Vol.2010-DPS-144 No.11
2010/9/18

知識型ネットワーク管理支援システムの構成法

笹井　一人[†1]　　北形　元[†1]　　木下　哲男[†1]

管理者の経験的知識や発見的な問題解決法を知識として利用可能とし、ネットワーク管理者の作業、ネットワーク運用・管理を知的に支援する事を可能とするネットワーク管理支援システムについて、その知識構成法、利用法等を提案し、その評価について述べる。

Constructing Method of Knowledge based Network Management Support System

Kazuto Sasai,[†1] Gen Kitagata[†1]
and Tetsuo Kinoshita[†1]

The constructing and utilizing methods of knowledge in network management support system are proposed in this paper. It applies network administrator's experiences and heuristics, as management knowledge, to intelligent support for treatment of network infrastructures. The proposals are evaluated by some experiments using an experimental network system.

1. はじめに

情報社会のユビキタス化、サービス提供の簡易化が進む近年において、それらを支える情報インフラの管理は、その対象が大小様々におよぶため、プロフェッショナル充足が困難であるという前提において、重要な問題となってきている。例えば、ホームネットワークや研究室内、部署内、モバイルネットワークにおけるレベルまでプロフェッショナルをおく事が不可能であり、これらの管理はコンシューマレベルのユーザが兼任せざるを得ない。このような状況においては、種々のネットワークインフラストラクチャが自らその管理を代行し、ユーザの管理負担を大幅に軽減することが強く要求される。ネットワーク管理の自動化を実現するネットワーク管理システム (NMS) について多くの研究がなされている (代表的な物として[1])。いくつかの NMS は実際に実装され、特定のネットワークインフラストラクチャソリューションで実用されている。典型的なアプローチは、集中型のポーリングおよび、静的かつ高度な計算資源を要する管理方式を採用している為、ネットワークシステムの発展めまぐるしい今後の動的ネットワーク環境上におけるネットワーク管理システムは、柔軟かつ適応的で、知的、そしてネットワークリソースの負荷を増大させない事が必要となる。ネットワークおよびサービスのインフラストラクチャにとって、障害は不可避であるが故に、スピーディな障害の発見およびその対処が、ネットワークをより強くそしてロバストに保ち、より多くの信頼できる選択肢を提供することができる[2),3)]。

上記を解決する為に、我々は知的かつ適応的で、自律的なネットワーク管理支援のパラダイムとして、能動的情報資源に基づくネットワーク管理システム (AIR-NMS) を提案してきた[4),5)]。能動的情報資源 (Active Information Resource, AIR) はある情報資源に、その利用を支援する知識と機能を付加した物であり、これらによって他の AIR と連携したり、自らの情報資源のメンテナンスを可能とする[6)]。本研究では、AIR を用いた形式化を導入する事で様々な情報資源、例えば人間の記憶や知識ベースを NMS と組み合わせて利用する事を可能し、ネットワーク管理者の知識横断的な作業負担を軽減する事を目的とする。

本稿では、現在構成されている AIR-NMS の試作システムにおける AIR の構成例等について説明し、それらを統合したシステムの動作検証を行う事で、システムの拡張性や性質について考察する事を目標とする。次章では、AIR のネットワーク管理システムおよび、障害診断システムへの適用例を説明する。また、3 章においてはこれらの相互作用形式、連携方式などについて説明し、それらを試作・実行した動作例について述べる。4 章では、まとめと今後の課題について述べる。

2. 能動的情報資源の構成例

情報ネットワーク管理における AIR は、ネットワーク上の状態情報を扱う I-AIR(status Information AIR)、そしてネットワーク管理者の経験的知識や発見的問題解決を扱う K-AIR(management Knowledge AIR) の二つに大別される。図 1 に AIR-NMS の概念図を示す。I-AIR は情報資源の性質によって静的情報資源と動的情報資源の 2 つに分類される。前者は、IP アドレスや MAC アドレス、ホスト名、ドメイン名、IP ルーティング等の情報

†1 東北大学電気通信研究所
Research Institute of Electrical Communication, Tohoku University

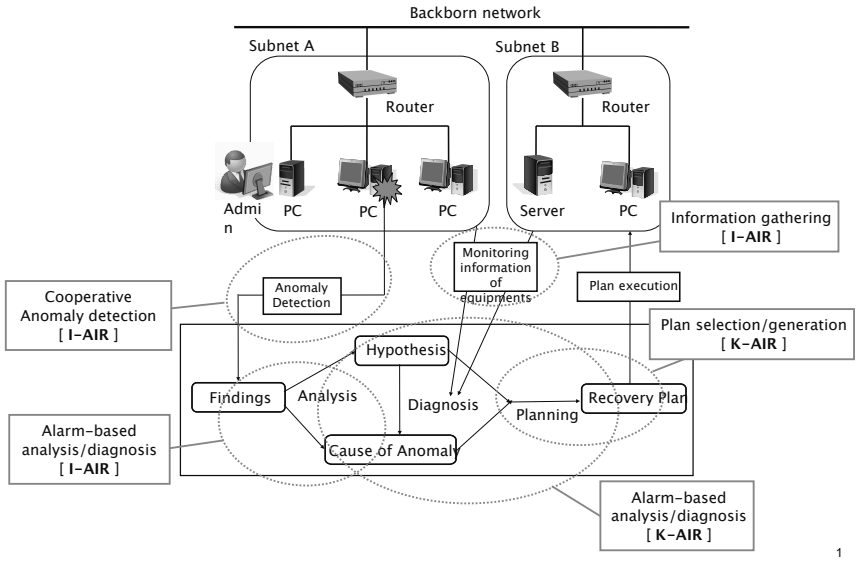

図 1 AIR-NMS の試作システム概観
Fig. 1 An overview of prototype AIR-NMS.

が格納され、後者には流通パケット情報や SNMP-MIB、サービスのログに関する情報が格納される。I-AIR はネットワークの稼働状態を監視、重大な状態はアラームとして通知する他、ネットワーク管理者からの要求に応えて、情報を分析・加工して提供する。K-AIR は管理者の専門的な知識や経験を、ネットワーク管理業務の知識として一般化したものである。これら二種類の AIR が適切に相互作用する事により、与えられた、または発見された問題を扱うことができるようになる。本章では、これらの AIR の例とその構成法について論じる。

2.1 I-AIR

あるネットワークシステムに付随する個々の管理ツールは周期的な収集によって状態情報を収集・統合し、これを用いて機器の状態を判定する。I-AIR は、これら独立したツールからは判定不可能なネットワーク障害の発見、さらに障害に関連した情報収集など、管理者視点のネットワーク状態把握について、これを部分的に代行する事で支援を行う。具体的な例として、実際に実装された I-AIR を**表 1** に示した。各 AIR の名前は、その AIR が実現する機能を表現している。

I-AIR として構成される情報資源として、平文テキストと RDF/XML の二種類が提供されている。例えば、ログ情報は Syslog(UNIX や LINUX における標準的なログ機能) のように平文で与えられ、I-AIR はこれをログのタイプによって分類し、RDF/XML 形式へと展開する。一方、I-AIR は情報資源に関する知識を合わせ持つ事で、収集された情報の取り扱いを機能的に行う事ができる。I-AIR において表現される、基本的な知識は以下の通りである。

- AIR identification Knowledge (ID) — I-AIR の識別子、タスク番号等
- knowledge about Information Resource (IR) — 情報資源のタイプ、更新周期、形式等
- knowledge about periodic investigation process Control Method (CM) — 情報取得プロセスの更新方法を制御する為の知識
- knowledge about Cooperation Protocol (CP) — 他の AIR と協調する為のプロトコルに関する知識

ID, IR, CP といった I-AIR に含まれる知識は、主に情報資源上での操作 i-AIR 間の協調や連携を行う為に必要とされるものである。**図 2** は、上記を含む I-AIR に記述された知識例である。I-AIR の特徴は、通常のネットワーク操作を阻害する障害を調査・分析を FI や CM を通して支援することができるという点である。

上で述べた I-AIR は主に動的な、時間変化する状態を扱っていたが、他の種類の情報として、ネットワーク機器の静的な情報を取り扱う I-AIR の設計について述べる。静的な情報は無時間的な、例えば IP アドレスやサービス構成、デバイス情報などの情報を指す。これらはネットワーク管理者や他の AIR がネットワーク機器の特性や特徴を把握為に必要となる情報であり、この種の AIR はそれらのタスクを支援する機能を保持している。**図 2** に XML 形式を用いて設計した静的 I-AIR の例を示す。静的 I-AIR はネットワーク管理者や他の AIR からの情報提示要求に対して、これに発火した場合、保持する情報の中から、要求された情報に関連する情報を追加して返信する。静的 I-AIR の中で情報提示要求のメッセージが回覧される事で、十分な量のネットワーク機器に関する情報が収集可能となる。

2.2 K-AIR

プロトタイプシステムとして、K-AIR は主に障害解決に関する知識を基に設計した。障害発見後の解決プロセスは次のようにモデル化される：

(1) Cause assuming — 発見された症状から想定される原因を仮定する。
(2) Cause diagnosing — 仮説を検証する為の診断を行い、障害原因の確定を行う。

表 1 実装された I-AIR の例
Table 1 Examples of implemented I-AIRs.

No.	Name (Role)
1	Network Disconnection detector
2	NIC configuration failure detector
3	SPAM mail detector
4	MSBlaster attack detector
5	Mail send/receive error detector
6	TCP/IP stack failure checker
7	NIC configuration failure checker
8	HUB failure checker
9	Router failure checker
11	DNS server process checker
12	SMTP server process checker
13	POP server process checker
14	DNS connection checker
15	Network route to host checker
16	Kernel information checker
17	Lease IP address checker
18	Mail server error checker
19	Number of SPAM mail

I-AIR-No.15

```
(ID   :air id          "i-air@w1:pcB1.example.com"
      :workplace id    "w1:pcB1.example.com"
      :task id         "0123456789" )

(IR   :info type       "ping_result"
      :path            "stdout"
      :format type     "text"
      :time            "2010/01/01/00:11:22" )

(FI   :failure name    "NIC_problem"
      :check name      "Ping_NIC_pcB"
      :check string    ("100% packet loss"
                       "Destination net unreachable"
                       "TTL expired in transit"
                       "Ping request could not find"
                       "unknown host")
      :check info      (CI:exit "yes") )

(CM   :method name     "ping"
      :arguments       ("-c"
                       "4"
                       "172.17.1.2" )
      :trigger info    (TI:interval "60000") )

(CP   :protocol        "Inform – failure Protocol"
                       "Report Protocol" )
```

図 2 I-AIR の知識記述例 (No. 15)
Fig. 2 Knowledge example of I-AIR (No.15).

Static I-AIR

```
<? Xml version="1.0"encording="Shift_JIS"?>
 <subnet>
  <subnetName>Subnet A</subnetName>          <domain>example.jp</domain>
  <addrspace>172.20.2.0/24</addrspace>       <gateway>172.20.2.1</gateway>
  <firewall>active</firewall>                <adminName>Mr. Noname</adminName>
  <adminMail>noname@example.jp</adminMail>
    <server>
      <service>SMTP</service>                <ipaddress>172.20.0.2</ipaddress>
      <name>smtp.a_lab.example.jp</name>     <process>Postfix2.1</process>
    </server>
 </subnet>
```

表 2 静的情報資源の記述例
Table 2 Example of static I-AIR.

(3) Measure planning — 確定された障害原因から解決策を策定する。

上記３種のプロセスに内包される知識要素として、**Symptom**(症状)、**Causes**(障害原因)、**Diagnosis methods**(原因診断手法)、**Diagnosis reports**(診断結果の提示法)、**Measures**(障害解決策) の５種類の知識要素を選択する。本研究では、K-AIR の設計として、モデル化された障害解決方法に従って、記述された５種類の知識を合成し３種類の知識資源を構成する。

- $\mathbf{K}_{\mathrm{SC}}$ **(Symptom, Cause)** — 発見された症状から障害原因の推定を行う操作に関する知識。主に、症状と想定される障害原因との組によって表現される。
- $\mathbf{K}_{\mathrm{CD}}$ **(Cause, Diagnosis method, Diagnosis report)** — 推定された原因を確定する為の診断方法に関する知識。障害原因についての仮説と、その成立条件、検証方法の組として表現される。
- $\mathbf{K}_{\mathrm{CM}}$ **(Cause, Measure)** — 確定された障害原因を解消する為の実践的対策案の策定に関する知識。障害原因と対策案の組によって表現される。

図 3 に記述された K-AIR の例を示す。構成された知識を情報資源化する形式として、我々は XML と XPath を用いた。$\mathrm{K_{SC}}$ において、Symptom(障害症状) は属性'symptom' として表現され、ここでは属性値として ("unable to send mail") が格納されている。Cause(障害原因) は要素 <cause> として表現され、想定される原因として ("unable to name resolve", "network connection failure", "over sendable size limit") が <cause> の内容として指定されている。

$\mathrm{K_{CD}}$ において、Cause は属性'cause' として表現され、その値として例では ("over sendable size limit") となっている。Diagnosis method(診断方法) は、要素 <dm> として表現され、内容として診断の為のコマンド <p> の系列が記述されている。ここでは、最初の <p> として、"request #//sent mail size# from #source#"が記述され、これによってクライアントの I-AIR から送信メールサイズが取得される。また３段目の <p> では、"true(#//sendable size limit# -lt #sent mail size#によって、送信可能メールサイズと送信されたメールのサイズが比較される。ここで、'#...#' は変数であり、I-AIR や他の K-AIR から (KUS, FUS) を用いて取得された値が格納される。例では、'#source#' にはメールクライアントの I-AIR の ID が代入され、'#sent mail size#' には'#source#' より取得された送信されたメールのサイズが代入される。Diagnosis report(診断結果) は要素 <dr> として表現され、その内容として診断報告テンプレートが記述される。診断報告テンプレートは診断方法と同様に変数を含んでおり、診断プロセスにおいて取得された情報

が代入され、診断結果として提示される。

K_{CM} においては、Measure(障害解決策) が要素 <m> として表現されており、解決策テンプレートがその内容として記述されている。解決策テンプレートもまた変数を含んでおり、さらにそれに加えて、情報要求先 (図 3(1))、および条件付き文 (図 3(2)) が記述可能となっている。

上記で示した表現法を用いる事で、知識要素の集合は多様な診断方法や具体的な診断報告、実践的な解決策を表現する事ができる。これによって、分類・合成された知識資源は再利用可能なものとなり、AIR-NMS における知識追加、導入における作業負担を大幅に減少することができる。

3. 協調動作の設計

本章では、AIR-NMS における AIR 間の協調動作について、その設計例を基に説明する。協調動作として、柔軟性や適応性を実現する、ネットワーク機器情報 (I-AIR) とネットワーク管理知識 (K-AIR) 間の相互作用形式を設計する。

3.1 AIR 間の相互作用

前章で説明した K-AIR はネットワーク管理者に完全な障害解決方法を与えるが、これを得る為には、分類された知識部品 (K_{SC}, K_{CD}, K_{CM}) が適切に選択され、そして合成される必要がある。仮にこれらのプロセスがネットワーク管理者に取って信頼性の低い物であれば、これらの扱いは管理者にとって有用な物ではなく、逆に大きな負担を強いることとなる。これを解決する為に、各知識部品をそれぞれ自律動作可能な K-AIR とすることで、それらが自ら組織化し、知識を合成することで、障害の解決法を作成し、個々の状況においてそれを持ちいることを可能とする。

図 4 は K-AIR の組織化形式を表す。ここで、(K_{SC}, K_{CD}, K_{CM}) が K-AIR として構造化されたものを (K_{SC}-AIR, K_{CD}-AIR, K_{CM}-AIR) と記述している。K-AIR の各タイプはワークプレース (AIR の動作環境) 上で活性化され、これらの自律的な組織化 (すなわち協調動作) を可能とする為のメッセージ表現形式、メッセージ交換形式を設計される。

図 5 は K-AIR 間のメッセージ表現形式の模式図である。図中で示されているように、ここでは 3 種類のメッセージ (Msg-S, Msg-C, Msg-I) が設計されている。各メッセージにおける表現の詳細は以下の通りである。

- **Msg-S** — この表現形式は障害診断を AIR-NMS に依頼する際に用いられるメッセージ形式であり、管理者によるインタフェースからの入力または、I-AIR からの障害報告

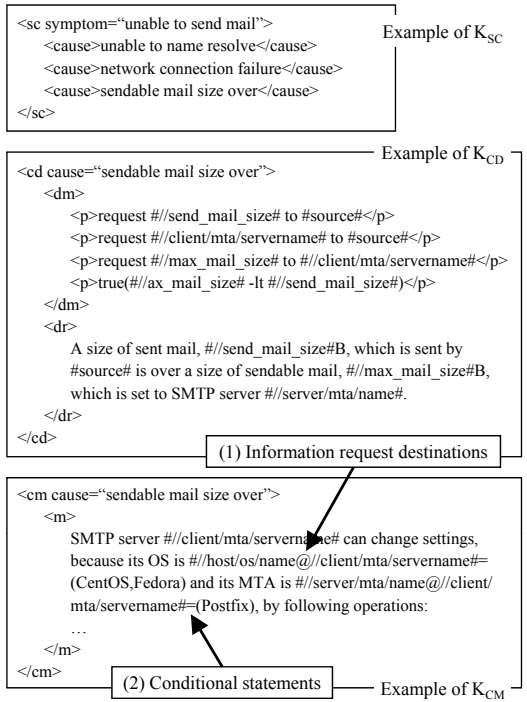

図 3 K-AIR の記述例
Fig. 3 Description examaple of AIR-NMS.

に用いられる形式である。

ここで、<**task id**> は障害診断タスクの識別子であり、各タスクにユニークに割り当てられ、K-AIR 間で共有され、タスクの重複実行を防ぐことができる。<**symptom**> は障害および症状を表し、AIR-NMS 駆動命令の中核をなす部分である。<**source**> は障害の観測されたホスト名、ネットワーク名、もしくは IP アドレスを表す。<**detail info**> は管理者によって入力された時点で明らかとなっている、障害解決に役立つ補足情報を表す。

- **Msg-C** — 各想定される原因についての診断依頼を行う為のメッセージ形式であり、主に K_{SC}-AIR から K_{CD}-AIR または他の K_{SC}-AIR へ向けてブロードキャスト送信され

るものである。また、このメッセージ形式は K_{CD}-AIR から K_{CM}-AIR への解決案策定依頼を行う際にも用いられる。

ここで、<**k-air id**> はメッセージ送信元の K-AIR の識別子であり、K-AIR 間もしくは I-AIR とのやりとりにおいて、返信先やその他の宛先として利用される。<**cause**> は、想定される原因に関する記述を表し、原因診断および解決案策定の依頼を行う際の主題となる。<**source**> は、Msg-S より継承される。<**cooperator**> は、障害解決タスクにおいて構成された組織に参加している K-AIR の識別子。<**detail info**> Msg-S から継承される補足情報。

- **Msg-I** — K_{CD}-AIR による原因診断、および K_{CM}-AIR による解決案策定の際に必要とされる情報を I-AIR や他の K-AIR に問い合わせる為の表現形式であり、主にブロードキャスト形式で送信される。

  ここで、<**k-air id**> は Msg-C と同様である。<**request info**> は K_{CD}-AIR もしくは K_{CM}-AIR が必要とする情報の詳細であり、これによって I-AIR 側からの応答を得ることができる。<**destination**> は、要求された情報を持つデバイスの IP アドレスまたはホスト名を特定する為に用いられる。I-AIR が情報要求のメッセージを受け取り、かつ <request info> および <destination> が一致した場合に、I-AIR は要求に応答して、<k-air id> で示されたメッセージ送信元へ向けて情報を返信する。

上述のメッセージ形式を用いる事で、K-AIR はそれらの間の関係性を障害診断タスクに応じて、またネットワーク状態の変化に応じて柔軟に変更しながら障害解決支援を行うことができると考えられる。さらに、上記の協調形式は、そのほとんどのプロセスをブロードキャスト方式によって並列的に実行可能であり、ネットワーク障害解決における推論プロセスを分散環境上、特にマルチエージェント環境において効率的に実行する事が可能となる。

3.2 動作検証および評価

本稿で述べた設計に基づいて実装した試作システムについて説明する。試作システムの実装は、エージェントリポジトリ機構による高い再利用性を実現する分散型エージェントフレームワークである ADIPS/DASH[7] 環境上で行った。試作システムのインタフェースおよび障害解決案を提示した画面のスクリーンショットを**図 6** に示す。ネットワーク管理者が、ネットワークに障害や異常を発見した場合、図 6 右上に示す障害診断要求インタフェースを用いて AIR-NMS にこれを入力する事ができる。障害診断要求の入力は、まず発見された障害症状 (Symptom) および障害発生箇所 (Detected site) を入力し、その後詳細情報 (Deatiled information) として補足情報を適宜入力する事が可能である。入力された情報

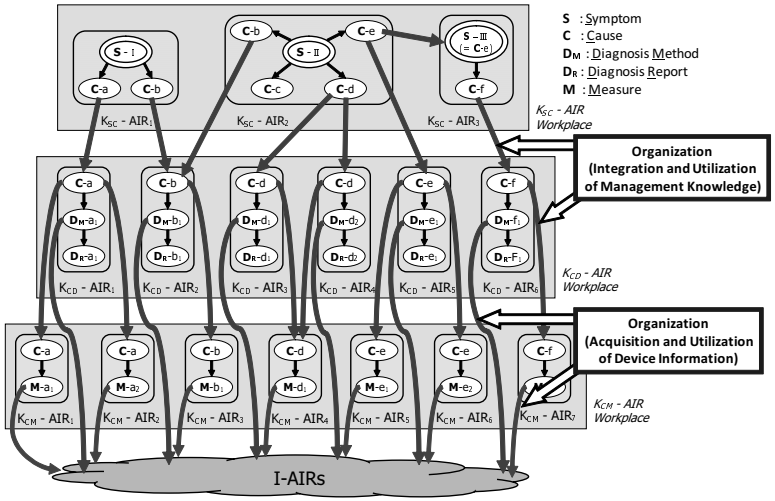

図 4 K-AIR の記述例
Fig. 4 Description examaple of AIR-NMS.

(Symptom, Detected site, Detailed information) は、それぞれ前節で定義された Msg-S 構成要素である (<symptom>, <source>, <detail info>) に埋め込まれて、障害診断要求メッセージとして AIR-NMS へ送信される。これを受け取った AIR-NMS は、診断要求に基づいて知識の組織化を行い、想定原因に関する仮説を作り、それらを検証プロセスによって同定した後、必要な情報を付加して、障害診断報告および、解決策の提示を行う。図 6 左下に示した二つのウィンドウの内、上に示したものが障害診断報告 (Report of diagnosis) であり、下に示したウィンドウに解決策が提示されている。各提示された情報内で、赤字で示された部分がテンプレートの変数部分に追加情報が入力されている箇所である。

図 6 の動作検証において、K-AIR 群として障害症状を示す知識要素 S(Symptom) に'unable to send mail' を用い、想定される原因 C(Cause) を 20 個、原因診断 D(Diagnosis method) を 20 個、そして解決策 M(Measure) を 20 個実装した。また、I-AIR は 2 章で述べた I-AIR および静的情報が記述された I-AIR を同じプラットフォーム上に配置して行った。図 6 で示した通り、今回の動作試験において、AIR-NMS は、I-AIR 上に埋め込まれた情報資源を正確に収集し、診断が行われた事が確認できる。しかしながら、今回の動作実験では、I-AIR 上におかれた情報は、あくまで我々が便宜上作成したものであり、実際にこれ

Vol.2010-DPS-144 No.11
2010/9/18

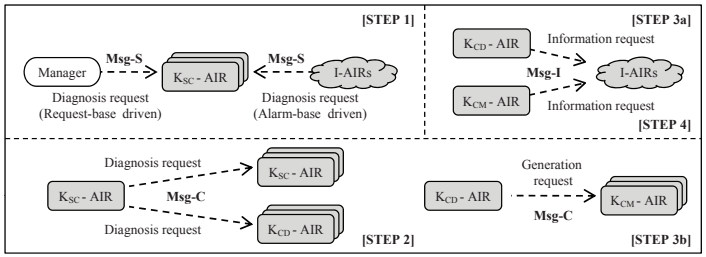

図 5 メッセージ形式の定義
Fig. 5 Definition of messages among AIRs.

を取得したり、更新したりするプロセスを構成し、これを実験ネットワーク上の機器に導入した状態での動作検証は、今後の課題であり、現在実装を進めている。

4. おわりに

本稿では、知識型ネットワーク管理システムの可能性の検証例として、能動的情報資源を用いたネットワーク管理システムについて、その試作システムの設計について延べ、さらに動作検証例として、実装された試作システムの動作について説明した。試作システムの動作例は、ネットワーク機器上の状態情報と、類別・合成されたネットワーク障害解決に関する知識が相互作用し、適切な解決策が提示される事を示した。今後の課題としては、現在便宜上用意している状態に関する情報資源を機器上から取得する方法と融合させる機構を構成し、一貫したシステムとしての動作を実環境に即した形で検証するとともに、その動作例を拡充するつもりである。

参 考 文 献

1) Consens, M. and Hasan, M.: Supporting network management through declaratively specified data visualizations, *Proc. IEEE/IFIP 3rd International Symposium on Integrated Network Management*, pp.725–738 (1993).
2) Martin-Flatin, J. P., Znaty, S. and Hubaux, J. P.: A survey of distributed enterprise network and systems management paradigms, *Network and Systems Management*, Vol.7, No.1, pp.725–738 (1999).

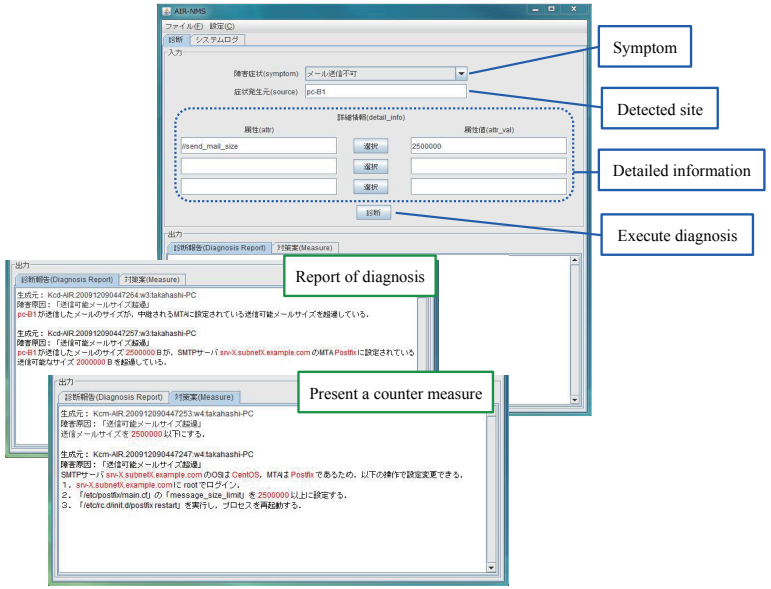

図 6 K-AIR の記述例
Fig. 6 Description examaple of AIR-NMS.

3) Stephan, R., Ray, P. K. and Paramesh, N.: Network management platform based on mobile agent, *International Journal of Network Management*, Vol.14, No.1, pp. 653–659 (2004).
4) Konno, S., Iwaya, Y., Abe, T. and Kinoshita, T.: Design of Network Management Support System based on Active Information Resource, *Proc. 18th Int. Conf. Advanced Information Networking and Applications (AINA 2004)*, pp.102–106 (2004).
5) Konno, S., Sameera, A., Iwaya, Y. and Kinoshita, T.: Effectiveness of Autonomous Network Monitoring Based on Intelligent-Agent-Mediated Status Information, *LNCS (LNAI)*, Vol.4570, pp.1078–1087 (2007).
6) Li, B. and Kinoshita, T.: Active Support for Using Academic Information Resource in Distributed Environment, *Int. J. Computer Science and Network Security*, Vol.7, No.6, pp.69–73 (2007).
7) DASH GROUP, DASH — Distributed Agent System based on Hybrid architecture, [Online]. Available: http://www.agenttown.com/dash/.