

ベイズ推論を用いたITシステム管理向け 構成情報推定方式の提案

坂下 幸徳^{1,2,a),b)} 東条 敏^{2,c)} 敷田 幹文^{3,d)}

概要: データセンターのITシステムは、マルチベンダーのハードウェア導入や仮想化技術の導入が進んでいる。これにより、ハードウェアと統合管理ソフトウェア間の通信インターフェースの違いや情報の隠蔽化が生じ、統合管理ソフトウェアにてITシステムの構成情報を一元管理することが困難になっている。その結果、一元管理された構成情報をもとに障害原因を特定する機能など統合管理ソフトウェアの先進機能の適用範囲が限られてしまう。

そこで、本論文では、ITシステムの標準モデルCIMとログデータを用いたベイズ推論による構成情報推定方式を提案する。これにより取得不可な構成情報を補完することで、統合管理ソフトウェアの先進機能の適用範囲を拡大する。

A Presumption Method of IT Resource Information by Bayesian Inference for Management System

YUKINORI SAKASHITA^{1,2,a),b)} SATOSHI TOJO^{2,c)} MIKIFUMI SHIKIDA^{3,d)}

Abstract: In recent years, a large-scale data center advance in the adoption of multi vender hardwares and virtual environments. Therefore, a unitary configuration management of the management system software is difficult. Because, it is a difference of the interface between a hardware and a management system software and concealment of configuration information. As a result, the coverage of technology of a management system software, that is root cause analysis by using configuration information, has been limited. In this paper, we propose a presumption method of IT resource information by Bayesian inference for management system software by using CIM and log data. This proposal enables configuration information which cannot be collected to be presumed. The support of advance management technology expands.

1. はじめに

近年、年々増加するITリソースや仮想化技術の導入に

より、データセンター内のITシステム数が年々増加している。しかし、多くのデータセンターでは、これを管理する管理者の数は一定もしくは減少している。このような中、データセンターの管理者は、少人数で多量のITシステムのサービスレベルを維持するため、管理をしなければならない。そこで、管理者の作業負担を軽減すべく、サーバ、ネットワーク、ストレージなどデータセンターのITシステムを運用管理する管理技術が進化している。

一例として、障害が発生した際の管理者の負担を軽減する技術を取り上げる。ITシステムが大規模化、複雑化したため、性能劣化などの障害が発生した場合、どのリソースが原因で障害となっているのが特定することが難しくなっている。そこで、サーバ、ネットワーク、ストレージなど

¹ (株)日立製作所 横浜研究所
Yokohama Research Laboratory, Hitachi Ltd., Yokohama,
Kanagawa 244-8555, Japan

² 北陸先端科学技術大学院大学 情報科学研究科
School of Information Science, Japan Advanced Institute of
Science and Technology, Nomi, Ishikawa 923-1292, Japan

³ 北陸先端科学技術大学院大学 情報社会基盤研究センター
Research Center for Advanced Computing Infrastructure,
Japan Advanced Institute of Science and Technology, Nomi,
Ishikawa 923-1292, Japan

a) yukinori.sakashita.hk@hitachi.com

b) sakasita@jasit.ac.jp

c) tojo@jaist.ac.jp

d) shikida@jaist.ac.jp

の IT システムから通知される障害通知と、接続関係などを格納した構成情報を使い、統合管理ソフトウェアにて、どの部位が障害の根本原因となっているかを解析することで、障害発生時の管理者の原因究明作業における負荷を軽減する管理技術がある。

これら高度化された管理技術は、サーバからストレージまでの End to End の一元管理された構成情報を前提とし実現している。しかし、データセンターの大規模化にともない IT システムが拡張し続けているデータセンターでは、異なるベンダーの機器で構成されることが多く、統合管理ソフトウェアと IT システム間の通信インターフェースが様々となることから、構成情報を一元管理するのが難しい。その結果、上記で述べた障害原因解析技術のような高度化された管理技術の適用範囲が限られてしまっている。

そこで、本論文では、ベイズ推論を用い IT システムより情報取得が行えなかった構成情報を補完を行うシステムを提案し評価する。

以下 2 章では、管理者の負荷を軽減する管理技術に関する従来研究を紹介する。次に 3 章にて提案方式の詳細を述べ、4 章にて提案方式を評価し、最後に 5 章で考察を述べる。

2. 従来研究

管理者の負荷を軽減する管理技術の代表的な研究として、大量発生する SNMP Trap などの障害通知から障害の根本原因を特定する障害原因解析技術がある [1][2][3]。図 1 に障害原因解析技術の処理の流れを示す。障害原因解析技術では、予めサーバ、ネットワーク、ストレージの構成情報を収集し、事前に定義している IF-THEN ルールにて記述された解析用の汎用ルールと組み合わせることで、データセンターの IT システムに合わせた解析ルールを事前に生成する。その後、障害が発生した際、この解析ルールと、受信した障害通知情報を用い障害の根本原因の特定を行う。この障害原因解析技術により、障害発生時における管理者の原因調査にかかる負荷を軽減する。

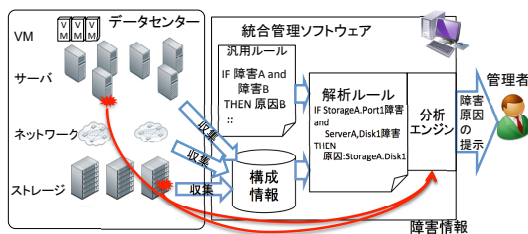


図 1 障害原因解析技術

Fig. 1 Root Cause Analysis Technology

その他の高度な管理技術として、管理の自律化を目指した技術がある。一例としては、性能劣化の原因を解析した後、性能とコストの観点から VM やストレージのデータの

適正な配置場所を求め、自動でデータ配置させる自動データ適正配置技術がある [4][5][6]。図 2 に自動データ適正配置技術の処理の流れを示す。自動データ適正配置技術では、サーバ、ストレージの構成情報と性能情報から性能ボトルネックとなっているリソースを特定し、事前に定義している SLA と、VM のデータが格納されているメディアのコストや性能情報から、データの適正配置場所を算出し、データの移動計画を求め移動させる。この自動データ適正配置技術により、管理者の定期的な性能チューニングにかかる作業負荷を削減する。

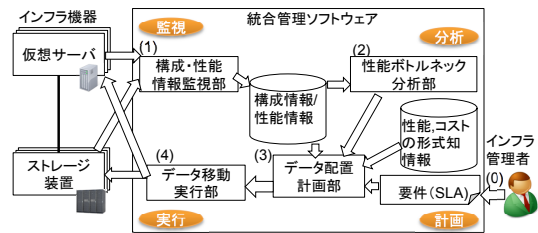


図 2 自動データ適正配置技術

Fig. 2 Automatic Optimization of Data Location Technology

これらのいずれの技術も IT システム全体の構成情報が必要不可欠であり、構成情報が取得出来ない IT システムには適用できない。特に、マルチベンダーの機器が導入されているヘテロ環境においては、SMASH(Systems Management Architecture for Server Hardware)[7] や SMI-S(Storage Management Initiative-Specification)[8] をなどの、標準仕様化インターフェースを利用することにより、適用範囲の拡大はある程度は狙えるものの完全ではな。標準仕様をサポートされていない機器がデータセンター内にあった場合には、一部の機器の情報が欠落してしまい、高度化された管理技術の適用範囲が限定的になる。

3. ベイズ推論を用いた構成情報推定方式

3.1 方針と適用ターゲット

本論文では、IT システムから取得できない構成情報をベイズ推論を用い推定する方式を提案する。

ベイズ推論は、Thomas Bayes により提唱されたベイズの定理を利用した推論機構である。ベイズの定理は、事象 A が起きた後での、事象 B の確率 $P(B|A)$ を算出する定理であり、提案方式では本定理を用い構成情報の推定を行う。

図 3 に、本提案方式を適用するターゲットの一例を示す。本例では、サーバとストレージ間のネットワークに関する構成情報が取得出来なかったケースを想定する。従来までは、ネットワークの構成情報が収集出来ないため、2 章に示す障害原因解析技術が適用出来なかった。本提案方式では、サーバから障害通知を受け取った際、障害通知を行ったサーバが、どのネットワークやストレージと接続関係にあるのかをベイズ推論を用い構成情報を推定することで、

障害原因解析技術を適用可能とする。

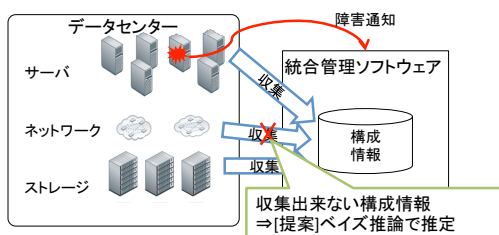


図 3 提案方式のターゲット例

Fig. 3 Example of Target of Proposal Method

3.2 推論手法

推論手法に関して述べる。まず、想定する IT システムの例を図 4 に示す。

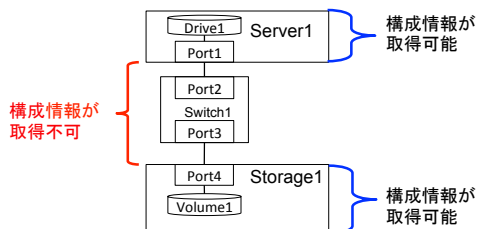


図 4 IT システムの構成例

Fig. 4 Example of Assumption System

図 4 の環境では、Server1 が Switch1 を経由し Storage1 に接続されている。このような環境において、Switch1 の構成情報 (接続関係) が取得出来ない場合を想定する。

図 5 に、各 Port の接続関係が発生する確率について示す。例えば、Server1 の Port1 が Switch1 の Port2 と接続関係にある場合の発生確率は $P(\text{Port2}|\text{Port1})$ となる。これらの構成の発生確率をベイズの定理に従い算出する。さらに、これら発生確率をネットワーク化 (ベイジアン・ネットワーク) し、取得出来ない構成情報を推定する。また、図 4 に示すように、実際の IT システムでは、Server1 の Port1 と Storage1 の Port4 は接続されていないが、本推論手法におけるベイジアン・ネットワークでは、全ての IT リソースが接続関係にあると仮定し、 $P(\text{Port4}|\text{Port1})$ のように実際には接続されていない接続関係の発生確率も含みネットワークを構築する。

このように、ベイジアン・ネットワークを構築することにより、図 4 の Server1 の Port1 から Storage1 の Port4 までの構成情報は、式 (1) から式 (4) の発生確率を求め大小比較することで推定する。

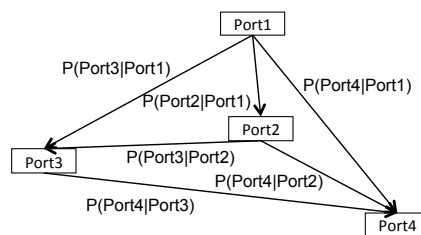


図 5 発生確率のネットワーク

Fig. 5 Example of Resource Probability Network

$$P(\text{Port3}|\text{Port1})P(\text{Port4}|\text{Port3}) \quad (1)$$

$$P(\text{Port2}|\text{Port1})P(\text{Port3}|\text{Port2})P(\text{Port4}|\text{Port3}) \quad (2)$$

$$P(\text{Port2}|\text{Port1})P(\text{Port4}|\text{Port2}) \quad (3)$$

$$P(\text{Port4}|\text{Port1}) \quad (4)$$

3.3 学習

ベイジアン・ネットワークでは、構成情報を推定する際に重要となるのが、事前に学習し準備するそれぞれの構成の発生確率 (事前確率) である。そこで、この発生確率を学習させるために、DMTF (Distributed Management Task Force, Inc) が定める IT システムの標準モデルである CIM (Common Information Model) [9] 及び、各種 IT システムが出力するログデータを学習データとして利用する。

(1) CIM を使った学習

図 6(a) に CIM で表現された IT システムの一部を示す。CIM では、IT システムの構成情報及びその接続関係がモデル化されている。さらに、CIM_ComputerSystem と CIM_FCPort 間に複数の関連が張られているように、各リソースの関連は、接続関係の意味の違いにより複数張られる。つまり、接続関係が複数あるリソースは、様々な意味の関連をもつことから、より関連が強い接続関係と見なすことが出来る。そこで、CIM で規定されたモデルを学習データとし、あるリソースをから接続関係にあるリソースの出現回数をカウントすることで発生確率を算出する。これにより、どのリソース同士が接続される可能性が高いのかを知り得ることが出来る。

また、CIM はオブジェクト指向により IT システムをモデル化しているため、図 6(b) に示すように継承関係がある。そこで、この継承関係に着目し、CIM_FCPort などの subclasses での学習に留まらず、CIM_NetworkPort, CIM_LogicalPort といった親クラスについても、学習対象とする。これにより、CIM にてモデル化されていない新たな IT システムが、既存のモデル化済みの IT システムと似ている場合、その接続関係を推定することが可能となる。これにより、管理者が新しい IT システムを導入した際、既存の IT システムとの類似性から新しい IT システムの接続関係を推測する、いわゆる管理者の勘を表現する。

(2) ログデータを使った学習

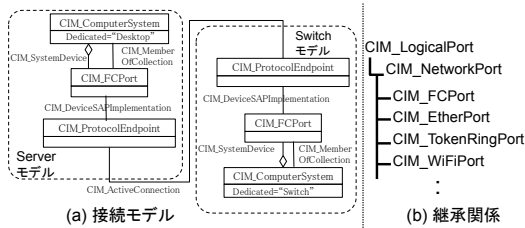


図 6 Common Information Model の一部
Fig. 6 A Part of Common Information Model

更に、学習データとして、各種 IT システムが出力する syslog などのログデータを学習データとして利用する。具体例としては、Server1 の syslog ファイルを対象とし、Port1 に関連したレコードを抽出した後、情報取得が出来る箇所の IT システムの各種リソース名と文字列比較し、その出現回数をカウントすることで、発生確率を算出する。

この CIM を使った学習とログデータを使った学習におけるリソース Ra とリソース Rb の接続関係の発生確率 $P(Rb|Ra)$ を式 (5) に示す。式 (5) に示すように CIM の静的な接続関係により算出される発生確率 $P_{cim}(Rb|Ra)$ と、ログデータ中の発生頻度、つまりは動的な接続関係により算出される発生確率 $P_{log}(Rb|Ra)$ の積を求め各リソース間の発生確率とする。また静的な接続関係と動的な接続関係について、重み付けを行うべく、それぞれの発生確率に W_1, W_2 を掛け合わせる。なお、いずれかの発生確率が 0 であった場合、 $P(Rb|Ra)$ が 0 となるケースを防ぐため、0 を除く任意の固定値 A を加算する。

$$P(Rb|Ra) = W_1(P_{cim}(Rb|Ra)+A) \times W_2(P_{log}(Rb|Ra)+A) \quad (5)$$

4. 評価

4.1 試作プログラムと評価環境

図 7 と表 1 に本提案方式の評価プログラム及び評価環境を示す。なお、評価システムの開発にあたっては、システム構成を推定するシステム構成推定エンジンを実行する前に、学習エンジンを使い 3.3 節に示す方式で発生確率を算出する構成とした。これにより、時間のかかる学習処理と構成情報の推論処理を分離し、推論処理の処理遅延を防ぐ。

4.2 評価結果

評価では、学習データの効果を測るため、CIM のみ学習に使った場合、ログデータのみ学習に使った場合、CIM とログデータの両方を学習に使った場合、のそれぞれについて Port1 から Port4 までの経路推定について試作プログラムを使い評価を行った。なお、表 2、表 3、表 4 に示す発生確率は、式 (5) における A の値を削除した値とする。図 7

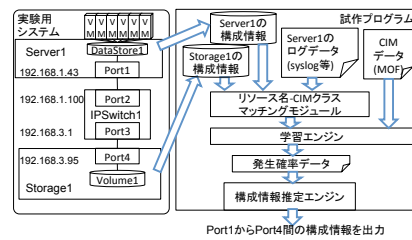


図 7 評価プログラムと評価環境
Fig. 7 The System and Program for Evaluation

表 1 開発・評価環境

Table 1 The Environment for Evaluation

サーバ	Dell OptiPlex (CPU: Intel Corei7 3.4GHz, Memory16G)
仮想環境	VMware ESXi5
VM	CentOS 6 x 6 台 (仮想ディスク数:1Disk/VM)
ストレージ	BUFFALO TeraStation IS TS-IGL/R5
IP スイッチ	BUFFALO BHR-4GRV
サーバ・ストレージ間のネットワークプロトコル	iSCSI
開発環境	MacBook Air(Intel Corei5, Memory 4G) Mac OS 10.8.2, Java6
構成情報 DB	MySQL5.5.2
学習用 CIM データ	CIM Schema 2.35.0 の MOF ファイル
学習用ログデータ	仮想環境の/var/log 配下のログファイル (2ヶ月分) (syslog.log, vmkernel.log 他 19 ファイル)
学習エンジンの実行マシン	SGI Altix UV1000(Intel Xeon 2.66GHz, 1536Core 中 32Core 使用),Memory12TB)
構成情報推定エンジンの実行マシン	開発環境と同一

に示す評価環境における取り得る経路のパターンを図 8 に示す。

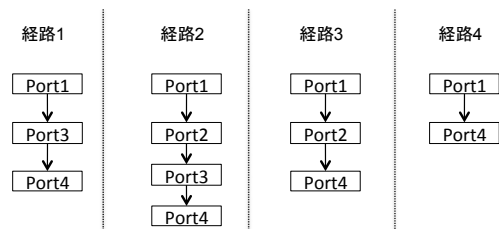


図 8 経路パターン
Fig. 8 Route Pattern

表 2 に CIM データのみ使い学習した場合の Server1 の Port1 から Storage1 の Port4 までの経路毎の発生確率を、表 3 にログデータのみ使い学習した場合の発生確率、表 4 に CIM とログデータを使い学習した場合の発生確率をそれぞれ示す。ログデータの学習においては、予め分かっている Server1, Storage1 のリソース情報と前方一致による文字列検索を行い、出現回数をカウントし発生確率を求めた。なお、本評価では、CIM とログデータの重み付けの値である式 (5) の $W_1 : W_2$ は、1 : 2 として評価実験を行った。

5. 考察

5.1 提案方式の有効性

本提案方式では、構成情報が取得出来ない IT システム

表 2 評価結果 (CIM)

Table 2 Evaluation Result (CIM)

経路	発生確率
経路 1	28.2 %
経路 2	37.5 %
経路 3	28.2 %
経路 4	16.0 %

表 3 評価結果 (ログデータ)

Table 3 Evaluation Result (Log Data)

経路	発生確率
経路 1	0%
経路 2	82.8%
経路 3	82.8%
経路 4	17.2 %

表 4 評価結果 (CIM とログデータ)

Table 4 Evaluation Result (CIM and Log Data)

経路	発生確率
経路 1	0 %
経路 2	62.1 %
経路 3	46.7%
経路 4	5.5%

向けにベイズ推論を使った構成情報の推定を行った。4.2 節の表 4 の結果と実環境の IT システムを比較すると、実環境の構成である経路 2 が最も高い発生確率となった。

CIM のみを学習に利用した場合の傾向として、経路 1,3 は同じ確率となり経路 2、経路 4 は異なる結果となった。経路 1,3 については、CIM のモデルとしては、サーバとスイッチ、スイッチとストレージと同じ接続にて表現されるモデルであったため同じ確率となった。経路 4 については、経路 1,2,3 はサーバとストレージが、スイッチを通して接続されるモデルであったが、経路 4 はサーバとストレージがダイレクトに接続されるモデルであるため、関連の数が異なり発生確率に違いが生じた。また、CIM では、接続関係を示すモデルだけでなくポート速度や CPU 周波数など機器の能力を表現するクラスへの関連など、構成情報に関係のない関連も数多く存在する。そのため、CIM を用いた学習では、1%未満の発生確率の構成が数多く算出される結果となった。

ログデータのみ学習に利用した場合の傾向として、経路 4 と比較し、経路 2,3 の発生確率が高い結果であった。また、経路 1 については、ログデータ中に Port3 に関する出力が存在しなかったため、発生確率が 0 となる結果となった。これは、今回評価に利用した学習データが Server1 が出力するログデータであったため、ログ内容に偏りが生じたためである。表 5 に今回利用したログデータに出現する Port1 に関連したリソース情報の出現回数 (トップ 5) を示す。今回評価に利用した Server1 のログでは、192.168.1.0 のネットワーク内の通信に関する情報が多く、IP Switch1 の Port2 に付けられた IP アドレスである 192.168.1.100

については、382 回出現した。続いて、iSCSI 接続された Storage1 への Login メッセージが多数あった。そのため、ログデータのみ利用した学習では、サーバとストレージがダイレクトに接続されている経路 4 に関しても 17.2%の発生確率となっている。今回の評価では、Server1 のログデータのみを学習データに用い評価を行った。そのため、同じメッセージが繰り返し出力されている傾向があり、偏りが生じている。ログデータに関しては、単一のログデータに限定してしまうと内容に偏りが生じてしまうため、サーバだけでなくスイッチやストレージなど様々な機器のログデータを学習データの対象とすることで、発生確率の偏りを防げると考える。

表 5 Port1 に関連したリソース情報出現回数 (トップ 5)

Table 5 Number of Resource Information Related to Port1 in Log Data(Top5)

リソース関連情報	出現回数
192.168.1.* (192.168.1.43 を除く)	3762
192.168.1.100	382
Volume1	313
Storage1	313
192.168.3.95	237

CIM とログデータを共に学習に利用した場合の傾向としては、CIM のみ、ログデータのみ学習に利用した場合に比べ、実際の評価環境の構成情報と同じ構成の経路 2 が他経路と比べて、より高い発生確率であった。CIM の学習では、経路 2> 経路 1、経路 3> 経路 4 の順で推定され、ログデータの学習では、経路 2,3> 経路 4> 経路 1 の順で推定された。CIM の学習のみでは、経路 1 と経路 3 が同じ発生確率であったが、ログデータの発生確率を加え算出することで、経路 1 が経路 3 よりも低い発生頻度となった。一方、ログデータのみでは経路 2 と経路 3 が同じ発生確率であったが、CIM の発生確率を加え算出することで、経路 3 が経路 2 より低い発生頻度となった。これにより、実際の評価環境の構成である経路 2 が最も高い発生確率で推定することが出来た。このように、CIM のみ学習に利用した場合、ログデータのみ学習に利用した場合では、同じ発生確率になる経路が推定されるケースがあるが、本提案手法では、これらを組み合わせる事で、同じ発生確率となるケースを少なくし、より精度の高い推定が可能となる。なお、本評価ではそれぞれの重み付けに関し、静的な値である CIM の学習による発生確率を 1、動的な値であるログデータの学習による発生確率を 2 の比率とした。これは、静的なモデルよりも、実際に動作している状態を示すログデータを重要視したためである。これにより、CIM による発生確率では各経路の発生確率の差分は小さい傾向であったが、ログデータにより重きを置くことで、CIM だけの学習に比べ表 4 に示すように各経路の発生確率の差を明確にした。これら結果より、静的な情報である CIM による学

習と、動的な情報であるログデータによる学習を組み合わせる本提案手法は有効である。

5.2 自律運用管理の必要性と提案方式の重要性

1章に述べたように、多くの管理者は、サービスレベルの維持に多大なる時間を費やしている。これの解決に向け、従来のサービスレベルの維持のための管理作業の負荷軽減を目指した運用管理の自律化が有望である。

運用管理の自律化の実現に向けたアプローチとして、ベンダーにてITシステムの構成を固定化し提供する垂直統合システムがある。垂直統合システムの代表的な製品としてIBM社のIBM Pure Systems、Oracle社のExadata Database Machineなどがある[10]。図9に垂直統合システム向けの管理ソフトウェアについてシステム構成を示す。垂直統合システムは、1つのラックにサーバ、ネットワーク、ストレージを搭載し、構成を固定化させ提供する。これにより、標準仕様にこだわらずベンダー独自の方法を駆使し、構成情報の一元管理している。この一元管理された構成情報を使うことで、予め管理者が、障害などに応じた対処方法を運用パターンとしてスクリプト化しておき、障害通知などのイベントに応じたスクリプトの自動実行を実現している。このような運用パターンを予め複数用意しておくことで一部の限られたの運用ケースに限定されるが、自律運用が可能となり管理者の負荷を軽減出来る。

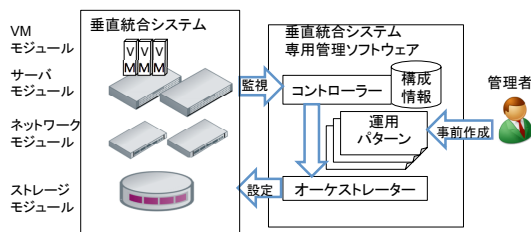


図9 垂直統合システムの管理

Fig. 9 Management System of Integrated System

一方、垂直統合システムは、DB運用に適したものやVM運用に適したものなど、特徴も異なっている。そのため、異なるベンダーの垂直統合システムが並ぶ環境では、DB部にはDB運用に適した垂直統合システムを用い、VM部にはVM運用に適した垂直統合システムを用いるなど、複数の垂直統合システムを跨りサービスを組まれることがある。しかし、ベンダー独自の手法にて構成情報の一元管理を行うことが多い垂直統合システムでは、管理アーキテクチャの統一化が難しく、垂直統合システムを跨った構成情報の一元管理が困難である。これにより、複数の垂直統合システムを跨ったサービスの管理が出来なくなるだけでなく、他の垂直統合システムへの影響範囲が把握出来ないことから、管理者が安心して自律運用管理技術を使うことが出来ない。

そこで、本提案方式を用い垂直統合システム間の構成情報を推定し、他の垂直統合システムに影響のあるリソースを特定することで、サービスの質の低下を招くようなリソースにはスクリプトの自動実行を抑制し、影響のないリソースにのみ自動実行を許可するといった制御が可能となる。このように、垂直統合システムの管理アーキテクチャによらず他システムとの構成情報を一元管理を実現することで、垂直統合システム内のどのリソースに対し自律運用管理技術を適用するか制御可能となり、管理者が安心して自律運用管理技術を利用出来るようになると思う。

6. おわりに

本論文では、静的な構成モデルであるCIMデータと動的な情報であるログデータを組み合わせ学習させることを特徴としたベイズ推論にて、構成情報を推定する方式を提案し、その有効性と今後の適用について考察を述べた。しかし、本論文では、一定の有効性を示したものの限定的な環境下での評価に留まっている。特に学習に用いるログデータに関しては、推定の確度に大きく影響するため、ログデータの種類やその収集方式が今後の課題である。

参考文献

- [1] 工藤 裕, 森村 知弘, 菅内 公徳, 増石 哲也, 廣田 憲久: 障害原因解析のためのルール構築方法及び解析実行方式, 電気学会論文誌.C, 電子・情報・システム部門誌 132(10), pp.1689-1697, 2012
- [2] 敷田 幹文: 大規模サーバ間の部品依存関係に基づく障害通知方式の提案, 情報処理学会論文誌, Vol49. No3, pp.1185-1193, 2008.3.15
- [3] 坂下 幸徳, 工藤 裕, 名倉 正剛, 草間 隆人: 大規模クラウドデータセンターの運用管理コスト削減を可能とするITリソース管理技術. 日立評論, 2012, pp.54-57, 2012.4.
- [4] 坂下 幸徳, 三神 京子, 金子 聡, 敷田 幹文: 仮想環境向け自動データ適正配置方式の提案, 情報処理学会論文誌, Vol54, No3, 2013.
- [5] 江丸 裕裕, 高井 昌彰: 仮想ボリュームクラスタリング法による動的階層制御ストレージの性能管理. 情報処理学会論文誌, Vol.52, No.7, pp.2234-2244, 2011.
- [6] Shinji Kikuchi and Yasuhide Matsumoto. Performance Modeling of Concurrent Live Migration Operations in Cloud Computing System using PRISM Probabilistic Model Checker. 2011 IEEE 4th International Conference on Cloud Computing. pp.49-56, 2011.
- [7] DMTF: Systems Management Architecture for Server Hardware. 入手先 (<http://www.dmtf.org/standards/smash/>)
- [8] SNIA: Storage Management Initiative-Specification. 入手先 (http://www.snia.org/forums/smi/tech_programs/smi_home/)
- [9] DMTF: Common Information Model. 入手先 (<http://www.dmtf.org/standards/cim/>)
- [10] 日経コンピュータ, 特集 SIと運用が消える, 日経BP社 (2012.6.7)