Extending a Method of Describing System Management Operations to Energy-Saving Operations in Data Centers

Matsuki Yoshino, Michiko Oba, Norihisa Komoda

Abstract— The authors propose a method for describing system management operations based upon patterns identified by analyzing operations in data centers. Combined with a CMS (Configuration Management System) defined in ITIL®(Information Technology Infrastructure Library), it is possible to calculate energy consumption of an information system managed by management operations described by the proposed method. To demonstrate the effectiveness of the method, examples of saving energy in operations described by the proposed method are shown with an example of calculating the energy savings.

Keywords— System management, Patterns, Green IT, Description method, CMS

I. Introduction

INFORMATION technologies can contribute to solving environmental issues via two different approaches. The first is to reduce the energy consumption of actual IT equipment. According to a survey conducted by the Ministry of Economy, Trade, and Industry of Japan in 2007, approximately 5% of the total electricity consumption in Japan was by IT equipment in 2006. The survey predicted that by 2025, if current trends continue, 20% of the total electricity consumption in Japan will be by IT equipment. Therefore, reduction of electricity consumption by IT equipment, sometimes called "*greening of IT*"[1], has become an urgent issue.

The second approach is to reduce the energy consumption in general human activities by taking advantage of state-of-the-art IT solutions. This is sometimes called "*greening through IT*"[2]. Typical examples of "greening through IT" activities are reducing business trips by using teleconferences, reducing usage of paper by using electronic devices to store and reference information, and replacing physical experiments by computer simulations.

In this paper, the authors focus on the greening of IT in large scale data centers. For a large scale data center, we need to consider energy consumption by the various facilities of the data center as well as by each piece of IT equipment[3]. For facilities such as buildings, research is being conducted on mechanisms to reduce energy consumption while keeping users comfortable[4][5].

Using IT equipment in an energy efficient manner is within the realm of system management, which has the goals of reducing costs and also ensuring the high availability of information systems. Automating system management operations is considered to be a promising solution for attaining the goals of system management[6]. However, although various research efforts have been made in this direction, there are still many problems to be solved with regards to building automated system management environments that are workable and effective for enterprise IT systems[7]. As preparation toward automating system management operations, the authors identified patterns of such operations by analyzing the operations in data centers[8]. The reason for extracting patterns of system management operations is that it is more efficient to devise automation strategies for patterns than for individual operations. In this paper, the authors propose a method for describing system management operations based upon the results of their previous work.

As mentioned, reduction of energy consumption at large scale centers has recently become a very serious issue[9]. Operations to save energy can be considered as an extension of system management operations[10]. To verify the descriptive ability of the proposed method, several energy saving operations listed in [11] are described by the proposed method. Furthermore, the authors shows the method can also be used to describe not only management operations of IT equipment but also those of data center facilities. Integration of system management and facility management operations is effective in ensuring optimal energy consumption by all components and facilities of a data center.

II. Overview of Patterns of System Management Operations

This chapter gives an overview of the results of the previous work[10]. This is necessary for understanding the current paper.

A. Model of System Management Operations

Each system management operation can be formulated as follows:

(1) An operation is described as a directed acyclic graph (DAG). The direction of an arc between nodes indicates the order of execution between the work units denoted by the nodes.

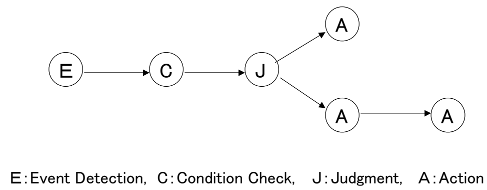

Fig. 1: A Sample Operation

(2) A work unit is categorized into one of the following four categories:

Event Detection (E): Detection of a status change that invokes an operation

Condition Check (C): Check condition for an object related to the operation

Judgment (J): Selection of a subsequent unit of work based upon results of a condition check

Action (A): Actions to be taken on the object of the operation

(3) The following restrictions apply to the graph configuration:

(a) The root of the graph must be a single Event Detection node.

(b) The leaves of the graph must be Action nodes.

(c) For every Judgment node there must exist at least one incoming arc from a Condition Check node.

(d) Targets of arcs starting from a Judgment node must be an Action node or Condition Check node.

Fig. 1 shows an example of a graph representing an operation by this definition.

B. Extracted Patterns

By analyzing the actual system management operations conducted in data centers, four patterns and six sub-patterns were identified and extracted. Table 1 lists the patterns.

Each pattern is described briefly below.

(1) *Pattern 1: Periodic Customary Operation*

This pattern is used for an action that is performed periodically: like invocation, termination, restart, or data manipulation.

(2) *Pattern 2: Non-Periodic Customary Operation*

This pattern is used for operations that are customary but not periodic because they can be performed at a time when a certain condition is satisfied, or should be performed after completion of another operation for which the termination timing is uncertain. A Condition Check (Pattern 2-2) may be performed before an Action is performed.

(3) *Pattern 3: Notification*

This pattern is used for operations that report a change in the status of a monitored object to an operator, so that the operator can then perform an appropriate action if necessary. In this pattern, as in Pattern 2, a Condition Check may be performed before an Action (Pattern 3-2).

(4) *Pattern 4: Error Recovery*

This pattern is used for operations to recover from system failures. An operation of this pattern is invoked when the status

Table 1: Operation Patterns

Category	Sub-category	Pattern ID
Periodic customary	–	Pattern 1
Non-periodic customary	Without Condition Check	Pattern 2-1
	With Condition Check	Pattern 2-2
Notification	Without Condition Check	Pattern 3-1
	With Condition Check	Pattern 3-2
Error Recovery	–	Pattern 4

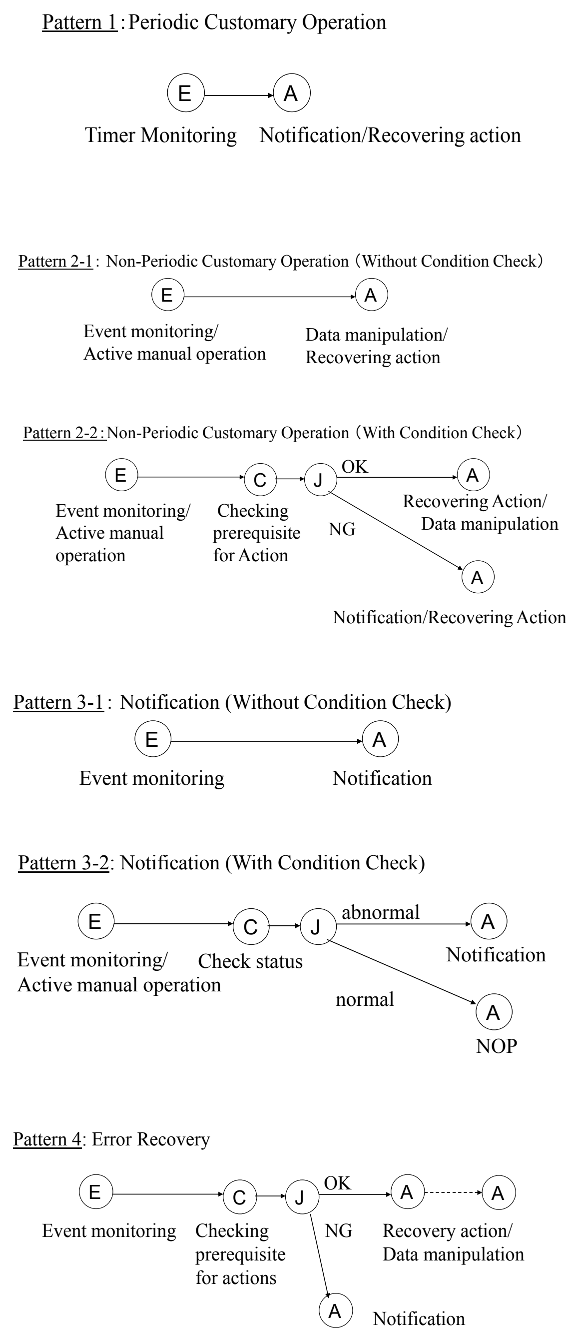

Fig. 2: Graphical Representations of Patterns

of a monitored object changes and after checking prerequisites of Action work units.

Fig. 2 shows a graphical representation of these patterns.

III. A Method for Describing System Management Operations

This chapter describes a method for describing system management operations (a *description method*), based upon the patterns described in Chapter II.

A. Definitions of Terms

In order to describe system management operations, this paper uses several terms defined in ITIL® [12]. ITIL® is a well known and widely accepted framework of best practices for delivering IT services including system management operations. The term definitions are given next.

Configuration item (CI):

Any component that needs to be managed in order to deliver IT services. Information about each CI is recorded in a configuration record within the configuration management system and is maintained throughout its lifecycle by configuration management. CIs are under the control of change management. CIs typically include IT services, hardware, software, buildings, people, and formal documentation such as process documentation and SLAs.

Configuration management system (CMS):

A set of tools and databases that are used to manage an IT service provider's configuration data. The CMS also includes information about incidents, problems, known errors, changes and releases, and may contain data about employees, suppliers, locations, business units, customers, and users. The term CMDB (configuration management database) is often used as a synonym of CMS.

B. Scope of the Description Method

The description method proposed in this paper provides abstract level semantics for descriptions of system management operations in semi-formal pseudo-code style. Although it is necessary to define precise concrete syntax and semantics to implement automation tools for operations, conceptual level design is essential as the first step. Once the conceptual level definition is established, it is possible to proceed to design concrete level rendering of conceptual level designs, in either XML, textural, or graphical format.

The ultimate purpose of the description method is full automation of system management operations. For such full automation, a CMS defined as in ITIL® is indispensable. Therefore, this paper assumes the existence of a CMS.

C. Requirements of the Description Method

This subsection discusses requirements of the description method.

By the definition of CI, not only tangible assets like servers and software, but also intangible assets like process documents of system management operation are considered as CIs and managed by a CMS. Each CI managed by a CMS should be identified by a unique identifier. Thus, each description of a system management operation must have a unique identifier.

```
<Operation> ::= <Operation ID><Pattern ID><Operation-body>
<Operation-body> ::= <Event-Detection> <Action-body>
<Action-body] ::= <Condition-Check-list> <Judgment>
                  { <Condition> <Action-list>}+ |
                  <Action-list>
<Condition-Check-list> ::= <Condition-Check> {<Condition-Check-list>}
<Action-list>  ::= {<Action>}
<Operation ID> ::= Operation : operation name
<Pattern ID> ::=  Pattern: pattern name
<Event-Detection> ::= Event: <Target> event detection description
<Condition-Check> ::= Condition: <Target> condition check description
<Judgment> ::= Judgment
<Condition> ::= When: check result description
<Action> ::= Action: <Target> action description
<Target> ::= Target: target description
```

Fig. 3: BNF Representation of the Description Method

Because all system management operations belong to one of the patterns described in Chapter II, the description must contain information about the pattern.

Work units that consist of operations are to be described. The CMS stores information about work units and unique IDs that are assigned. The CMS also stores information about the targets of work units, such as devices to be monitored by an Event-Detection work unit and artifacts to be affected by an Action work unit, and unique IDs that are assigned.

Because an Action work unit may issue an event and can invoke another system management operation, complex operations can be implemented by a sequence of operations denoted by one of the patterns.

Action work units subsequent to a Judgment work unit must be marked by result of a Condition-Check work unit to identify the condition of their execution.

D. BNF representation of the Description Method

Considering the scope and requirements explained in the previous sections, system management operations can be described in the format shown in Fig. 3, in BNF notation. In this paper terminal symbols denoted by italics are given in natural language descriptions for explanations. Note that Fig. 3 shows the grammar for the pseudo-language used for explanations, and does not give the grammar used for input into system management tools.

For input into automation tools, a concrete definition of the format (for example, an XML schema and interface definition to the CMS) needs to be specified from the format defined in Fig. 3. However, the definition given in Fig. 3 is adequate for explaining the usefulness and power of the proposed description method.

IV. Description of Energy Saving Operations by the Proposed Method

This chapter gives examples of energy saving operations, using the proposed description method.

Operation : shutdown
Pattern : Periodic Customary
Event : Target : timer at 11 p.m. on business days
Action : Target : Server123 turn-off

Operation : wakeup
Pattern : Periodic Customary
Event : Target : timer at 6 a.m. on business days
Action : Target : Server123 turn-on

Fig. 4: Example of a Periodic Customary Pattern Operation

Operation : rolling upgrade
Pattern : Non-periodic Customary
Event : Target : timer at specified time during off peak period
Action: Target: turned_off_servers turn-on
Action: Target: turned_off_servers upgrade
Action: Target: all_servers
configuration change to use turned_off_servers
Action : Target : unused_servers upgrade
Action : Target : unused_servers turn-off

Fig. 5: Example of a Non-Periodic Customary Pattern Operation

A. Examples of Energy Saving Operations

Fig. 4 gives an example of the periodic customary pattern. This sample describes an operation to shut down Server123 (which hosts an application that is used only in business hours) at 11 p.m., and to wake it up at 6 a.m. This shutdown saves the electricity consumed unnecessarily by the server during off hours. If the CMS stores power consumption information about the server, it is possible to estimate the amount of electricity that is saved by these operations.

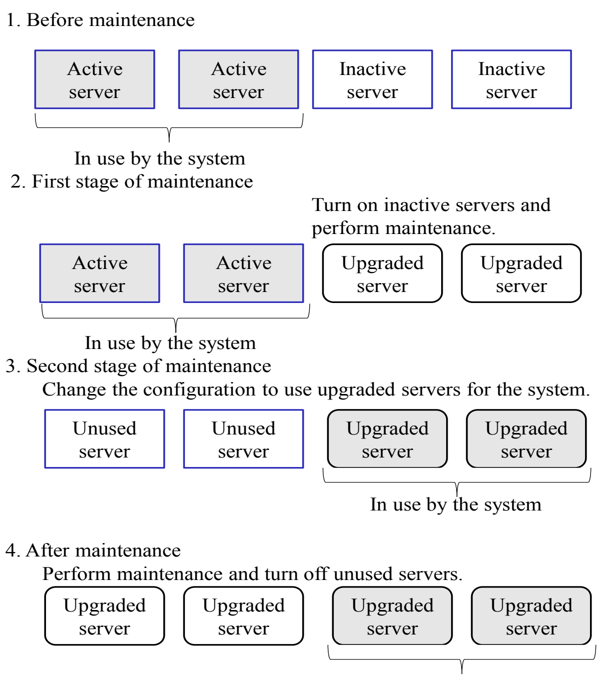

Fig. 6: Rolling Upgrade

Operation : environment change notification
Pattern : Notification
Event : Target : environment-sensor234 when a threshold is reached
Action : Target : Environment-management-server
issue an event notifying environment change

Fig. 7: Example of a Notification Pattern Operation

Operation : Resolve Hot Spot
Pattern : Error Recovery
Event : Target : Temperature_sensor1
when temperature exceeds threshold
Condition: Target: Server-pool availability of servers
Judgment:
When: Available
Action : Target : Server-pool Provisioning of servers available
Action : Target : in-use-Servers-and-newly-added-servers
Reconfigure to move workload to newly added servers
Action: Target: in-use-Servers turn-off
When: Not-available
Action : Target : System-Operator
Notify that temperature exceeds the limit

Fig. 8: Example of an Error Recovery Operation

Fig. 5 gives an example of the non-periodic customary pattern. The operations in Fig. 5 use a “rolling upgrade”[13], a technique for performing maintenance operations without shutting down services. Fig. 6 explains the rolling upgrade. During these operations, the minimum number of servers are powered on.

Fig. 7 gives an example of the notification pattern. In this operation, environment information useful for saving energy is reported: for example, information such as temperature and humidity changes in the server room. Because operations of this pattern only perform notifications, actual energy saving operations are performed by subsequent automated operations or by manual operations. For example, actions taken when an environment change is reported might include adjusting the temperature and humidity of the server room in order to avoid condensation or static electricity, and moving the workload from servers in a hot area to servers in a cooler area in order to improve the efficiency of cooling in the server room.

Fig. 8 gives an example of the error recovery pattern, which can be applied if actions performed manually by operators in the example shown in Fig. 7 can be performed automatically. Fig. 9 shows the situation of operations of Fig. 8.

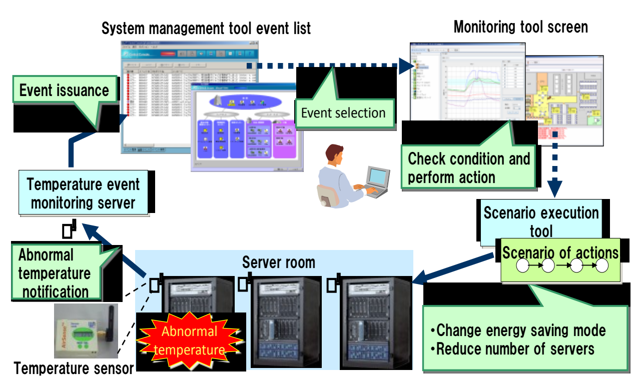

Fig. 9: Overview of Fig. 8 Error Recovery Operations

Operation : Dynamic Workload Management -add servers
Pattern : Error Recovery
Event : Target : Workload-Sensor-for-the-system
when the workload reaches over upper limit
Condition: Target: Server-pool availability of servers
Judgment:
When: Available
Action : Target : Server-pool Provisioning of servers available
Action : Target : in-use-Servers-and-newly-added-servers
Reconfigure to utilize newly added servers
When: Not-available
Action : Target : System-Operator
Notify that workload exceeds the limit

Operation : Dynamic Workload Management-delete servers
Pattern : Error Recovery
Event : Target : Workload-Sensor-for-the-system
when the workload reaches under lower limit
Condition: Target: Number-of-servers-used-for-the-system
Judgment:
When:
Over-Lower-Limit
Action : Target : Servers-used-by-the-system
Reconfigure to reduce number of servers
Action : Target : Removed-servers
Turn off
When:
Lower-Limit

Fig.10 Another Example of an Error Recovery Operation

Fig. 10 gives one more example of operations in the error recovery pattern. This example assumes that the system configuration is scalable and unused servers in a data center are managed in the CMS as a server pool. These operations have two parts: adding servers when the workload is predicted to exceed the capacity of the current servers, and removing servers when the workload is predicted to stay lower than the capacity of the current servers.

B. *Evaluation of the Description Method*

In previous work[5], system management operations are described in natural language. For example, the operation in Fig. 10 is described as follows:

> When the workload of the system is predicted to remain low based on monitored statistics, unneeded servers are turned off. Likewise, when the system workload is predicted to increase based on monitored statistics, inactive servers are turned on.

By using the proposed description method, targets of operations are clearly stated and the procedures to be performed can be precisely defined. By using a standardized description method like the proposed one, it is possible to normalize the level of descriptions even when a team of designers with different skills composes them. Furthermore, a tool can be used to check the description format. Although the purpose of the proposed description method is to automate system management, human operators can easily understand standardized descriptions. This makes review sessions of system management operations more efficient and productive.

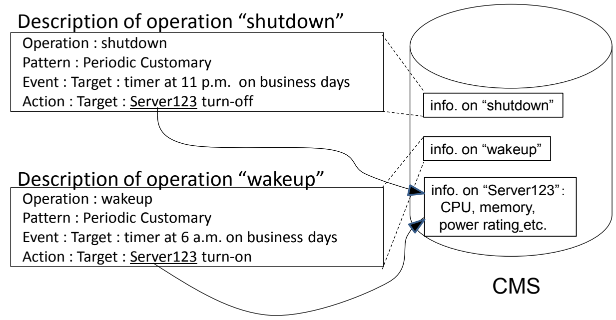

Fig. 11: Relationship between Descriptions of System Management Operations and the CMS

The amount of energy saved by operations described in the proposed description method can be estimated, assuming that the CMS stores the appropriate information (such as, power consumption of devices in a data center, and estimation of workloads) as shown in Fig. 11. This is useful for estimating energy consumption changes, when system management operations are modified or new devices in the data center are installed. The next chapter gives an example of estimating such changes in an actual system.

V. Example of Estimating Energy Saved by System Management Operation

This chapter gives an example of estimating the energy saved by a system management operation in an actual data center.

A. *Overview of the System Used in the Estimation*

Table 2 summarizes the system used in the estimation. The system is a work-collaboration system used in a large scale enterprise. Because the enterprise headquarters are located in eastern Asia, during off-peak hours in the eastern Asia area, the system workload reduces to about 25% of the peak hours. Fig. 14 shows the typical workload variation in a business day.

Fig. 13 shows the system configuration of this system. The Web/application servers in Fig. 13 are scalable. Therefore, during off-peak hours, the number of these servers can be

Table 2: System Used in the Estimation

Types of applications	Web-based collaboration in a large-scale enterprise, including email, scheduling, and BBS
Number of registered users	More than 300,000
Geographical distribution of users	More than 90% of users are in eastern Asia
Number of servers	Web/application servers: 30 Database servers: 5 Other servers: 50
Power consumption of servers (kW/h)	Web/application servers: 0.3 kW Database servers: 0.7 kW Other servers: 30 kW
Workload during off-peak hours	25% of peak workload in eastern Asia off-peak hours

```
Operation : Collaboration system management -add servers
Pattern : Periodic Compulsory
Event : Target : Timer
          at 8 a.m. on business day
Action : Target : Turned-off-servers
           Turn on
 Action : Target: Servers-for-this-system
           Reconfigure to utilize turned on servers

Operation : Collaboration system management-delete servers
Pattern : Periodic Compulsory
Event : Target : Timer
         at 10 p.m. on business day
 Action : Target : Servers-used-by-the-system
          Reconfigure to reduce number of servers by 22
 Action : Target : Removed-servers
           Turn off
```

Fig. 12: Operations Used in the Estimation

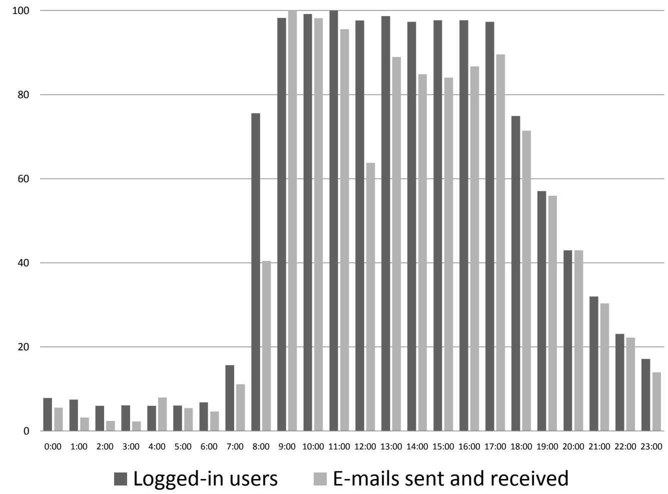

Fig. 14: Hourly Workload Change in a Work-Collaboration System Over One Day

reduced to approximately 25% of those used at peak time. Database servers and other servers, however, are not scalable and the number of these servers cannot be reduced even when the workload of the system is very small.

B. System Management Operation Used for the Estimation

Because peak hours and off-peak hours are predictable for this system, an operation of the periodic customary pattern is applicable. Fig. 12 shows descriptions of the operations. To estimate the reduction in the amount of electricity by the system management operations, information on changes in power usage of servers and a calendar of the company's working days are necessary. This information is expected to be stored in the CMS.

If the workload of the system can be monitored and workloads can be predicted based upon the data accumulated during monitoring, operations of the error recovery pattern similar to those given in Fig. 10 can be applied. Operations of the error recovery pattern are more flexible than those of the periodic customary pattern, because operations of the error recovery pattern can automatically handle unexpected changes in workload. Furthermore, in this case, information on the workdays calendar is not needed.

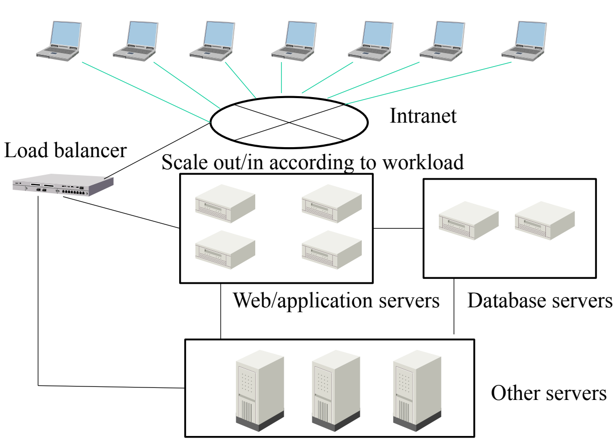

Fig. 13: Server Configuration in the Model Case

C. Estimating the Energy Saved by the Operation

As described in Fig. 12, the number of reduced servers during off-peak hours is 22. The total number of off-peak hours is 115 days * 24 hours = 2,760 hours for weekends and holidays and 250days * 10 hours = 2,500 hours for work days, giving a total of 5,260 hours. Assuming that the information stored in the CMS about the power consumption of web/application servers is 0.3 kW/h, the amount of saved electricity can be calculated as 22 * 5,260 hours * 0.3 kW/h = 34,716 kW. This is equivalent to the amount of electricity consumed by eight average Japanese households in one year.

The amount of electricity saved in this scenario is not trivial. However, when looking at the entire system, only 0.35% of total power consumption is reduced. This is due to the existence of a large number of resource-intensive servers that are not scalable, as described in Table 2. From this analysis, to achieve greater power reduction in the system, the architecture must be changed so that the number of "other servers" can be changed according to the system workload. In planning the next generation of the work-collaboration system, such an estimation is valuable for planning an energy efficient system.

D. Estimation of Improved Architecture

The estimation described in the previous section clarifies problems in the current architecture. To make the system more energy conscious, it is necessary to transform non-scalable servers to scalable servers and to replace old power-consuming servers by new power-efficient ones.

Recent versions of the work-collaboration system use an improved architecture that enables non-scalable servers of the former architecture to be scalable. The basic idea of the new architecture is to separate the functionality that was performed by non-scalable servers into scalable parts and non-scalable parts, thereby increasing the number of scalable servers. By this improvement, 34 servers among the 50 "other servers" could be replaced by 16 scalable web/application servers

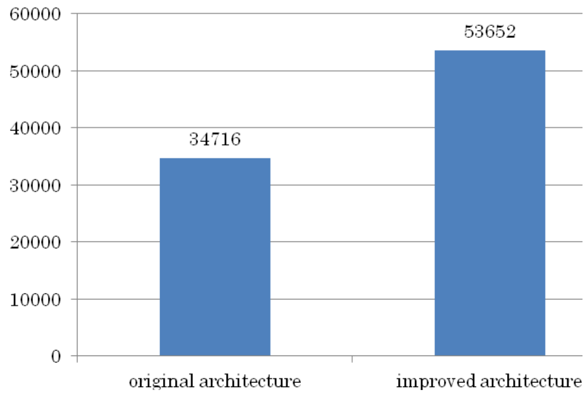

Fig. 15: Comparison of Electricity to be Saved in a Year

(which could be replaced by hardware that consumes less power), and 4 non-scalable database servers (which could not be replaced by more power-efficient hardware). The power saved by adopting the improved version of the work-collaboration system and the operation of reducing the number of running scalable servers while off-peak hours can be calculated as follows: 12* 5,260 hours * 0.3 kW/h= 18,936 kWh. Summing the savings of 18,936 kW with the savings of 34,716 kW described in section C gives the total saved energy as 53,652 kW. This means that 0.9% of the total power consumption of the entire system can be saved by reduced operation in off-peak hours using the improved architecture.

Fig. 15 shows how much electricity can be saved by comparing the estimates of the amount of electricity saved with energy-saving operations in the original architecture (described in section C) with the amount saved by using the improved architecture. By improving the architecture, the amount of saved electricity is more than 150% of the original architecture.

As shown above, the combination of a CMS and the proposed description method is powerful and useful for assessing the energy consumption when changing the system architecture or replacing servers. For the estimation shown above, calculations were performed manually. However, for large scale data centers, manual calculations are not realistic and the existence of a CMS containing system management operation descriptions is required for periodic management of energy efficiency.

VI. CONCLUSION

In this paper, the authors propose a method for describing system management operations based on patterns of system management operation. The authors demonstrate the effectiveness of the method by showing examples of descriptions of energy saving operations, including not only IT devices but also data center facilities like environmental sensors. In order to demonstrate the effectiveness of the combination of a CMS and the proposed description method, estimates of energy savings by system management operations are described for a real enterprise system. The estimates clarify architectural problems related to energy efficiency, and show the expected effects of an improved architecture.

REFERENCES

[1] J. Lamb: "Greening of IT, The: How Companies Can Make a Difference for the Environment," IBM Press, 2009.
[2] B. Tomlinson: "Greening through IT: Information Technology for Environmental Sustainability," MIT Press, 2010.
[3] D. E. Popescu, M. F. Prada, and A. Mancia: "Some Aspects about the New Green Data Center Challenge", in *Proc. of the 11th WSEAS International Conference on Sustainability in Science Engineering(SSE'09),* pp.230-235, 2009.
[4] S. Lin, X. Guo, W. Chen, Z. Zhang, and Y. Lin, "An Automation Model for the Building Energy Management Systems. A Theoretical Study," in *Proc. of the 10th WSEAS International Conference on Robotics, Control and Manufacturing Technology (ROCOM'10)*, pp.41-46, 2010.
[5] D. E. Popescu, M. F. Prada, and D. Gombos: "Building Management Systems – Tool for Green Building and for Monitoring Building Degradation", in *Proc. of the 11th WSEAS International Conference on Sustainability in Science Engineering(SSE'09),* pp.336-339, 2009.
[6] J.O. Kephart, and D.M. Chess, The Vision of Autonomic Computing, *IEEE Computer*, Vol.36, Issue 1, pp. 41-50, 2003.
[7] S. Dobson, R. Sterritt, P. Nixon, and M. Hinchey, "Fulfilling the Vision of Autonomic Computing", *IEEE Computer*, Vol. 43, Issue 1, pp.35-41, 2010.
[8] M. Yoshino, N. Komoda, and M. Oba, An Analysis of Patterns for Automating Information System Operations, *WSEAS Tran, on Information Science & Applications,* Issue 11, Vol.5, pp.1618-1627, 2008.
[9] S. Murugesan, Harnessing Green IT: Principles and Practices, *IT Professional*, vol. 10, no. 1, pp. 24-33, 2008.
[10] G. Spafford, The Governance of Green IT: The Role of Processes in Reducing Data Center Energy Requirements, IT Governance Ltd., 2008.
[11] M. Yoshino, N. Komoda, N. Nishibe, M. Oba, Classification of Energy-Saving Operations from the Perspective of System Management, in *Proc. of 8th IEEE International Conference on Industrial Informatics 2010 (INDIN2010)*, pp. 651-656, 2010.
[12] U.K. Office of Government Commerce, *The Official Introduction to the ITIL Service Lifecycle,* Stationery Office, 2007.
[13] E.A. Brewer, "Lessons from Giant-Scale Services," *IEEE Internet Computing*, vol. 5, no. 4, pp. 46-55, July/Aug. 2001.

TRADEMARKS

Matsuki Yoshino Born in 1958. He earned his B.S. degree in mathematics from University of Tokyo in 1982 and M.S. degree in computer science from Graduate School of Engineering and Applied Science of Columbia University in the city of New York in 1988.

He joined Hitachi Ltd. in 1982. He participated in various development projects as compilers, CASE tools, and application servers. From 2003 to 2006, he participated in Japan's national project, "business grid computing project". He is currently a senior chief engineer at Software Division, Hitachi Ltd.

Mr. Yoshino was a board member of Information Processing Society of Japan(IPSJ) from 2007 to 2009. He is an advisory committee representative of World Wide Web Consortium and a voting representative of Organization for the Advancement of Structured Information Standards(OASIS) of Hitachi Ltd. He is a member of IPSJ and IEEE. He can be reached by an e-mail address: matsuki.yoshino.pw@hitachi.com.

Michiko Oba Born in 1957. She earned her B.S. degree at Japan Women's University in 1982. She received her Ph.D degree in computer science from Graduate School of Information Science and Technology, Osaka University in 2001.

She joined Hitachi Ltd. in 1982. She worked at Systems Development Laboratory and Software Division for more than 25 years. From 2010, she is a professor at the School of Systems Information Science, Future University – Hakodate.

Prof. Oba is a board member of IPSJ from 2009. She is a member of IPSJ, IEEJ(The Institute of Electrical Engineers in Japan) and IEEE. She can be reached by an e-mail adress : michiko@fun.ac.jp.

Norihisa Komoda Born in 1950. He earned his B.S.and M.S. degreen at Faculty of Engineering, Osaka Universigy in 1972 and 1974 respectively.. He received Ph.D degree from Osaka University in 1982

He joined Hitachi Ltd. In 1974. He worked at Systems Development Laboratory for 17 years as a researcher. From 1991, he joined University of Osaka as a faculty member. He is currently a professor at Dept. of Multimedia Engineering, Graduate School of Information Science and Technologies, Osaka University

Prof. Komoda has been awarded IEEJ Fellow (2008) , IEEJ Technical Development Award, IEEJ, Japan(2006), IEEJ Technical Development Award, IEEJ, Japan(2000) IEEJ Distinguished Paper Award, IEEJ, Japan(1998), Award for Outstanding Technology, SICE, Japan(1987), Award for Outstanding Paper, SICE, Japan(1986), and Kusumoto Award (Best Student Award), Osaka University(1972) . He is a member of IEEJ and IEEE. He can be reached by an e-mail address: komoda@ist.osaka-u.ac.jp.