

論文

日中環境モニタリングシステムの 構築と効率的運用・保守方法

志方 正樹 富岡 雅樹 藤崎 祐生 諏訪 敬祐

環境モニタリングシステムとは、観測地に設置した環境観測装置をインターネットに接続することにより、自宅や研究室のPCから現地の気象データ等を調査することができるシステムである。中国の北京林業大学(以下現地)では、観測地まで直接赴いてデータを取得する環境観測装置を使用しており、より効率的に観測を行えることが望まれていた。本研究では、現地の環境観測を利便化するための環境モニタリングシステムの導入、そして効率的に運用保守を行うための運用保守体制の見直しを目的とした。無線通信機能付気象観測装置とデータ受信・表示端末(以下コンソール)を使用することにより、現地のシステムの利便化を達成した。また、システムに異常が発生した際にその旨を運用者に通知するプログラムを作成し、運用保守の効率化を図った。これにより、運用者の負担低減および復旧への対応の遅れによるシステムの稼働率低下を抑制できることを明らかにした。

キーワード：日中、環境モニタリングシステム、運用、保守

1 はじめに

環境モニタリングを行なうには多大なコストを要する。特に環境観測装置を観測地に設置し、人が直接設置場所に出赴いて気象データを回収する必要がある場合、頻繁に気象データの回収へ向かう必要があるだけでなく、観測装置に異常が発生した際においても、観測装置自体を調べない限り異常の発生を知る術がないなどの問題点が存在する。本学内においては既に環境モニタリングシステムを構築しており、これらの問題に対応済みであるが、北京林業大学(以下現地)においては依然として未対応のままであり、効率的に観測を行っているとは言い難い。そのため、本学内で培った経験をもとに同様のシステムを導入することで、現地の環境観測の利便性向上を図るとともに、取得した気象データを双方の大学で閲覧可能にすることにより、気象データの活用に寄与する。

一方で本システムを新たに構築するにあたり、運用保守の負担が増すことによる環境モニタリングシステムの稼働率低下が予想される。稼働率の低下は利用者が望むデータを提供できなくなる可能性がある。また、今後も

新たなシステム構築を行う度に運用者の負担増加が予想されるため、負担をできるだけ軽減する必要がある。さらに、長期の運用においてシステム運用者が交代しても安定したシステム運用が求められる。

以上のことから、新規のシステムの構築を行なう際には、運用保守体制の見直しを行い、運用者の負担低減及びシステムの稼働率向上を図ることが重要である。運用保守体制見直し後の運用保守の流れは図1の通りとなる。

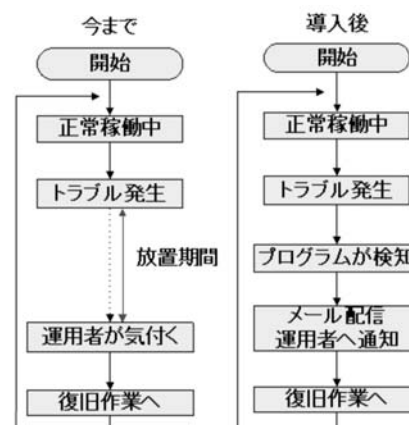


図1 運用保守の流れ

システム運用者の交代による連絡の不備を防止するために、トラブルが発生した際は、過去に同一の事例が存在していないかどうかを調べた上で、実際に対応した事例が存在した場合については、運用者にメールにより通知する。これにより、交代以前と変わらない運用保守体制を実現する。また、新たな事例に対応した場合は、問題解決後に復旧作業の内容を記録しておくことにより、次

SHIKATA Masaki

武蔵工業大学環境情報学部情報メディア学科 2006 年度卒業生

TOMIOKA Masaki

武蔵工業大学環境情報学部情報メディア学科 2006 年度卒業生

FUJISAKI Yusei

武蔵工業大学環境情報学部情報メディア学科 2006 年度卒業生

SUWA Keisuke

武蔵工業大学環境情報学部情報メディア学科教授

回以降に同一のトラブルが発生した際においても、早期解決が図れるようになる。

これらのシステムを実装することにより、トラブル発生後から復旧作業開始までの時間を短縮し、システムの稼働率低下を防止する。さらに、運用者の負担低減を行ない、これまでより環境モニタリングシステムの運用・保守を効率的に行なうことを可能にする。

2 環境モニタリングシステム

日中環境モニタリングシステムの全体の構成を図2、3に示す。横浜キャンパス内の環境モニタリングシステムはField Server DataServer に蓄積プログラムをイン

ストールすることにより構築されている。現地においては新たにNC DataServer を設置し、同サーバを中心とした環境モニタリングシステムを構築した。

2.1 気象観測装置のシステム

現地に設置する気象観測装置(DAVIS 社製) を図4に示す。

気象観測装置が取得した気象データは無線によってコンソール(図5)へ送られる。コンソールに送られたデータはRS232-Cケーブルを用いてパーソナルコンピュータへと吸い出す。その際は気象観測装置とセットになっていたソフトウェアを用いる。パーソナルコンピュータに移したデータはFTPを用いることでNC DataServerへ

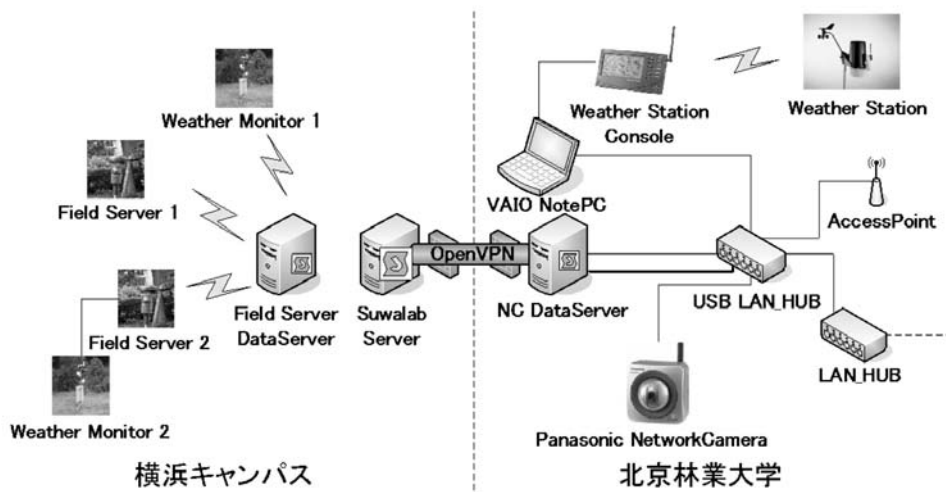


図2 日中環境モニタリングシステムの構成1

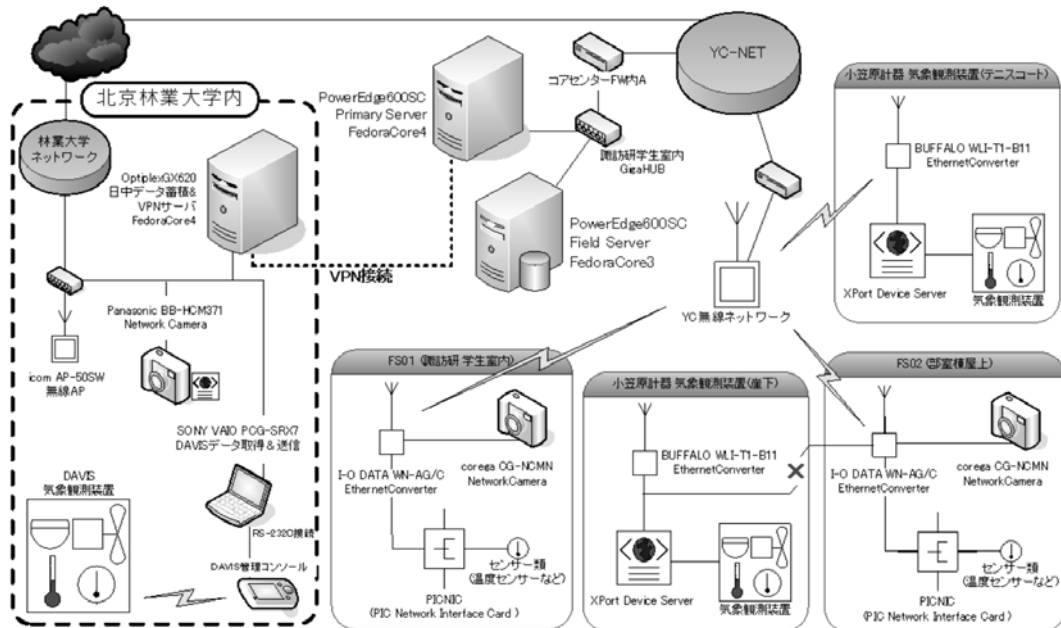


図3 日中環境モニタリングシステムの構成2

アップロードする。アップロードされたファイルは蓄積プログラムにより日時毎に分類、蓄積する。



図4 気象観測装置



図5 コンソール

2. 2 ネットワークカメラのシステム

現地に設置するネットワークカメラ (Panasonic 製) を図6に示す。このカメラはカメラコントロール(パン, チルト), デジタルズーム (10 倍) が可能であり, 画素数は約 32 万画素である。Network Camera が取得した画像データは, NC DataServer 内の蓄積プログラムが一定間隔毎にネットワークカメラに内蔵された表示ページへアクセスして収集・蓄積を行なう。サーバ構成, ソフトウェア構成については次章で詳しく説明する。



図6 ネットワークカメラ

3 日中環境モニタリングシステム概要

実際に日中間で構築した環境モニタリングシステムをサーバネットワーク構成, ソフトウェア構成に分けて記述する。

3. 1 サーバのネットワーク構成

現地における環境モニタリングシステムのサーバネットワーク構成を図7に示す。

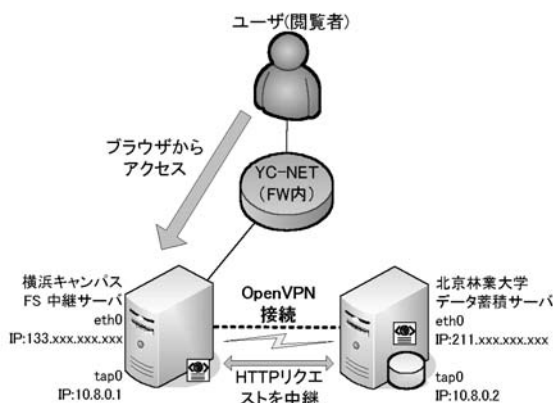


図7 サーバのネットワーク構成

今回のシステムを構築する際の制限事項として, 現地側でグローバル IP アドレスが取得できず, ユーザが現地のサーバへ直接アクセスできない点と, 現地の規約により取得したデータを現地の構内に保存する必要があった点が挙げられる。

これらを解決する工夫として, 現地用データ蓄積サーバの他に, 本学横浜キャンパスに中継用サーバを設置し, 現地用データ蓄積サーバとの間に VPN セッションを常時継続する構成とした。実際にユーザがデータを参照する場合は中継サーバ内のプロキシサーバが現地のデータ蓄積サーバ内の Web ページをユーザへ転送する。

このような構成とすることで, 中継サーバ上で柔軟なアクセス制御を行うことができ, また現地用データ蓄積サーバのメンテナンスをリモートで行うことが可能であるので現地で特別な操作を行う必要がない。また, 今後データ蓄積サーバを別の場所へ新たに設置する場合も, 中継サーバを軸に構成することで比較的容易に拡張することが可能である。

3. 2 中継サーバのソフトウェア構成

横浜キャンパス内の中継サーバは, OS に Linux の Fedora Core4 を使用した。

現地との通信にはソフトウェア VPN である OpenVPN を使用し, 横浜キャンパスの中継サーバを VPN サーバ側, 現地用データ蓄積サーバを VPN クライアント側とし, 現地からの VPN アクセスを横浜キャンパスで受け付けている。この VPN セッションの中での仮想的な IP アドレスを用いてサーバ間の通信を行っている。

HTTP によるアクセスを中継するソフトウェアには Apache を使用し, リバースプロキシのモジュールを使用した。中継サーバ上で現地専用仮想ホスト名を割り当て, ユーザが中継サーバ上の仮想ホスト名にアクセスを行うと, Apache 上のリバースプロキシモジュールが現地用データ蓄積サーバへ HTTP リクエストを送信し, また返ってきたパケットをユーザに中継することでユーザは Web ページのデータを得る。

現在はセキュリティ上の理由から学内からのアクセスのみを Apache 上で許可して運用しているが, 状況に応じてアクセス権の設定を変更することができる。

3. 3 データ蓄積サーバのソフトウェア構成

現地用データ蓄積サーバは, 中継サーバと同じく OS に Linux の Fedora Core4 を使用し, VPN ソフトウェアに OpenVPN, Web サーバソフトウェアに Apache を使用した。

サーバを構築するにあたり, いずれもすべてフリーのソフトウェアを使用している。フリーソフトを利用することで, 設置箇所が増えたり, 利用者が多くなったりすることで, サーバにかかる負荷が大きくなった場合に, サ

ーバの追加がコスト等の面で困難になることを防ぐためである。

そのほか、Web 上にデータ表示させるための Web アプリケーションとして、サーバサイドスクリプトである PHP を用い、表示プログラムを柔軟に作成可能とすることでユーザの細かな要求を実現できるようにした。

また、現地用データ蓄積サーバは起動後自動的にネットワークに接続される設定になっており、また VPN セッションが切断されている場合はすぐにリトライを行う設定となっているため、原則的に現地での操作は不要となるようになっているが、実際にはいくつかのトラブルが発生し、稀にデータの取得や閲覧ができない状態となり、システムとして安定しているとはいえない状況にある。

本研究ではこれらの対策としてトラブルが発生した際にシステムが自動的に修復できない場合、何らかの形でユーザに通知し、稼働率を向上させるプログラムを作成した。実際に気象データを取得するプログラム、データを表示するプログラム、トラブルの際にユーザに通知するプログラムの詳細については次節以降で述べる。

3.4 データ蓄積システムの構成

データの蓄積をシステムにより行うために、現地の環境に即したデータ蓄積プログラムを作成した。実際に作成したデータ蓄積プログラムの流れを図8に示す。なお、プログラムの作成にはOSに依存しないJavaを使用した。

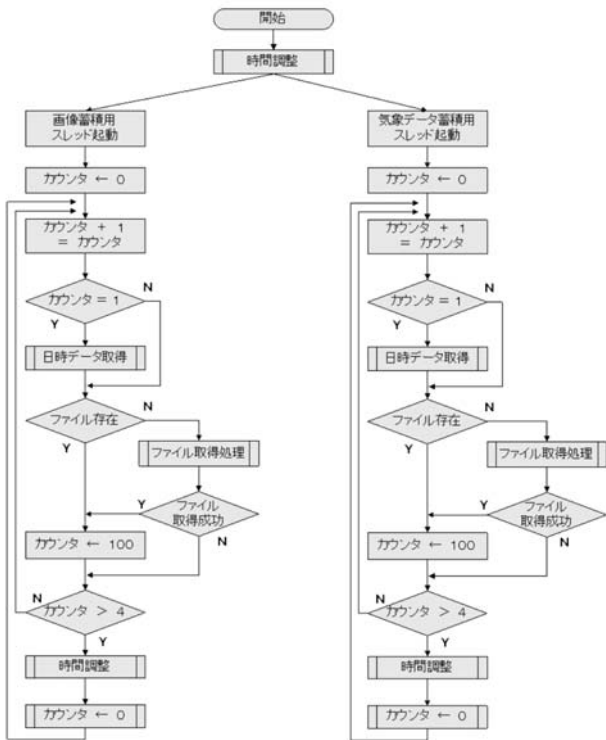


図8 データ蓄積プログラムの処理の流れ

3.5 Web 表示システムの構成

データ取得プログラムにより蓄積した気象データをユーザがより活用しやすくするには Web 上でデータ参照できることが好ましい。しかし、単純な HTML で書かれたファイルでは、データ参照を行うことができないため、Web アプリケーションの作成が必要となる。今回作成した Web 表示アプリケーションは PHP, JavaScript, HTML などの言語で構成されている。日中環境モニタリングシステムで閲覧できるメインページを図9に示す。

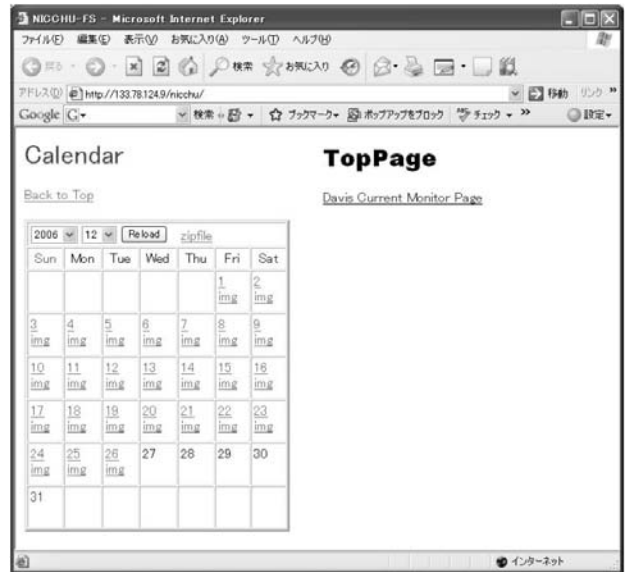


図9 表示メインページ (2006年12月26日現在)

ページはフレームで構成されており、右側のフレームにリアルタイムデータの閲覧が可能なページへのリンクを表示し、左側に年・月・日で日付管理および蓄積されたデータを閲覧するためのカレンダーページが表示されている。リアルタイムデータを閲覧できるページを図10に示す。

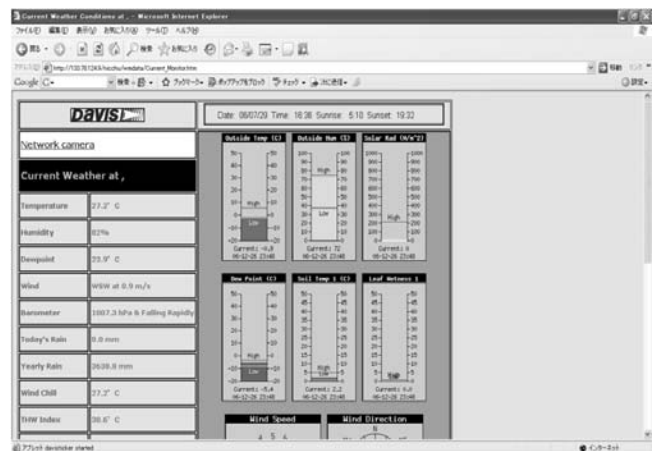


図10 リアルタイムデータ閲覧ページ

このページでは、温度・湿度・雨量・風速など多岐に

わたるデータが表示されており、各データをテキストとイメージの両方で閲覧することができる。ページは10分ごとに自動で更新され、そのつど最新のデータを閲覧できる。日毎の蓄積データを閲覧するページを図11に示す。

Date	Time	Outside Temp	Outside Hum	Solar Rad	Dew point	Soil Temp	Leaf wetness	Wind Speed	Wind Direction	Image
Y/M/D	HM	℃	%	W/m ²	℃	℃	-	m/s	-	画像
06-12-01	00:00	11	53	0	-7.5	2.8	0	09	24	画像
06-12-01	00:10	07	46	0	-8.8	2.8	0	19	53	画像
06-12-01	00:20	02	40	0	-10.9	2.8	0	09	244	画像
06-12-01	00:30	00	40	0	-11.1	2.8	0	19	41	画像
06-12-01	00:40	-02	42	0	-11.6	2.8	0	00	0	画像
06-12-01	00:50	-03	43	0	-11.4	2.8	0	13	57	画像
06-12-01	01:00	-03	40	0	-12.3	2.8	0	09	26	画像
06-12-01	01:10	-03	38	0	-12.9	2.8	0	09	360	画像
06-12-01	01:20	-03	38	0	-12.9	2.8	0	00	360	画像
06-12-01	01:30	-03	38	0	-13.0	2.8	0	00	360	画像
06-12-01	01:40	-03	38	0	-12.6	2.8	0	00	360	画像
06-12-01	01:50	-03	40	0	-12.3	2.8	0	00	360	画像
06-12-01	02:00	-03	40	0	-12.2	2.8	0	00	360	画像
06-12-01	02:10	02	37	0	-12.6	2.8	0	09	26	画像
06-12-01	02:20	04	36	0	-12.9	2.8	0	13	22	画像

図11 気象観測装置から取得したデータの閲覧ページ

このページでは、温度・湿度・雨量・日射量などのデータをテキストとテーブルを用いて表示しており、データ蓄積プログラムにより10分置きに保存された気象データを閲覧することができる。また、月の平均値や積算値などのデータを閲覧することができる。ネットワークカメラの静止画面画像のアニメーションページを図12に示す。



図12 画像のアニメーションページ

このページでは10分ごとに蓄積された静止画像を時間順に自動表示させることができる。これによりネットワークカメラで撮影された当日分の画像をアニメーション表示させながら閲覧することができる。

4 運用保守効率化のためのシステム

日中環境モニタリングシステムを構築したことにより、

新たな運用保守の機会が生まれた。これにより、運用者の負担増加などに伴いシステム全体の稼働率の低下が懸念される。また、現在の運用保守体制では運用者が頻りにシステムの稼働状態を確認する必要があり、効率的に運用保守が行われているとは言い難い。そのため、運用保守を行う際に必要となる作業のうち、ルーチンワークをプログラムにより行うことで、運用保守の効率化を図る。プログラムの流れは図13の通りである。

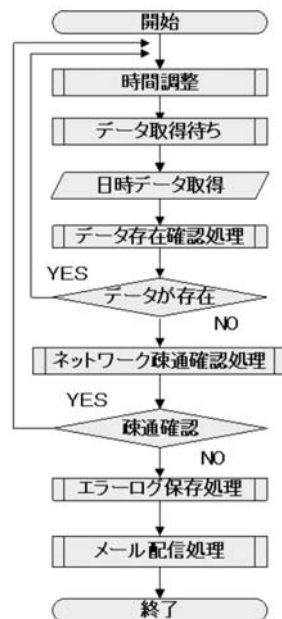


図13 運用保守効率化のためのシステムの処理の流れ

本プログラムの作成にはデータ蓄積プログラム同様Javaを用いる。これらの処理については観測装置ごとにスレッド化を行い、観測装置が増えた際にも対応可能とする。

データ存在確認処理においては、ファイルのタイムスタンプを用いるため、前処理において日時データの取得を行っている。この際に取得した日時データは後のエラーログ保存処理、メール配信処理などにおいても用いる。また、メール配信処理においては、過去の復旧対応内容が記載されているファイルを検索し、存在しない場合はエラー内容のみを、存在した場合は対応方法も本文に記載した上で、指定した送信先メールアドレスに対してメールを送信する。

5 システム検証

実際に設計したシステムをプログラミングし、その有意性について検証する。なお、過去の復旧対応内容を記載したファイルについては、今回は検証を行なうために仮のファイルを作成し、実際に読み込まれた際には、そ

の旨を伝えるようにした。また、運用保守効率化のプログラムの検証にあたり、本キャンパス内の FieldServer01 を停止した状態で実験を行った。

5.1 運用保守効率化のためのシステムの検証

環境モニタリングシステムを効率的に運用保守するためのシステムについて検証する。実際に作成したプログラムを稼働させている状態を図14に示す。



図14 プログラム稼働時のLinux上での表示画面
(セキュリティ上設定情報はぼかしている)

本システムでは最初に対象となる全機器のデータ取得間隔の最小公倍数を軸として、分単位で時間調整が行われている。また、各機器のデータ取得間隔が異なる場合がある、データが確実に取得されたと判断できるまで待つ必要があるなどの理由から、すぐ後に各機器に合わせて時間調整を行なっている。これらの調整が終わると、データの取得確認チェックが行われ、ファイルのタイムスタンプが取得できていない時、あるいはファイルに対してアクセスが行えない時にはデータの存在が確認できなかったことを伝えている。一方、実際にタイムスタンプの確認が行なえたデータに関してはその旨を伝えている。タイムスタンプが確認できなかったファイルに関連

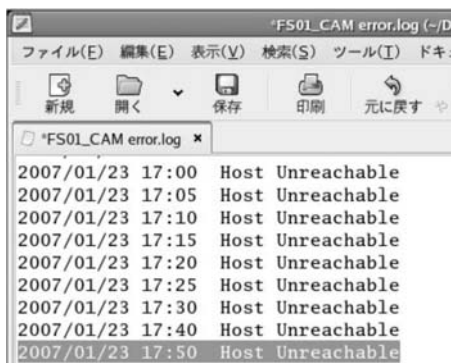


図15 エラーログの保存結果

する機器については、ping処理によるネットワーク疎通確認処理が行なわれている。またpingが通らなかった機器については図15のように、pingの応答結果がログファイルに保存されている。

pingによるネットワーク疎通確認処理においてエラーが発生した機器については、図16に示すように該当機器の過去の復旧対応内容を記載した仮ファイルを読み込んだ上で、エラー内容を記載したメールの配信を指定した送信先アドレスに行なっている。実際に送信されたメールの内容が図17である。



図16 過去の復旧対応内容を記載する仮ファイル



以下のエラーが発生しましたので通知します

133.78.126.13[FS01_PIC] Host Unreachable
FS01_PIC ログファイルを読み込みました
133.78.126.14[FS01_CAM] Host Unreachable
FS01_CAM ログファイルを読み込みました

図17. エラーメールの配信結果

5.2 結果と考察

以上、作成したプログラムに関して検証した結果、データ取得確認処理、機器のネットワーク疎通確認処理、メール配信処理は設計したどおりに処理が行なわれた。これにより、蓄積プログラムのデータ取得や気象観測装置自体が停止した際に、自動でそれを検知して運用者に通知するため、従来より効率的に運用保守を行なうことが可能となり、運用者の負担低減および復旧作業の遅れによるシステムの稼働率低下を軽減することができた。また、今後は復旧作業を行なった際にその内容を確実にファイルに残しておくことで、同一の事例が発生した際は誰でも復旧にあたるのが可能となった。

一方でファイルが存在しているにもかかわらず、タイムスタンプが取得できない時があるなど、現在のプログラムの問題点が浮上した。ただし、機器が稼働している場合、ネットワーク疎通確認処理において応答があるため、大きな問題点とは言えない。しかし、その分処理を行なうサーバや機器に対して余計な負荷がかかってしまうため、改善すべき点である。

6 おわりに

6.1 まとめ

北京林業大学内に新たに環境モニタリングシステムを構築した。これにより、運用保守の負担が増加したが、データの存在確認や機器に対する ping によるネットワーク疎通確認などをシステム化したことにより、運用者の負担軽減という目的が果たせた。また、運用者がトラブルに直ぐに気が付くという点で、トラブル発生期間も短縮でき、稼働率も上がると考えられる。また、今後においてシステムの規模が増大し、トラブルの数が増えた際においても、復旧対応内容を記録しておくことで、運用保守方法の伝え忘れを防ぐことが可能となった。

6.2 今後の課題

現在のシステムには、以下の課題がある。

まず、今回は日中環境モニタリングシステムの構築を行ったが、通信環境には若干の問題が残っており、改善していく必要がある。

次に、運用保守を効率化するために作成したプログラムについても、機器の負荷の軽減を行い、また、よりシステムを安定に稼働させるためにタイムスタンプを確実に取得可能にする必要がある。

今回は非常に簡易な部分の運用保守の効率化を行った。しかし今後においてシステムの規模が大きくなり複雑化した際に、より多くのトラブル対応データおよびトラブル判定の材料が必要となる。そのため、より細かく分類して復旧対応データを作成する必要がある。また判断材料を増やすために、そして ping 処理に用いる ICMP に応答しない機器がある場合を考慮して、ping コマンドだけではなく他のネットワークコマンドについてもプログラムに組み込むことが望ましい。

謝辞

本研究を進めるにあたり、様々な面でご協力を頂き、日頃から貴重な指摘、支援を頂いた環境情報学科吉崎真司教授及び吉崎研究室、諏訪研究室内の各位に深く感謝致します。

また、北京林業大学内に環境モニタリングシステムを構築するにあたり、様々な面でご協力を頂いた北京林業大学の関係者、MSI の夏燕江様に深く感謝致します。本研究は文部科学省平成 18 年度サイバーキャンパス整備事業の研究予算を一部使用している。

参考文献

[1] Welcome to Davis Instruments

<http://www.davisnet.com/>

[2] ヴァンテージ プロ 2

<http://www.seidensha-ltd.co.jp/~musenlan/weather5.html>

[3] Windows プログラミング愛好会, “Java500 の技,” 技術評論社, 2001 年 6 月

[4] 内藤康二, 諏訪敬祐, “柔軟な構築が可能な環境モニタリングシステム,” 武蔵工業大学環境情報学部情報メディアセンタージャーナル第 7 号, pp. 52-59, 2006 年

[5] Java TIPS INDEX - @IT

<http://www.atmarkit.co.jp/fjava/javatips/index.html>

[6] PHP マニュアル

<http://php.s3.to/man/index.html>

[7] PHP の小部屋

<http://www.komonet.ne.jp/~php/index.shtml>

[8] 田中公祐, 佐藤裕樹, 諏訪敬祐, “ワイヤレスセンサネットワークにおける画像伝送技術に関する研究,” 武蔵工業大学 慣用情報学部情報メディアセンタージャーナル 2007.4 第 8 号, pp. 34-39