivation is required. If there is any change, new set of features are sent to ConfigurationValidator to check possible conflicts. As it is stated in the Feature Model of RMM Systems section of this article , RMM have some conflicting and interdependent features therefore ConfigurationManager must handle these situations.

4.2. Context Awareness & Dynamic Binding

The set of active features make up the current configuration of RMM system. New profile deployment is the trigger to create a new configuration and consequently a set of active features. Whenever new profile is gathered by the RMM agent of client workstation during execution, reconfiguration of agent is needed. Besides that, during the silent execution, some conditions arise that affect the states of the features. For example, an IT manager can define a profile in such a way that, whenever Idle State Monitor detects that workstation is in idle state at some point, Offline Detection Monitor of Security Feature must be activated (See Figure 3). The FeatureActivor component of the framework is responsible to determine which features must be

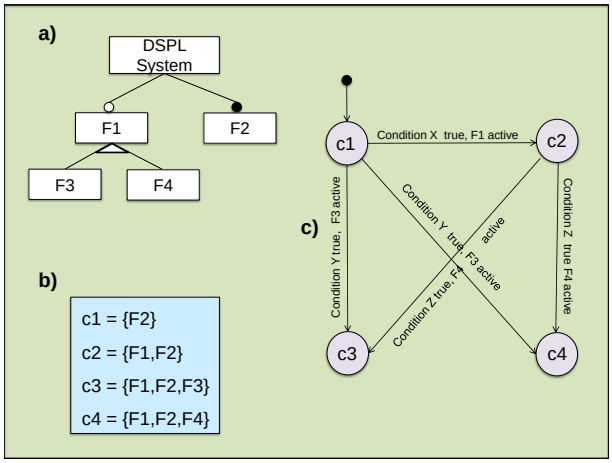

Figure 4: Configuration Validation Example

activated or deactivated conforming to the changes in profile or client workstation modifications (results of the monitors as stated in the previous example) and dynamically binds the necessary configuration to the running system by passing a set of features to ConfigurationValidator component. This behavior of the system is called as context awareness and self adaptation on post deployment phase of the product.

To explain the approach applied in framework more simply, four featured DSPL system can be given as an example (See Figure 4.a). In such a system with the given feature model, four different configurations are possible: c1, c2, c3 and c4. In all these configurations F2 is included because it is a must feature, and either F3 or F4 is present alternatively (See Figure 4.b) The system always monitors the changes in its execution environment and captures the conditions X, Y and Z. When the current configuration is c1 and condition X is true, F1 is eligible to be activated but configuration validation mechanism checks that the resulting set of features F1,F2 does not contain any contradiction and this configuration is one of the available configurations(which is c2). As an another example, the fulfillment of condition Y requires the activation of F3 feature. It is safe to activate this feature if the current configuration is c2, because resulting configuration will be c3 which is in the available list, but it is not safe if the current configuration is c1 or c4 because there are no such feature sets like {F2,F3} and {F1,F2,F3,F4} in the possible configurations (See Figure 4.c).

4.3. Reuse of Assets

All software components of the framework are designed in such a way that they can be reused for different products in remote monitoring and management system product families.

Components like MonitorManager, AbstractPlugin or SmartMonitor are the generic classes defined by the framework. It also presents a set of already implemented and ready to use monitors. They are performance and resource utilization concerned monitors including CPU, Disk Space, RAM, Event Log, Folder Size, File Size, Process and Service Monitors. These are predefined software components and they can be utilized on different RMM implementations as is or can be reconfigured according to different operating environment and system needs.

5. Discussion and Future Work

Reconfigurable Framework for Remote Monitoring and Management of Computer Systems deals with the functional requirements of RMM systems. Work is ongoing to implement the framework based on the dynamic software product line paradigm.

Regarding feature model given in Section 3, only functional dimension of commonality and variability points are evaluated and related monitors are brought up as features of RMM system. For future work, non functional requirements and design decisions of such systems will also be addressed and an extended feature model will be provided.

6. Acknowledgements

This work has been supported by TÜBİTAK ARDEB 1001 program under grant number 215E188. Technical support has been provided by Comodo Group, Inc., Ankara.

References

[1] CGS, "What is Remote Monitoring and Maintenance (RMM)?." http://searchitchannel.techtarget.com/definition, Accessed 5 March. 2016.

[2] R. Capilla, J. Bosch, P. Trinidad, A. Ruiz-Cortés, and M. Hinchey, "An Overview of Dynamic Software Product Line Architectures and Techniques: Observations From Research and Industry," *Journal of Systems and Software*, vol. 91, pp. 3–23, 2014.

[3] C. Cetina, V. Pelechano, P. Trinidad, and A. R. Cortés, "An Architectural Discussion on DSPL," in *SPLC (2)*, pp. 59–68, Citeseer, 2008.

[4] R. Capilla, J. Bosch, K.-C. Kang, *et al.*, "Systems and Software Variability Management," *Concepts Tools and Experiences*, p. 32, 2013.

[5] K. C. Kang, S. Kim, J. Lee, K. Kim, E. Shin, and M. Huh, "FORM: A feature Oriented Reuse Method With Domain Specific Reference Architectures," *Annals of Software Engineering*, vol. 5, no. 1, p. 143, 1998.