

Systemwalker Centric Manager

A decorative graphic consisting of two rectangular blocks. The left block is white with horizontal grey lines, and the right block is dark blue with a lighter blue abstract shape inside.

SNMPトラップ変更ツールキット使用手引書

第1.2版 2004年03月22日

まえがき

本書の目的

SNMPトラップ変更ツールキット（以降、T2Eツールキット(*1)と呼びます）は、Systemwalker Centric Manager(*2)において、機器個別Trapを特定のメッセージに変換するためのT2E変換定義ファイルをより簡単に作成するためのツールを集めたものです。

T2Eツールキットは、T2E変換定義ファイル開発についての以下の作業を支援します。

MIB定義ファイルからの T2E変換定義ファイルの作成とテストの実施
Trap仕様からの T2E変換定義ファイルの作成とテストの実施

T2Eツールキットを使用してT2E変換定義ファイルを作成する場合、SNMPについての基本的な知識とSystemwalker Centric ManagerのTrap受信処理についての基本的な知識が必要です。最初に"付録C T2E変換機能の概要"を参照し、Systemwalker Centric ManagerのTrap受信処理についての基本的な知識を得てください。

また、SNMPについての基本的な知識がない場合、"付録B SNMPの概要"を参照される事をお勧めします。

- *1) T2Eとは、Trap to Eventの略で、Systemwalker Centric Managerで受信したTrap(SNMPトラップ)を、SystemwalkerのEvent(イベント)形式に変換して監視イベント一覧にメッセージを出力する機能を指します。SNMPトラップ変更ツールキットは、T2Eによって出力されるメッセージをカスタマイズするツールであり、その略称としてT2Eツールキットと呼んでいます。
- *2) 本文中は全てSystemwalker Centric Managerで表記しています。旧名称(Systemwalker CentricMGR、またはSystemWalker/CentricMGR)の製品を利用している場合は、それぞれ読み替えてください。

本書の読者

本書とT2Eツールキットは、次のような方を対象者としています。

特定の機器とSystemwalker Centric Managerとの連携機能を提供する開発者
Trapを任意のメッセージで出力するようにSystemwalker Centric Managerをカスタマイズする者

本書の構成

本書は、大きく分けて以下の3章で構成されています。

第1章 動作環境の設定

第2章 使用方法

第3章 応用事例

本書の位置づけ

本書は、以下の製品に対応しています。

SystemWalker/CentricMGR 10.0

Systemwalker CentricMGR 10.1

Systemwalker Centric Manager 11.0 以降

SystemWalker/CentricMGR V10.0L10

Systemwalker CentricMGR V10.0L20

Systemwalker CentricMGR V10.0L21

Systemwalker Centric Manager V11.0L10 以降

注意事項

エディションについて

Systemwalker Centric Manager V11.0L10/11.0 以降では、Systemwalker Centric Managerのエディションの、Standard Editionを“SE”、Enterprise Editionを“EE”、およびGlobal Enterprise Editionを“GEE”と省略していますので、各エディションで読み替えてください。

表記上の注意点

プラットフォーム固有の記述については、記述に先立って以下のラベルを付けてあります。

Windows®の場合



Solaris™ Operating Environmentの場合



Linuxの場合



略語表記について

1. Microsoft® Windows NT® Server network operating system Version 4.0 およびMicrosoft® Windows NT® Workstation operating system Version 4.0を“Windows NT®”と略しています。
2. Microsoft® Windows® 2000 Professional operating system、Microsoft® Windows® 2000 Server operating system およびMicrosoft® Windows® 2000 Advanced Server operating systemを“Windows® 2000”と略しています。
3. Microsoft® Windows® XP Professional operating systemを“Windows® XP”と略しています。
4. Microsoft® Windows Server™ 2003, Standard Edition、Microsoft® Windows Server™ 2003, Enterprise Edition、Microsoft® Windows Server™ 2003, Datacenter EditionおよびMicrosoft® Windows Server™ 2003, Web Editionを“Windows Server™ 2003”と略しています。
5. Windows NT®、Windows® 2000、Windows® XP、およびWindows Server™ 2003上で動作するSystemwalker Centric Managerを“Windows版”と略しています。
6. Systemwalker Centric Manager Enterprise Editionを“Systemwalker Centric Manager EE版”およびSystemwalker Centric Managerと略しています。
7. Solaris™ Operating Environmentを“Solaris OE”と略しています。
8. Solaris™ Operating Environment上で動作するSystemwalker Centric Managerを“Solaris OE版”と略しています。
9. Linux上で動作するSystemwalker Centric Managerを“Linux版”と略しています。

商標について

MS-DOS、Microsoft、Windows、Windowsロゴ、Windows NTは、米国Microsoft Corporationの米国およびその他の国における商標または登録商標です。

Sun、Sun Microsystems、Sunロゴ、Java™ およびすべてのJava™に関連する商標およびロゴは、米国およびその他の国における米国Sun Microsystems, Inc.の商標または登録商標であり、同社のライセンスを受けて使用しています。

UNIXは、米国およびその他の国におけるオープン・グループの登録商標です。

Solaris™およびすべてのSolaris™に関連する商標およびロゴは、米国およびその他の国における米国Sun Microsystems, Inc.の商標または登録商標であり、同社のライセンスを受けて使用しています。

Red HatおよびRed Hatをベースとした全ての商標とロゴは、米国およびその他の国におけるRed Hat, Inc.の商標または登録商標です。

Linuxは、Linus Torvalds氏の米国およびその他の国における登録商標または商標です。

Systemwalkerは、富士通株式会社の登録商標です。

その他、本書に記載の会社名および製品名などは、該当する各社の商標または登録商標です。

謝辞

TclおよびTkの著作権表示・使用許諾条件について

本ソフトウェアは、カリフォルニア大学、サン・マイクロシステムズ社、Scriptics社他が著作権を所有しているフリーソフトTcl/Tkのプログラム、およびライブラリを含んでいます。

TclおよびTkの著作権表示・使用許諾条件の表示は以下の通りです。

This software is copyrighted by the Regents of the University of California, Sun Microsystems, Inc., Ajuba Solutions, and other parties. The following terms apply to all files associated with the software unless explicitly disclaimed in individual files.

The authors hereby grant permission to use, copy, modify, distribute, and license this software and its documentation for any purpose, provided that existing copyright notices are retained in all copies and that this notice is included verbatim in any distributions. No written agreement, license, or royalty fee is required for any of the authorized uses. Modifications to this software may be copyrighted by their authors and need not follow the licensing terms described here, provided that the new terms are clearly indicated on the first page of each file where they apply.

IN NO EVENT SHALL THE AUTHORS OR DISTRIBUTORS BE LIABLE TO ANY PARTY FOR DIRECT, INDIRECT, SPECIAL, INCIDENTAL, OR CONSEQUENTIAL DAMAGES ARISING OUT OF THE USE OF THIS SOFTWARE, ITS DOCUMENTATION, OR ANY DERIVATIVES THEREOF, EVEN IF THE AUTHORS HAVE BEEN ADVISED OF THE POSSIBILITY OF SUCH DAMAGE.

THE AUTHORS AND DISTRIBUTORS SPECIFICALLY DISCLAIM ANY WARRANTIES, INCLUDING, BUT NOT LIMITED TO, THE IMPLIED WARRANTIES OF MERCHANTABILITY, FITNESS FOR A PARTICULAR PURPOSE, AND NON-INFRINGEMENT. THIS SOFTWARE IS PROVIDED ON AN "AS IS" BASIS, AND THE AUTHORS AND DISTRIBUTORS HAVE NO OBLIGATION TO PROVIDE MAINTENANCE, SUPPORT, UPDATES, ENHANCEMENTS, OR MODIFICATIONS.

GOVERNMENT USE: If you are acquiring this software on behalf of the U.S. government, the Government shall have only "Restricted Rights" in the software and related documentation as defined in the Federal Acquisition Regulations (FARs) in Clause 52.227.19 (c) (2). If you are acquiring the software on behalf of the Department of Defense, the software shall be classified as "Commercial Computer Software" and the Government shall have only "Restricted Rights" as defined in Clause 252.227-7013 (c) (1) of DFARs. Notwithstanding the foregoing, the authors grant the U.S. Government and others acting in its behalf permission to use and distribute the software in accordance with the terms specified in this license.

目次

第1章 動作環境の設定	1
1.1 前提プログラム	2
1.1.1 OS	2
1.1.2 Systemwalker Centric Manager	2
1.2 インストール	3
1.3 アンインストール	4
第2章 使用方法	5
2.1 MIB定義ファイルの情報からT2E変換定義を作成する	6
2.1.1 MIB拡張	6
2.1.2 変換	6
2.1.3 変換結果の修正	8
2.1.4 テストデータの生成	9
2.1.5 テストの実施	10
2.2 Trap仕様からT2E変換定義ファイルを作成する	12
2.2.1 Trap仕様の確認	12
2.2.2 ファイルの作成	12
2.2.3 テストデータの生成	14
2.2.4 テストの実施	14
第3章 応用事例	17
3.1 機器開発元が連携機能を提供する場合	18
3.2 機器連携アプリを提供する場合	20
3.3 既存のTrapメッセージを変更する	22
3.4 特別なMIBしきい値監視を行う	24
付録A コマンドリファレンス	27
A.1 CNSetCnfMg	28
A.2 mib2cnf	29
A.3 name2dot	30
A.4 t2check	32
A.5 t2edcnf	34
A.6 疑似トラップ生成ツール	36
付録B SNMPプロトコルの概要	37
B.1 概要	38
B.2 プロトコル	39
B.2.1 下位プロトコル	39
B.2.2 PDU	40
B.2.3 PDUの形式	42
B.2.4 補足事項	45
B.3 MIB	46
B.3.1 OBJECT IDENTIFIER	46
B.3.2 クラス, 型およびインスタンス	47

B.3.3 MIB の特徴.....	50
B.4 認証 / 権限機構.....	51
B.5 MIB定義ファイルの読み方.....	52
付録C T2E変換機能の概要.....	59
C.1 T2E変換.....	60
C.2 MIB拡張操作とT2E変換機能.....	63
C.3 メッセージのカスタマイズ.....	64
C.4 変換メッセージの決定論理.....	69
C.5 T2E変換定義ファイルの適用.....	70

第1章 動作環境の設定

本章では、T2Eツールキットのインストールについて説明します。

1.1 前提プログラム

1.1.1 OS

以下のオペレーティングシステム上での動作を確認しています。

Microsoft® Windows NT® Workstation 4.0 ServicePack 5/6a 以降

Windows NT® Server , Windows® 2000 , Windows® XP、Windows Server™ 2003 上での操作も可能です。以降、動作可能なOSの総称としてWindowsとして表記します。

1.1.2 Systemwalker Centric Manager

本ツールキットは、開発からテストまでの一連の作業を行うにあたり、Systemwalker Centric Managerの機能を前提としています。このため、V10.0L10以降のWindows版 運用管理サーバ機能、もしくは、V10.0L10 / 10.0以降の運用管理クライアントがインストールされているWindowsシステムに必ずインストールしなければなりません。

本ツールキットは以下のVLに対応しています。

Systemwalker Centric Manager V/L	運用管理サーバ	運用管理クライアント
Windows版 V10.0L10 以降		
Linux版 V11.0L10 以降	×	
Solaris OE版 10.0 以降	×	



注意

運用管理クライアントでは、適用コマンド (CNSetCnfMg.exe) 機能が使用できません。“C.5 T2E変換定義ファイルの適用”を参照し、運用管理サーバ上で適用してください。

1.2 インストール

本ツールキットのインストールは、以下の手順で行います。

- 1) 管理者権限でログインする
インストール先システムの運用管理サーバ（または運用管理クライアント）にSystemwalker管理者のユーザIDまたはAdministrator権限でログインする。
- 2) 適当なディレクトリに展開する
本ツールキットは、自己解凍圧縮ファイル“t2etk10.exe”に格納されています。これを適当なディレクトリで実行します。
- 3) アプリケーションを終了する
不要なアプリケーションやウィンドウを全て閉じます。
- 4) インストールコマンドを実行する
コマンド プロンプトを起動し、展開したツールキットのインストールコマンドを実行します。コマンドの引数にはインストール先ディレクトリを指定します。

実行例

```
C:¥TEMP¥t2etk10¥T2Emodules¥setup.bat C:¥WIN32APP¥T2EtoolKit
```

- 5) 環境変数の設定
[設定] [コントロールパネル] [システム] [環境]を開き、システム環境変数“PATH”にツールキットインストール先ディレクトリの指定を追加設定します。設定を有効にするためにシステムを再起動してください。

1.3 アンインストール

本ツールキットのアンインストールは、以下の手順で行います。

- 1) 管理者権限でログインする
インストール先システムの運用管理サーバ（または運用管理クライアント）にSystemwalker管理者のユーザIDまたはAdministrator権限でログインする。
- 2) アプリケーションを終了する
不要なアプリケーションやウィンドウを全て閉じます。
- 3) インストールディレクトリを削除する
エクスプローラ等でインストールディレクトリを削除します。
- 4) 環境変数を設定する
[設定] [コントロールパネル] [システム] [環境] を開き、システム環境変数“PATH”からツールキットインストール先ディレクトリの指定を削除設定します。設定を有効にするためにシステムを再起動してください。

第2章 使用方法

ここでは、一連の作業の流れにそって 本ツールキットの使い方を説明します。

2.1 MIB定義ファイルの情報からT2E変換定義を作成する

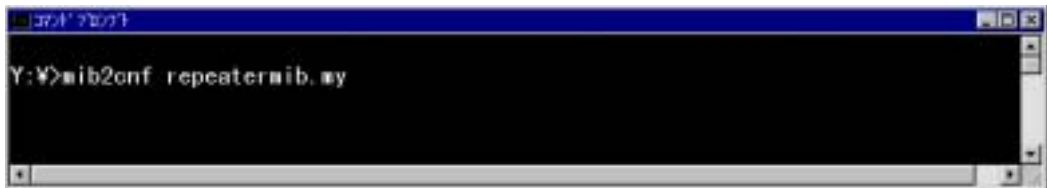
個別の機器の拡張MIB定義ファイルが入手できた場合の作業の流れを以下に説明します。

2.1.1 MIB拡張

Systemwalker コンソールを起動し、MIB拡張操作を行います。MIB拡張操作についてはSystemwalker Centric Managerのマニュアルを参照してください。

2.1.2 変換

mib2cnfコマンドを使用して mib定義ファイルのデータを変換します。



コマンドが終了すると、カレントディレクトリ配下に、*.cnfという名前のファイルが生成されます。

生成されるファイルが複数存在する事もあるので注意して下さい。これは、入力としたmib定義ファイルに、複数の Enterprises-01D の Trapに関する定義が存在する場合があるためです。

以下に処理結果の例を示します。

```

snmpDot3RptrMgt - 変換
776(K/E) 編集(E) 検索(S) ^67(H)
# enterprises=1, 3, 6, 1, 2, 1, 22
# trap -> event conversion setup file
## rptrHealth ( Enterprise=snmpDot3RptrMgt 1, 3, 6, 1, 2, 1, 22 ) definition begin
# TRAP-NAME : rptrHealth
# Enterprises : snmpDot3RptrMgt (1, 3, 6, 1, 2, 1, 22)
# Specific : 1
# description : The rptrHealth trap conveys information related to the
#               operational status of the repeater. This trap is sent
#               only when the oper status of the repeater changes. The
#               rptrHealth trap must contain the rptrOperStatus object.
#               The agent may optionally include the rptrHealthText
#               object in the varbind list. See the rptrOperStatus and
#               rptrHealthText objects for descriptions of the
#               information that is sent. The agent must throttle the
#               generation of consecutive rptrHealth traps so that
#               there is at least a five-second gap between them
# Variables : rptrOperStatus
#
# - rptrOperStatus information
#   id : 1, 3, 6, 1, 2, 1, 22, 1, 1, 2
#   type : Integer
#           other(1), ok(2), rptrFailure(3), groupFailure(4), portFailure(5),
# generalFailure(6)
#   description : The rptrOperStatus object indicates the
#                 operational state of the repeater. The
#                 rptrHealthText object may be consulted
#                 for more specific information about the
#                 state of the repeater's health. In the
#                 case of multiple kinds of failures (e.g.,
#                 repeater failure and port failure), the
#                 value of this attribute shall reflect
#                 the highest priority failure in the
#                 following order: rptrFailure(3)
#                 groupFailure(4) portFailure(5)
#                 generalFailure(6).
#
#
### Warning !! You should add Japanese and English message texts.
TRAP:
- 4 - - 1, 3, 6, 1, 2, 1, 22 - 6 1 -
VARBIND:
- - - -
MESSAGE:
SnailJ:
" [%hostx:%AGENT_ADDR%] rptrHealth トラップを受信しました. "
SnailE:
" [%hostx:%AGENT_ADDR%] Trap (rptrHealth) received. "
### Warning !! You should check EventType parameter.
EVENTTYPE:
ERROR
### Warning !! You should check Treshold parameter.
Threshold:
1
## rptrHealth ( Enterprise=snmpDot3RptrMgt ) definition end

```

変換結果には、コメントの形（先頭が“#”で始まる行）で、TRAPの定義情報が整理して出力されます。

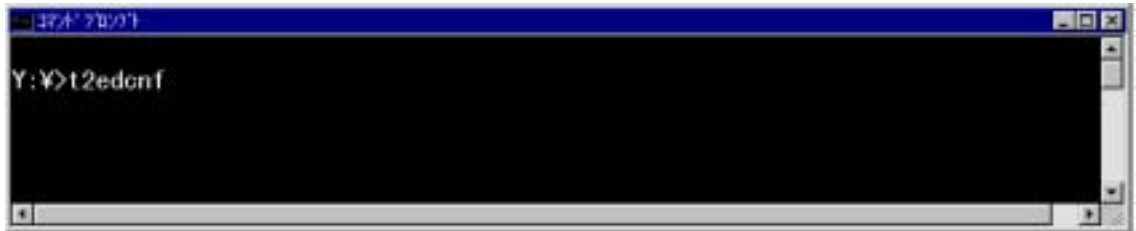
この情報をもとにして、T2E変換定義ファイルエディタ等を用いて、変換メッセージ等（“!! Warning !!”で始まるメッセージが出力されています）の変更を行って下さい。

2.1.3 変換結果の修正

前項の例に示すように、変換結果内には、Trapの情報を整理したリストおよび "!! Warning !!" または "!! Error !!" で始まるメッセージがコメントの形で出力されています。

T2E変換定義ファイルエディタを用いて、以下のようにメッセージに従いT2E変換定義を修正します。

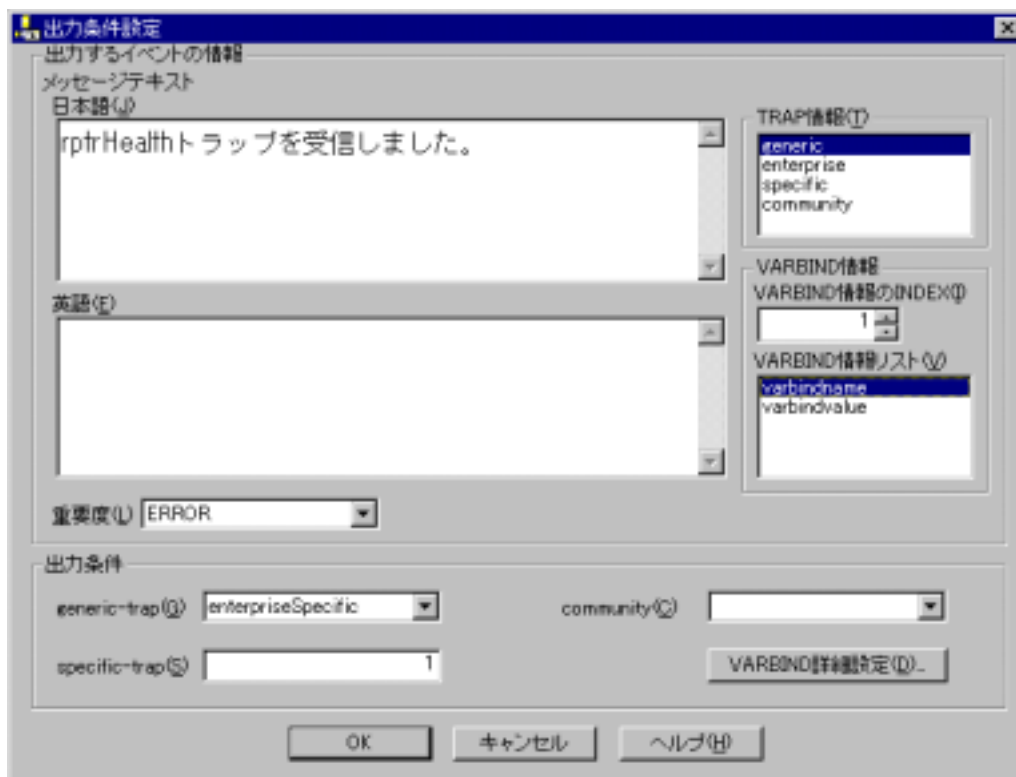
“t2edcnf”コマンドを入力してトラップイベント変換定義作成画面を起動します。



トラップイベント変換定義作成画面で [ファイル参照] ボタンをクリックし、T2E変換定義ファイルを読み込みます。



修正対象のメッセージを選択して [編集] ボタンをクリックします。

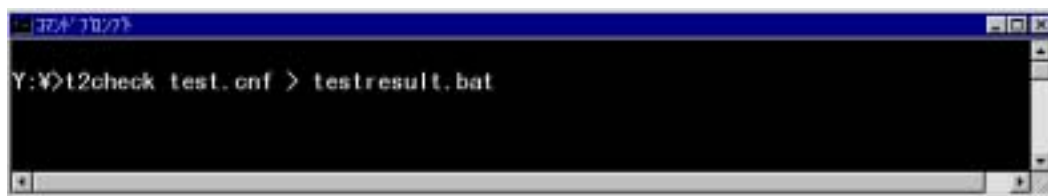


編集画面で変換するメッセージを修正します。

この例では、MIB定義中のdescriptionの記述をもとに、和文および英文メッセージを修正し、必要であればVarbind情報をメッセージに挿入する事になります。

2.1.4 テストデータの生成

t2checkコマンドを用いてテストデータを生成します。この例では、“test.cnf”というファイル中のT2E変換定義をチェックして、結果を“testresult.bat”というファイルに出力します。

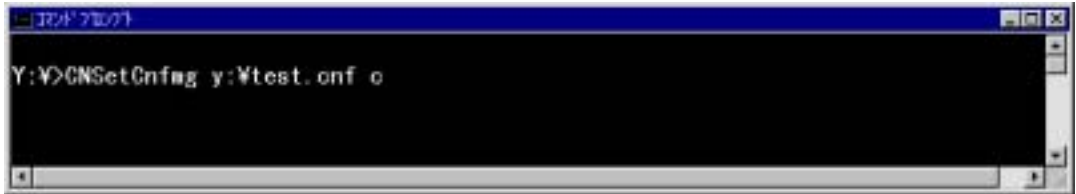


チェックの結果、Warningまたは Error で始まるメッセージが出力される事があります。この場合、メッセージの内容に従い、T2E変換定義ファイルの内容を修正してください。

2.1.5 テストの実施

a) T2E変換定義ファイルの適用

CNSetCnfMgコマンドでT2E変換定義ファイルを適用します。



上記の例では、“y:\test.cnf”というファイル（フルパス指定である事に注意）を、置き換え指定で提供する場合を示しています。

コマンド終了後、Systemwalkerコンソールを起動してポリシー配付を行います。



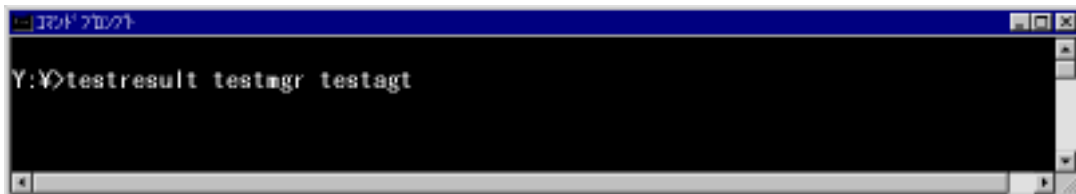
注意

T2Eツールキットを運用管理クライアントにインストールした場合、運用管理サーバ上のCNSetCnfMg.exeコマンドを使用します。（“付録C.5 T2E変換定義ファイルの適用”を参照）

b) 疑似Trapテストの実施

予め、起動したSystemwalkerコンソールで、最新のイベントIDを記録しておきます。

コマンドプロンプトから t2checkコマンドで生成した疑似トラップ生成コマンドを投入します。



上記の例では、“testmgr”という運用管理サーバに対して“testagt”というシステムからTrapを上げるようテストコマンドを投入しています。

コマンド処理の結果、Systemwalkerコンソールにアラームが表示されます。MIB定義ファイルにもよりますが、出力されるイベントの数は複数の場合もあります。イベントログの内容をcsvファイルに格納し、以下を確認します。

意図した通りに変換が行われているかどうか

その他のTrapの変換に影響を与えていないかどうか（詳細は“A.5 t2check”のT2Eテストプログラム作成を参照）。

c) 実機テスト

前項までで、作成したT2E変換定義ファイルが意図した通りの変換を行う事まで確認できました。

最後に必ず、実機を用いて実際のTrapを上げてテストを実施して下さい。ここまでのテストで確認した事は、外部から提供された "仕様" を元に全てを設定した時にそれが正しいかどうか、という事です。提供された "仕様" が正しいかどうかのテストではありません。

Trapの内容によっては全てをテストするのは難しいかもしれません。しかし、上述のように "仕様" が正しいかどうかという観点から、実機テストが必ず必要です。

2.2 Trap仕様からT2E変換定義ファイルを作成する

ここでは、拡張MIB定義ファイルが入手が困難でTrap仕様の情報のみをもっている場合の作業の流れを説明します。

2.2.1 Trap仕様の確認

以下のいずれかの方法でTrap仕様の確認を行います。

Trap通知元の開発元に確認する
実際にTrapをSystemwalker Centric Managerで受信してみる

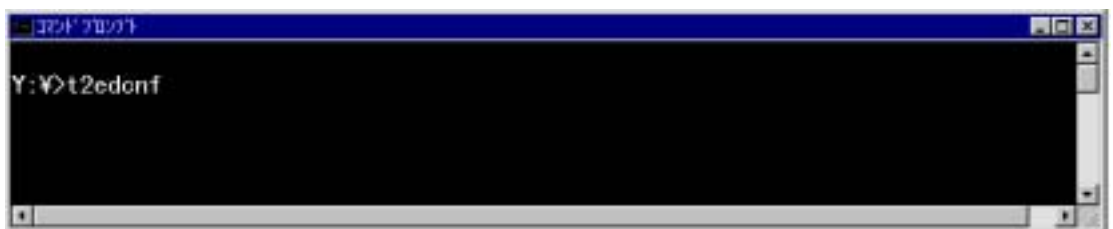
確認が必要なのは以下の項目です。

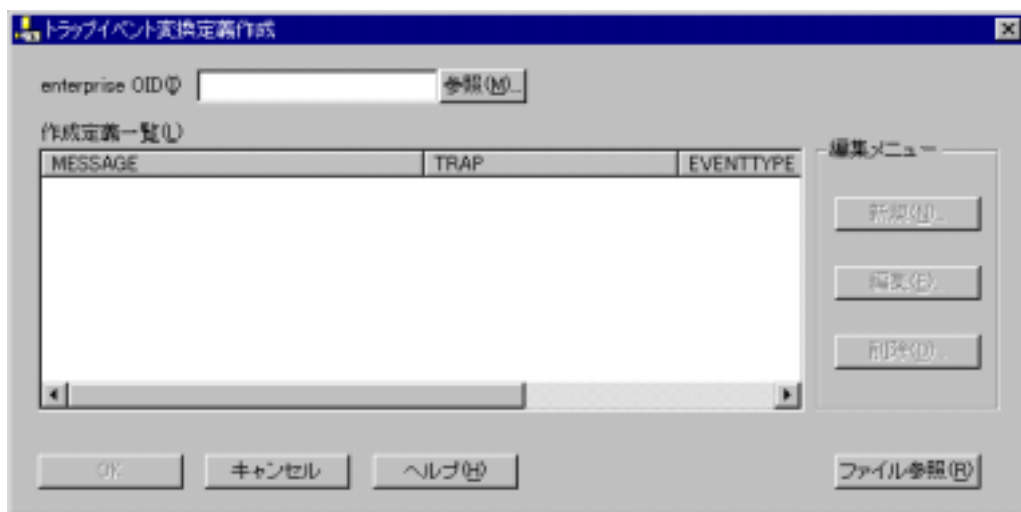
1. Enterprise-OIDの値（ドット表記）
2. Specific-Codeの値
3. Varbind情報の仕様（一つのSpecific-Codeに複数の意味をもたせている場合、それを識別するためのドット表記のObject識別子、型、値、意味）
4. メッセージを作成するために必要なTrapの意味

2.2.2 ファイルの作成

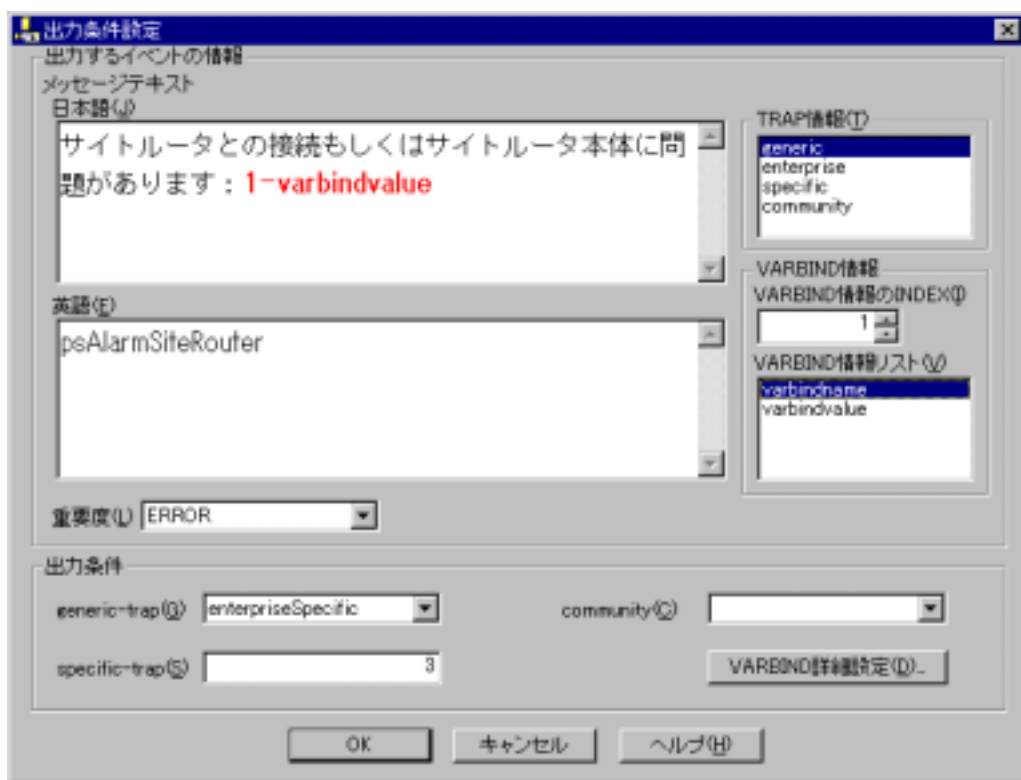
T2E変換定義ファイルエディタを用いて、以下のようにメッセージに従いT2E変換定義を作成します。

“t2edcnf”コマンドを入力してトラップイベント変換定義作成画面を起動します。





enterprise OID入力フィールドに、TrapのEnterprise-OID情報を入力します。
[新規] ボタンをクリックして、メッセージ変換エントリの作成を行います。



編集画面で、以下の入力フィールドに出力条件を設定します。

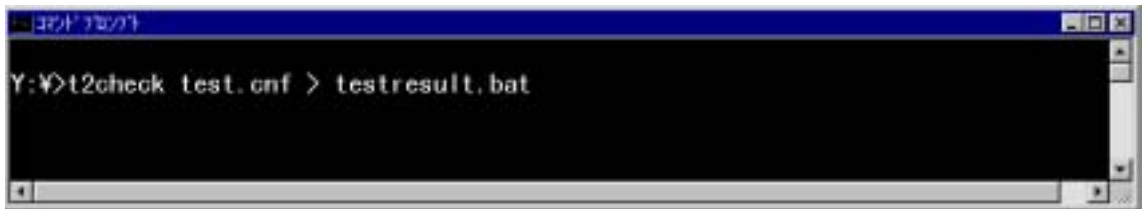
入力フィールド	出力条件
generic-trap	Generic-Codeに対応したトラップ種別
specific-trap	Specific-Codeに対応した値

次にメッセージテキストを作成します。カーソルを編集中のテキストに置いて、TRAP情報やVARBIND情報の該当情報をダブルクリックすると、その文字列がテキスト中にうめこまれます。編集方法の詳細は、ヘルプを参照ください。

メッセージテキストを終えた後 [OK] ボタンをクリックし、トラップイベント定義作成画面に戻ります。[OK] ボタンをクリックし、ファイルに変換定義情報を格納します。

2.2.3 テストデータの生成

t2checkコマンドを用いてテストデータを生成します。この例では、“test.cnf”というファイル中のT2E変換定義をチェックして、結果を“testresult.bat”というファイルに出力します。



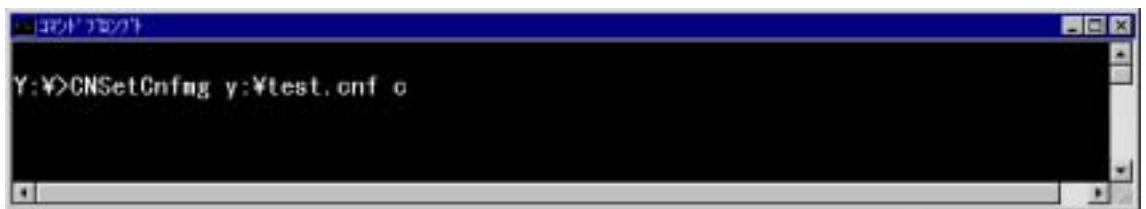
チェックの結果、Warningまたは Error で始まるメッセージが出力される事があります。この場合、メッセージの内容に従い、T2E変換定義ファイルの内容を修正してください。

2.2.4 テストの実施

a) T2E変換定義ファイルの適用

CNSetCnfMgコマンドでT2E変換定義ファイルを適用します。

上記の例では、“y:¥test.cnf”というファイル(フルパス指定である事に注意)を、置き換え指定で提供する場合を示しています。



コマンド終了後、Systemwalkerコンソールを起動してポリシー配付を行います。

**注意**

T2Eツールキットを運用管理クライアントにインストールした場合、運用管理サーバ上のCNSetCnfMg.exeコマンドを使用します。(“ 付録C.5 T2E変換定義ファイルの適用 ” を参照)

b) 疑似Trapテストの実施

予め、起動したSystemwalkerコンソールで、最新のイベントIDを記録しておきます。

コマンドプロンプトから t2checkコマンドで生成した疑似トラップ生成コマンドを投入します。



上記の例では、“testmgr” という運用管理サーバに対して “testagt” というシステムからTrapを上げるようテストコマンドを投入しています。

コマンド処理の結果、Systemwalker監視画面にアラームが表示されます。イベントログの内容をcsvファイルに格納し、以下を確認します。

意図した通りに変換が行われているかどうか

その他のTrapの変換に影響を与えていないかどうか (詳細は “ A.5 t2check ” のT2Eテストプログラム作成を参照)。

c) 実機テスト

前項までで、作成したT2E変換定義ファイルが意図した通りの変換を行う事まで確認できました。

最後に必ず、実機を用いて実際のTrapを上げてテストを実施して下さい。ここまでのテストで確認した事は、外部から提供された “仕様” を元に全てを設定した時にそれが正しいかどうか、という事です。提供された “仕様” が正しいかどうかのテストではありません。

Trapの内容によっては全てをテストするのは難しいかもしれません。しかし、上述のように “仕様” が正しいかどうかという観点から、実機テストが必ず必要です。

第3章 応用事例

ここでは、T2Eツールキットを使用する具体的な応用事例について紹介します。

3.1 機器開発元が連携機能を提供する場合

あなたは、ある機器（たとえばネットワークプリンタ、ルータ、ハブ）の開発者であるとして。

あなたの開発した機器とSystemwalker Centric Managerを連携させる場合、T2E変換定義ファイルを機器に添付することで、機器のアラーム情報をユーザフレンドリな形で監視イベント一覧に表示する事が可能になります。

これを実現するためには以下のような作業を行います。

- 1) T2E変換定義ファイルを作成する
あなたの開発した機器のTrap仕様にしたが、T2E変換定義ファイルを作成します。作成手順については、2章を参照してください。
- 2) 適用コマンドを作成する
あなたの作成したT2E変換定義ファイルを顧客が容易に適用できるようにコマンド（バッチファイル）を作成します。T2E変換定義適用コマンドは本ツールキットのCNSetCnfMg.batではなく、Systemwalker Centric ManagerにインストールされているCNSetCnfMg.exeを使用するようにしてください。詳細については“C.5 T2E変換定義ファイルの適用”を参照してください。
- 3) T2E変換定義ファイルと適用コマンドを提供媒体（フロッピーディスク）に格納し、使用説明書とともに機器に添付して出荷する。
使用説明書には、適用コマンドの使用方法和コマンド実施後にSystemwalker Centric Managerにて“ポリシ配付”を実施する事を明記します。



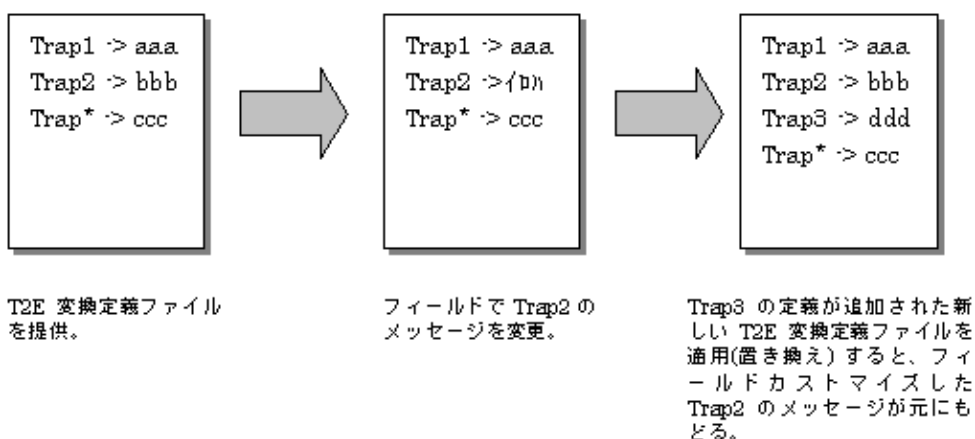
a) 変換定義に関して

T2E変換定義ファイルの作成にあたっては、個々のTrapそれぞれの変換定義を作成し、ワイルドカード的な変換定義を用いない事を推奨します。

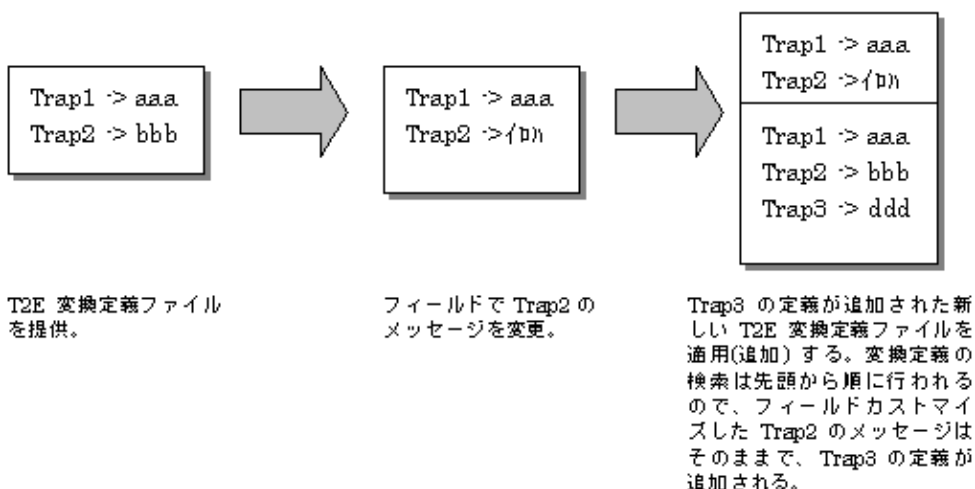
上記を推奨する理由は、ワイルドカード的な変換定義を用いていると、Trapが追加された場合にもT2E変換定義ファイル全体の置き換えが必要になってしまうためです。このような方式では、フィールドにおけるカスタマイズで変換定義ファイルの修正を行っている場合、Trap追加による修正（置き換え）のためにカスタマイズが初期状態にもどることになります。

個々のTrapそれぞれの変換定義を格納した形式ならば、追加指定で適用することが可能になり、フィールドでのカスタマイズに影響を与える事なく修正（追加）が可能になります。（次図参照）。

[ワイルドカード指定の場合]



[個々に指定した場合]



b) 適用パッチコマンドに関して

適用パッチコマンドについて、前項で説明した理由から追加指定でT2E変換定義ファイルを適用するように作成する事を推奨します。

本ツールキットでは、パッチコマンドのひな型を、docディレクトリ配下の“samplebat.bat”というファイル名で提供しています。参考にしてください。

3.2 機器連携アプリを提供する場合

あなたは、ある機器（たとえばネットワークプリンタ、ルータ、ハブ）をターゲットとしたSystemwalker Centric Manager上で動作する連携アプリケーション開発者であるとして。

あなたの開発した連携アプリケーションの付加価値を高めるため、機器から受信するTrapについて個別のメッセージを提供したい場合、T2E変換定義ファイルをアプリケーションに添付することで対応できます。

これを実現するためには以下のような作業を行います。

- 1) T2E変換定義ファイルを作成する
連携対象の機器のTrap仕様にしたいT2E変換定義ファイルを作成します。作成手順については、3章を参照してください。
- 2) 適用コマンドを作成する
あなたの作成したT2E変換定義ファイルを顧客が容易に適用できるようにコマンド（バッチファイル）を作成します。T2E変換定義適用コマンドは本ツールキットのCNSetCnfMg.batではなく、Systemwalker Centric ManagerにインストールされているCNSetCnfMg.exeを使用するようにしてください。詳細については“C.5 T2E変換定義ファイルの適用”を参照してください。
- 3) T2E変換定義ファイルと適用コマンドをアプリケーションの提供媒体に格納し、使用説明書とともに機器に添付して出荷する。
使用説明書には、適用コマンドの使用方法和コマンド実施後にSystemwalker Centric Managerにて“ポリシ配付”を実施する事を明記します。



注意

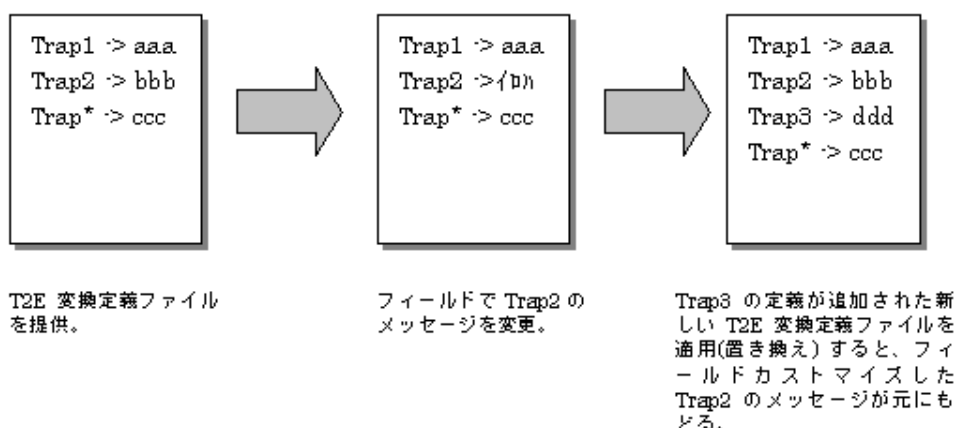
a) 変換定義に関して

T2E変換定義ファイルの作成にあたっては、個々のTrapそれぞれの変換定義を作成し、ワイルドカード的な変換定義を用いない事を推奨します。

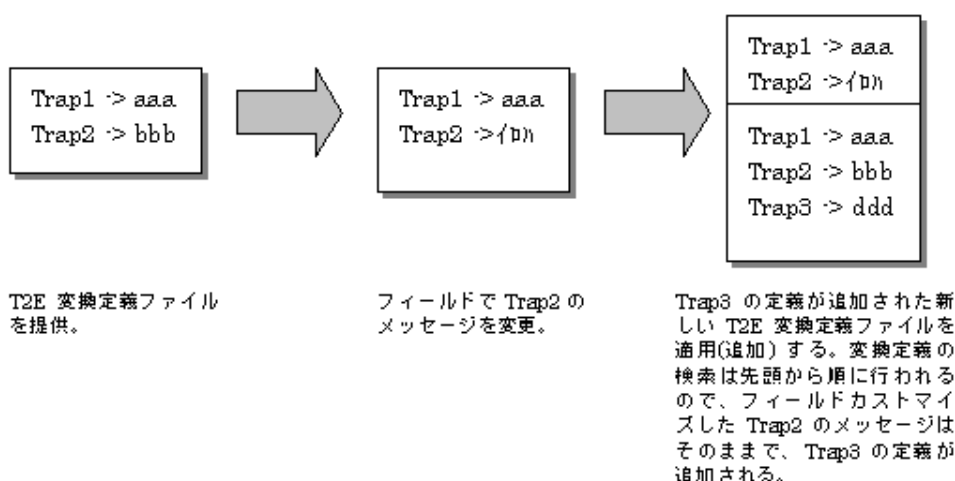
上記を推奨する理由は、ワイルドカード的な変換定義を用いていると、Trapが追加された場合にもT2E変換定義ファイル全体の置き換えが必要になってしまうためです。このような方式では、フィールドにおけるカスタマイズで変換定義ファイルの修正を行っている場合、Trap追加による修正（置き換え）のためにカスタマイズが初期状態にもどることになります。

個々のTrapそれぞれの変換定義を格納した形式ならば、追加指定で適用することが可能になり、フィールドでのカスタマイズに影響を与える事なく修正（追加）が可能になります。（次図参照）。

[ワイルドカード指定の場合]



[個々に指定した場合]



b) 適用バッチコマンドに関して

適用バッチコマンドについて、前項で説明した理由から追加指定でT2E変換定義ファイルを適用するように作成する事を推奨します。

本ツールキットでは、バッチコマンドのひな型を、docディレクトリ配下の“samplebat.bat”というファイル名で提供しています。参考にしてください。

c) アプリケーションのインストーラとの関係

あなたの作成するアプリケーションが、Systemwalker Centric Manager運用管理サーバと同一のシステムにインストールされる仕様ならば、インストーラの中から適用バッチコマンドを呼び出して、T2E変換定義ファイルの適用を自動化する事も可能です。

3.3 既存のTrapメッセージを変更する

あなたは、ある特定のお客様のためにSystemwalker Centric Managerによる管理システムの構築を請け負っているとします。

お客様から、ある特定の機器から通知されるTrapに対して、特別なメッセージを表示して管理したいという意向があった場合、T2E変換定義ファイルを作成しシステムに適用する事で対応できます。

これを実現するためには以下のような作業を行います。

- 1) 対象Trapの仕様を確認する
機器から通知されるTrapの仕様を確認します。拡張MIBファイルが入手可能ならばこれを入手しておきます。
- 2) 既にT2E変換定義ファイルが提供されているかどうか確認する
対象の機器から通知される Trapについて、既にT2E変換定義ファイルが存在しているかどうか確認します。確認方法は、注意事項として後述します。
- 3) T2E変換定義ファイルを作成する
T2E変換定義ファイルが存在すれば、既存のT2E変換定義ファイルをひな型にお客様の意向に沿うようにメッセージを変更します。既存のT2E変換定義ファイルが存在しない場合は、Trap仕様からT2E変換定義ファイルを作成します。
作成方法については、第3章を参照してください。
- 4) T2E変換定義ファイルを適用しテストを実施する



a) Trapの仕様確認に関して

確認が必要なのは以下の項目です。

Enterprise-OIDの値 (ドット表記)

Specific-Codeの値

Varbind情報の仕様 (一つのSpecific-Codeに複数の意味をもたせている場合、それを識別するためのドット表記のObject識別子、型、値、意味) メッセージを作成するために必要なTrapの意味

詳細は、“3.3 Trap仕様からT2E変換定義ファイルを作成する”を参照してください。

b) T2E変換定義ファイルが既に提供されているかどうかの確認に関して

特定のTrapに対してT2E変換定義ファイルが提供されているかどうかの確認は、以下に行います。

1. TrapのEnterprise-01Dのドット表記を確認する。
2. Systemwalker Centric Managerインストールディレクトリ配下の以下のディレクトリに、“B[ドット表記].cnf”というファイル名が存在するかどうかを確認する。
Windows NT版
[Install-dir]¥Mpwalker.dm¥MpCNappl¥MpCNagt¥etc¥master
Solaris OE版/Linux版
/opt/FJSVfwntc/MpCNagt/etc/master
3. ファイルが存在する場合、T2E変換定義ファイルはすでに適用されています。

c) T2E変換定義ファイルの作成に関して

特定のTrapに対して既にT2E変換定義ファイルが提供されていた場合、以下に注意してください。

Systemwalker Centric Managerインストールディレクトリ配下のディレクトリから既存のT2E変換定義ファイルの取り出しは、Copyを用いてください。不用意にこのディレクトリ配下のファイルを削除した場合、他の変換定義に影響がでます。

Systemwalker Centric Managerインストールディレクトリ配下のディレクトリから取り出したT2E変換定義ファイルの先頭には内部制御レコード (“# CHNGE “または “# ADD”) が含まれています。このファイルをひな型としてT2E変換定義ファイルを作成する場合、この先頭レコードを削除する必要があります。

あらたなT2E変換定義ファイルを作成する前に、既存のT2E変換定義ファイルを必ずバックアップしておいてください。バックアップファイルは、T2E変換定義を元に戻す場合に必要となります。

3.4 特別なMIBしきい値監視を行う

あなたは、ある特定のお客様のためにSystemwalker Centric Managerによる管理システムの構築を請け負っているとします。

お客様から、ある特定の機器の拡張MIBのしきい値監視を行い、特別なメッセージを表示して管理したいという意向があった場合、T2E変換定義ファイルを作成しシステムに適用する事で対応できます。

これを実現するためには以下のような作業を行います。

1. しきい値監視時にSystemwalker Centric Managerが発行する内部Trapの仕様を設定する
2. T2E変換定義ファイルを作成する
Systemwalker Centric Managerが発行する内部Trapに対応してお客様の意向に沿うようにメッセージを表示するような定義を作成します。
作成方法については、第2章を参照してください。
3. T2E変換定義ファイルを適用しテストを実施する



注意

a) 内部Trapの仕様設定に関して

Systemwalker Centric Managerは、MIBしきい値監視の結果を内部的に生成するTrapにより通知します。デフォルトでは、このTrapの仕様は以下のようになります。

```
Enterprise-OID      :  
enterprises.211.4.19.3(1.3.6.1.4.1.211.4.19.3)  
Specific-Code       : 0  
VarBind              : 監視対象のMibオブジェクト
```

これらの仕様は、MIBしきい値監視の設定時に変更する事が可能です。変更は以下のダイアログボックスで行います。



複数のMIBについてしきい値監視を行う場合、SpecificCodeを変更する事で区別する事ができます。

Enterprise-OIDについては、特に変更はいりません。デフォルトの値は、富士通がSystemwalker Centric Managerの内部Trapのために用意したIDです。

付録A コマンドリファレンス

本ツールキット中では、各種コマンドや作業上役に立つ情報が提供されています。

この章では、提供コマンドのシンタックスについて説明します。コマンド起動は、コマンドプロンプトウィンドから行ってください。

A.1 CNSSetCnfMg

形式

```
CNSSetCnfMg T2E変換定義ファイル { a | c }
```

機能

T2E変換定義ファイルをSystemwalker Centric Managerに適用します。
本コマンドを運用管理サーバ上で発行し、ポリシー配付を実施する事により、運用管理サーバおよび全部門管理サーバにT2E変換定義を適用する事ができます。

オプション

オプション	意味
a	追加モードで定義ファイルを適用する。
c	上書モードで定義ファイルを適用する。

オペランド

オペランド	意味
T2E変換定義ファイル	適用する定義ファイルを絶対パスで指定します。

注意事項

本コマンドは運用管理クライアント上では動作しません。T2Eツールキットを運用管理クライアントにインストールした場合は、運用管理サーバのCNSSetCnfMgコマンドを実行してください。（詳細は“付録C.5 T2E変換定義ファイルの適用”を参照してください。）

実行例

実行例1

C:¥TEMP¥t2e¥test.cnfファイルのT2E変換定義を追加します。

```
C:¥t2eToolKit¥CNSSetCnfMg C:¥TEMP¥t2e¥test.cnf a
```

実行例2

C:¥TEMP¥t2e¥test.cnfファイルのT2E変換定義に変更します。

```
C:¥t2eToolKit¥CNSSetCnfMg C:¥TEMP¥t2e¥test.cnf c
```

A.2 mib2cnf

形式

mib2cnf 入力ファイル

機能

MIB定義ファイルからのT2E変換定義ファイルの作成を支援します
MIB定義ファイルを入力として、

Trapの定義を整理した情報、と
T2E変換定義ファイルフォーマットのひな型情報

を出力します。入力となるMIB定義ファイルは、あらかじめ本ツールキットをインストールしたシステムのSystemwalker Centric ManagerにおいてMIB拡張操作を行っていない必要があります。

本コマンドを実行すると、カレントディレクトリに "*.cnf" (*は不定)というファイル名で新たにT2E変換定義ファイルが作成されます。

Trap定義を整理した情報は、出力されたT2E変換定義ファイル中にコメントの形で示されます。また、注意すべき事柄について、コメントの形でメッセージが出力されます。

オプション

なし。

オペランド

オペランド	意味
入力ファイル	MIB定義ファイル名 (xxxx.my) を指定します。

注意事項

本コマンドの入力となるMIB定義ファイルは、あらかじめSystemwalker Centric ManagerにてMIB拡張操作を行い登録しておく必要があります。

生成されるファイルが複数存在する事もあるので注意して下さい。これは、入力としたMIB定義ファイルに、複数の Enterprise-01D の Trapに関する定義が存在する場合があるためです。

実行例

実行例

rcpcatrcmib.myファイルをT2E変換定義ファイルに変換します。コマンドが終了すると、カレントディレクトリ配下に、*.cnfという名前のファイルが生成されます。

C:\t2eToolKit\mib2cnf rcpcatrcmib.my

A.3 name2dot

形式

name2dot oid [oid ...]

機能

MIB の変数名とドット形式表記の相互変換を行うコマンドです。オペランドにMIBの名前かドット表記を指定して実行すると、相互変換した結果とそのMIBオブジェクトまでのオブジェクトIDの全階層が表示されます。また、複数のオペランドを指定することも可能です。

相互変換は、本ツールキットの動作するシステムのSystemwalker Centric Managerに登録してある範囲のMIBについて行われます。

オプション

なし。

オペランド

オペランド	意味
oid	MIB名またはMIBのOIDを指定します。複数のOIDを指定することも可能です。

注意事項

本コマンドでは、Systemwalker Centric Managerに登録してある範囲のMIBについて相互変換を行います。拡張MIBに定義されているMIBオブジェクトの変数を変換する場合、予めSystemwalker Centric ManagerのMIB拡張操作で対象の拡張MIBを登録しておく必要があります。

実行例

実行例 1

```
C:\t2eToolKit¥name2dot ifIndex.1
namepart = ifIndex
dotpart = .1
ifIndex.1 (1.3.6.1.2.1.2.2.1.1.1)
>>iso
  +-org
    +-dod
      +-internet
        +-mgmt
          +-mib-2
            +-interfaces
              +-ifTable
                +-ifEntry
                  +-ifIndex
                    +-1
```

オペランドにifIndex.1を指定した場合。

実行例 2

```
C:\t2eToolKit¥name2dot 1.3.6.1.2.1.2.2.1.1.1
ifIndex.1 (1.3.6.1.2.1.2.2.1.1.1)
>>iso
  +-org
    +-dod
      +-internet
        +-mgmt
          +-mib-2
            +-interfaces
              +-ifTable
                +-ifEntry
                  +-ifIndex
                    +-1
```

オペランドに1.3.6.1.2.1.2.2.1.1.1を指定した場合。

A.4 t2check

形式

t2check 入力ファイル [出力ファイル]

機能

T2E変換定義ファイルのシンタックスチェックを行います。シンタックスチェックの結果にエラーがない場合、テストデータの生成を行います。

シンタックスチェックでは、文法面のみのチェックを行います。定義中に記述されているMIB名等の妥当性チェックは行いません。ただし、特に間違いやすい定義内容については、出力データ中にコメント形式でワーニングメッセージを出力して注意を喚起するようになっています。

出力データは、出力ファイルオペランドを指定しなかった場合、標準出力に出力されます。出力ファイルオペランドを指定した場合、指定したファイルに出力されます。

出力されるテストデータはバッチファイルのデータ形式です。出力データを適当な名前を持つバッチファイル("*.bat")に格納してください。このバッチファイルを起動すると、入力ファイル中のT2E変換定義にヒットするパラメータを含む疑似Trapが生成されます。

また、出力ファイルを明に指定した場合、生成されるバッチファイルにはレグレッションテスト用のテストデータが付加されます。

レグレッションテストの結果確認には、以下のファイルを用います。

ツールインストールディレクトリ¥Doc¥t2regdata.csv

このファイル中のデータは、t2checkコマンドで出力したレグレッションテストデータを使い実際にTrapをSystemwalker Centric Managerに送信した際のシステム監視ログをcsv出力したものです。

テスト結果を同様にcsv出力してこれと比べる事によりレグレッションテストの確認を行う事ができます。

オプション

なし。

オペランド

オペランド	意味
入力ファイル	T2E変換定義ファイル名 (xxx.cnf) を指定します。
出力ファイル	出力ファイル名 (xxx.bat) を指定します。出力ファイルを指定した場合、出力されるバッチファイルデータにレグレッション用のテストデータが付加されます。 省略した場合はバッチファイルデータを標準出力に出力します。

注意事項

なし。

実行例**実行例**

コマンドを実行すると、xxx.cnfの定義ファイルのシンタックスチェックを行い、xxx.batファイルにテスト用のバッチデータを出力する。

```
C:\¥t2eToolKit¥t2check xxx.cnf xxx.bat
```

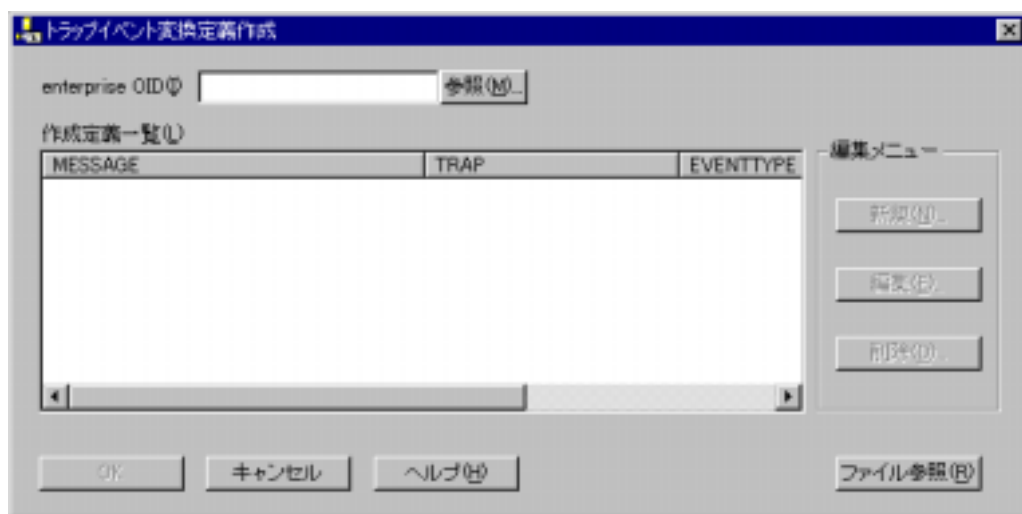
A.5 t2edcnf

形式

t2edcnf

機能

トラップイベント変換定義作成画面を起動します。トラップイベント変換定義作成画面は、T2E変換定義ファイルの作成/編集が可能となります。
トラップイベント変換定義作成画面の初期画面を以下に示します。



enterprise OID入力フィールドにOIDを指定する事により、Enterprise-OIDに対応したT2E変換定義ファイルを作成する事ができます。[ファイル参照]ボタンをクリックし、既存のT2E変換定義ファイルを読み込んで内容を編集する事も可能です。

操作方法の詳細については、ヘルプを参照してください。

オプション

なし。

オペランド

なし。

注意事項

テキストエディタ等で編集したT2E変換定義ファイルは読み込ませないで下さい。

実行例

実行例

コマンドを実行すると、T2E変換定義ファイルの作成を行うためのGUIが起動される。

C:\¥t2eToolKit¥t2edcnf

A.6 疑似トラップ生成ツール

形式

コマンド名 Trap送出先ホスト [Trap送出元ホスト]

機能

t2checkコマンドの出力として生成されるテストツールです。テスト用の疑似Trapを生成します。

オプション

なし。

オペランド

オペランド	意味
Trap送出先ホスト	Trapの送出先ホストを指定します。DNS名またはドット表記のIPアドレスの形で指定します。
Trap送出元ホスト	Trapの送出元ホストを指定します。DNS名またはドット表記のIPアドレスの形で指定します。省略した場合は、host1と同じホストが指定されたものと見なします。

注意事項

本コマンドのコマンド名は、t2checkコマンドの出力データを格納したバッチファイルのファイル名となります。

付録B SNMPプロトコルの概要

B.1 概要

SNMPは、ネットワーク管理ステーション（マネージャ）とネットワークエレメントのエージェント間で管理情報の通信のために使用されるプロトコルです。SNMP単体としては、情報交換のプロトコル規定のみを含み、管理する対象（管理対象 object）についての規定は含まれていません。

しかし、通常 SNMP サポートと言えば、プロトコル本体と標準的なobjectの規定（管理情報ベース MIB-II）の双方を満たしていること及び標準的な管理対象の名付け方式と認証／権限機構を備えていることが最低条件となります（当然のことながら、SNMP マネージャはこの他に、MMI、ロギング等の機能を持ちます）。

ここでは SNMP v1について、プロトコル、MIBオブジェクト（クラス、型、名前）、認証／権限機構、およびMIB定義ファイルの読み方について説明します。

B.2 プロトコル

SNMPプロトコルについて記述します。

B.2.1 下位プロトコル

SNMPは、一般的にUDP/IP上にインプリメントされます。

その理由は、ネットワーク管理というある意味で特殊な処理を行うため。ネットワーク管理機能を使用したい場合とは、トラブルが発生して切りわけを早く行いたいというような緊急事態です。このような時、一般アプリケーションが使用するような信頼性のあるコネクション型の通信は使用出来ない場合があります。ネットワーク管理機能はよりプリミティブな機能を用い、信頼性への問題は自身の処理で解決した方がよいからとされています。

B.2.2 PDU

SNMPのPDUは次の種類があります。

```
get-request
get-next-request
set-request
get-response
trap
```

get	管理対象として規定された情報を取り出すオペレーションです。マネージャは、管理対象の名前を指定して <code>get-request</code> を発行します。エージェント側は、 <code>get-response</code> という応答の形で管理対象の情報を提供します。
get-next	マネージャ側より、エージェント側に存在する管理対象の名前を検索する機能を提供します。マネージャ側は、ある管理対象の名前を指定して <code>get-next-request</code> を発行します。エージェント側は、管理対象の名前を辞書的に配列した並びから、指定された管理対象の次に位置する管理対象の名前とその値を <code>get-response</code> として返します。 <code>get-next</code> 要求を次々に発行することにより、エージェント側に存在する管理対象を全て知ることが可能です。
set-request	管理対象として存在する情報の内容を書き換えるオペレーションです。マネージャは、これを利用してエージェントの制御を行います。マネージャ側は、ある管理対象の名前と値を指定して <code>set-request</code> を発行します。エージェント側は、マネージャ側から指定された管理対象の値を変更します。そして、結果を <code>get-response</code> の形で返します。
trap	エージェント側に何らかの例外事象が発生した場合、エージェント側からマネージャ側に事象を通知する非同期型のメッセージです。 Trapの種類については以下が規定されています。
	<i>cold start</i> エージェントが初期化され起動した。
	<i>warm start</i> エージェントが初期化されず（オブジェクト状態は変更なく）再起動した。
	<i>link down</i> ネットワークとのインタフェースがUPからDOWN状態となった。
	<i>link up</i> ネットワークとのインタフェースがDOWNからUP状態となった。
	<i>authentication failure</i> [コミュニティ（後述）を使用した] 認証違反を検出した。
	<i>egp neighbor loss</i> [GW動作時、EGP に規定された] 近隣ノードとの通信がDOWN状態となった。
	<i>enterprise specific</i> 特殊イベント [プライベートに使用可能]。

上記のTrapのうち、"cold strat" から "egp neighbor loss" までは generic Trap と呼ばれ、SNMPのプロトコル自体に規定されているTrapです。全ての SNMP エージェントは、generic Trapのサポートを期待されています。

これに対して、"enterprise specific" は、specific Trapと呼ばれ、各々の企業（ベンダ）が拡張してプライベートに用いるTrapです。specific Trapには、このTrapを識別するための Specific-Codeが含まれています。

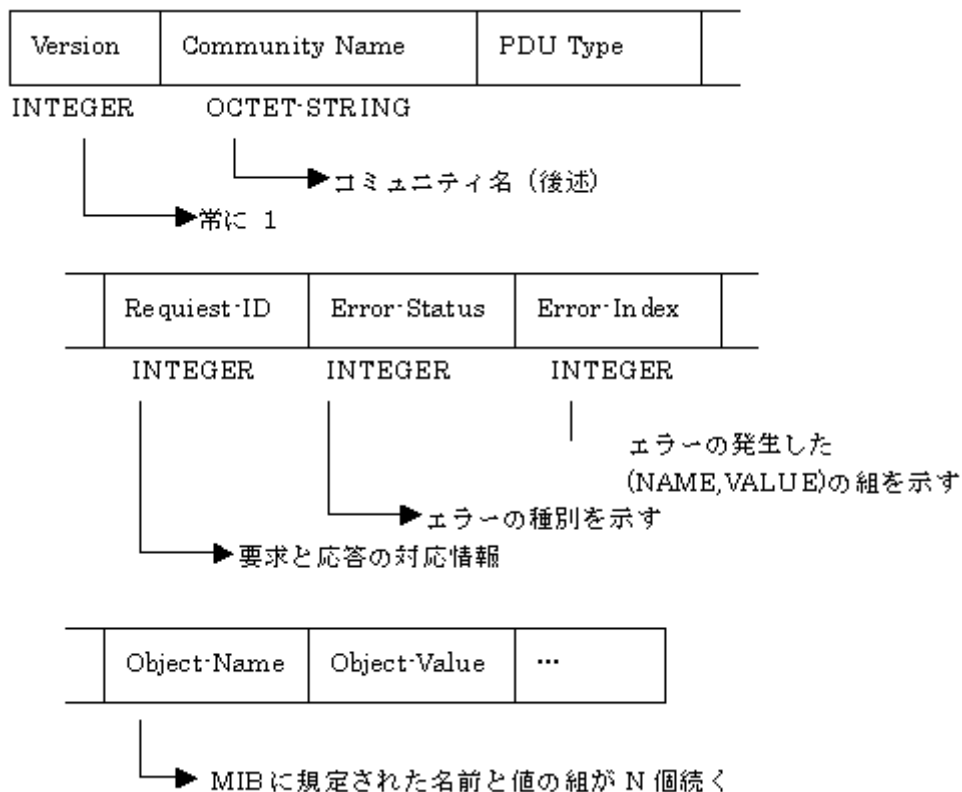
全てのTrapは、その送出責任元を示す ID（Enterprise-0ID）と、発生した非同期事象に関連する管理対象の名前と値の組を含みます。Generic Trapの場合、Enterprise-0IDには責任元はMIB-IIであるというIDが多くの場合設定されます。Specific Trapの場合、Enterprise-0IDには、各企業（ベンダ）に割り当てられた ID が設定されます（多くの場合、このIDは対象機器の機種単位に割り当てられています）。

Specific Trapには、Trapの種別を識別するためのSpecific-Codeが含まれています。Specific-Codeは、Trap送出責任元によって割り振られます。このため、Specific Trapの区別は、一般にEnterprise-0IDとSpecific-Codeの組み合わせにより決定する事ができます。

B.2.3 PDUの形式

SNMPでは、Trap以外のPDUの形式は同一です。PDUは、全てASN.1を用いて規定されています。これらについては、SNMPの標準文書(RFC 1989)を参照してください。ここでは、模式的な説明のみを行います。

a) `get-request`, `get-next-request`, `get-response`, `set-request`



REQUEST-ID は、要求の発行元で割り当てるIDです。応答時には、この要求と同じ値が設定されます。

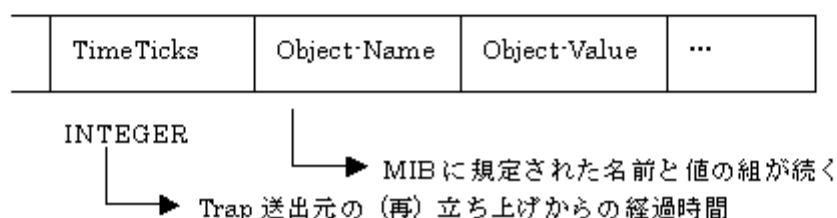
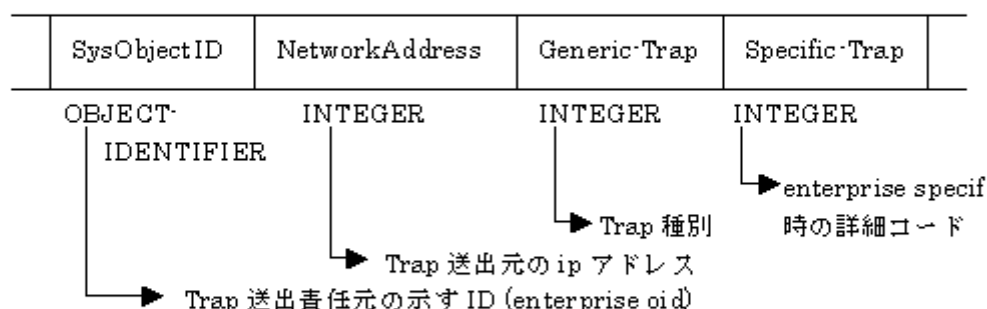
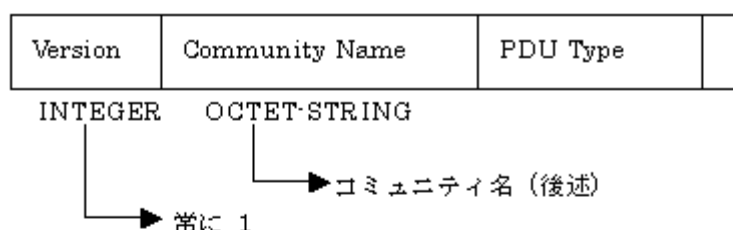
ERROR-STATUS とERROR-INDEX は、get-responseの場合のみ意味を持ちます (他は 0を設定)。ERROR-STATUS の値は以下の通りです。

ERROR-STATUS	値
noError	0
tooBig	1
noSuchName	2
badValue	3
readOnly	4
genError	5

OBJECT-NAME, OBJECT-VALUE の組の意味は、それぞれの要求と応答で以下の通りです。

命令	パラメタ	要求	応答
get	NAME	OBJECT名	OBJECT名
	VALUE	意味なし	そのOBJECTの現在の値
get-next	NAME	OBJECT 名 (又はその一部)	辞書的配列で, request に含まれるNAMEの次ぎに来る
	VALUE	意味なし	そのOBJECTの現在の値
set	NAME	OBJECT名	OBJECT名
	VALUE	そのOBJECTに設定する値	そのOBJECTに設定した値

b) Trap



Trap種別 (generic-Trapフィールド) の値は以下の通りです。

Trap種別	値
cold start	0
warm start	1
link down	2
link up	3
authentication failure	4
egp neighbor loss	5
enterprise specific	6

Trap種別が 0から5の場合、specific-Trapフィールドは意味を持ちません。

Trapには、オブジェクトの名前と値の組が複数付加される事があります。これらの名前の組を VarBind (Variable Binding) と呼びます。

B.2.4 補足事項

下位プロトコルがUDP/IP (信頼性のないコネクションレス通信) であることに注意してください。このことから、以下の事柄が導かれます。

Trapは信用出来ません。なぜならば、Trapメッセージがマネージャに届くかどうかの保証はないからです。このため、SNMPでは、マネージャ側からの定周期のポーリングを行う必要があります。

コネクションレスのため、マネージャ/エージェント間に通信に関する事前定義は、TRAPの宛て先定義を除き必要ありません(実際には、認証/権限機構のための定義は存在します。ただし、デフォルト定義として、どのようなマネージャとでも通信できる定義がエージェント側に存在するようです)。

B.3 MIB

SNMPで使用されるIDやオブジェクトの名前と値の型は、標準的な規定がなされています。ここでは、これらの命名規約について説明します。

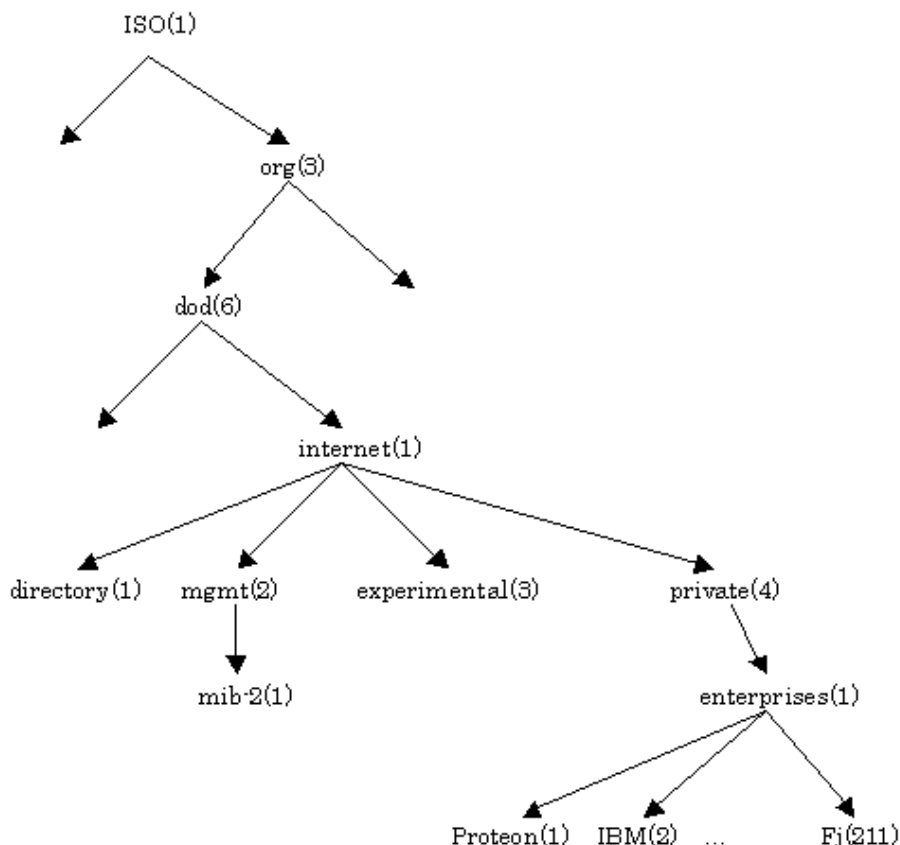
B.3.1 OBJECT IDENTIFIER

MIB 中のオブジェクトの名前は、OBJECT IDENTIFIER で表されます。

OBJECT IDENTIFIER は、ISO/CCITT によりオーソライズした識別子です。OBJECT IDENTIFIER は木構造の体系を持ち、各 node をある特定の組織 / 規定 / 対象に割当てることにより一意性を保っています。

OBJECT IDENTIFIER 自身は、複数のオクテットからなる連続した数値です。各オクテットの値が、木構造のNODEの持つ値に対応しています。

OBJECT IDENTIFIER には、MIB-IIを示す ID や、各企業に割り当てられたIDが存在します。これを木構造の根から辿ると以下の図のようになります。



上図に示すように、たとえばMIB-IIをしめすidは、

```
mib-2 OBJECT IDENTIFIER ::= { 1.3.6.1.2.1 }
```

です。また、富士通の企業IDは

```
Fujitsu OBJECT IDENTIFIER ::= { 1.3.6.1.4.1.211 }
```

と規定されています。

企業固有のOBJECT IDENTIFIERの割り当てを受けた企業は、そのID配下に自身の権限で新たな OBJECT IDENTIFIER を割り当てる事ができます。その OBJECT IDENTIFIERは、企業拡張MIBのオブジェクトクラスの識別子として用いられ、specific Trapのenterprise-oidとして用いられたりします。

また、mib-2配下でも同様な拡張が行われ、RFCに規定されているさまざまな拡張MIBのIDとして用いられています。

B.3.2 クラス、型およびインスタンス

MIB中には管理対象（オブジェクト）のクラスとその型が規定されます。

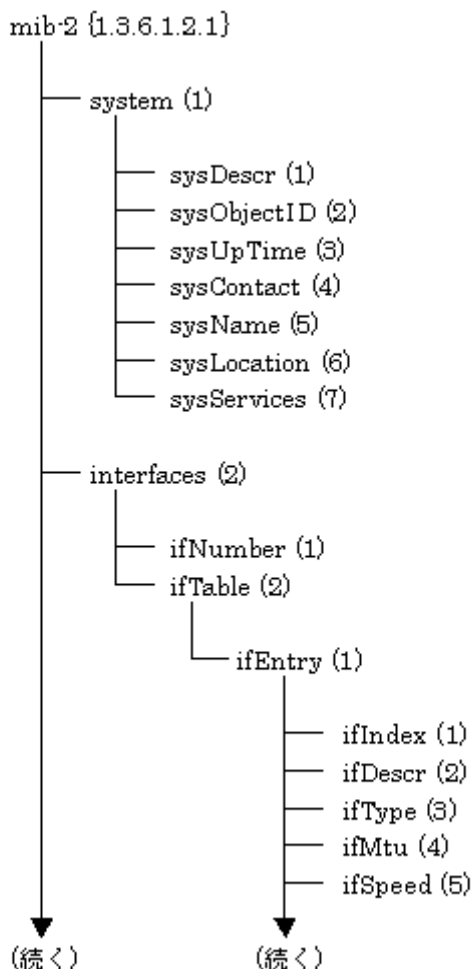
また、実際にgetオペレーションなどでオブジェクトにアクセスする場合は、そのオブジェクトのインスタンスの名前を指定する必要があります。

a) クラス

MIB 中のオブジェクトのクラスは、OBJECT IDENTIFIER で表されます。

クラスの値は、そのMIBを示す OBJECT IDENTIFIER（企業拡張のMIBならばその企業に割り当てられた OBJECT IDENTIFIER 配下に、その企業が新たに割り当てたもの）配下に、やはり OBJECT IDENTIFIER として定義されます。

mib-2 の場合、その配下のObject identifierとして、以下のように幾つかのグループに分けて定義されています。



b) 型

MIBオブジェクトはクラス単位に特定の型をもちます。

MIBオブジェクトの型としては、ASN.1で定義された型の一部とApplicationレベルで意味付けをした特定の型以外は使いません（これは、データ転送方法の単純化のためです）。

ASN.1 定義のうち使用する型は、以下の通りです。

型	意味
INTEGER	整数型
OCTET STRING	16進数
OBJECT IDENTIFIER	OBJECT IDENTIFIER
NULL	型なし

Applicationレベルで規定されている型は、以下の通りです。

型	意味
IpAddress	IPアドレス
NetworkAddress	ネットワークアドレスを表す型。現在は、IPアドレスしか定義されていない。
Counter	非負の整数であり、単調増加して最大値まで行くと0に戻ります。
Gauge	非負の整数であり、最大値と最小値の間で増加減少する。
TimeTicks	非負の整数であり、ある時点（通常はそのシステムの立ち上げ時点）からの10ms単位の時間のカウンタ。
Opaque	任意のエンコードされた型（一種の制限緩和のためのescape機構）

例えば、mib-IIの中のオブジェクトは以下のような型で定義されています。

sysDescr	OCTET STRING 実際の値は、Asciiコードの "システム情報"。
sysObjectID	OBJECT IDENTIFIER 実際の値は、当該システムを識別するOIDの値。
sysUpTime	TimeTicks
sysContact	OCTET STRING 実際の値は、Asciiコードの "連絡先情報"。

c) インスタンス

MIBオブジェクトのインスタンスは、クラスに個々のオブジェクトを示す識別子をサフックスとして付加したID（インスタンス名）を用いて識別します。インスタンス名の名付け方は以下の二種類があります。

ホスト（システム）にただ一つ存在するオブジェクト
これらのオブジェクトのインスタンス名としては、クラスIDに 識別子 "0" を付加したものを uses。たとえば、MIB-IIに規定されているsysDescrやsysObjectID等がこれにあたります。インスタンス名は以下のようになります。

インスタンス	Object ID
sysDescr	sysDescr.0 {1.3.6.1.2.1.1.1.0}
sysObjectID	sysObjectID.0 {1.3.6.1.2.1.1.2.0}

ホスト（システム）に複数存在するオブジェクト
これらのオブジェクトは、MIB中ではテーブルのエントリを構成するように定義されています。これらのオブジェクトのインスタンスの識別子としては、エントリを構成するオブジェクトのなかで値が一意となるオブジェクトの値を用います。たとえば、MIB-IIに規定されているifIndexやifDescr等がこれにあたります。

ます。ifIndexやifDescrは、テーブル ifTable の エントリ ifEntry の構成要素です。ifTable中のオブジェクトのインスタンスの識別子には、ifIndexという整数が用いられます。ifIndexはインタフェースの番号を意味します。

テーブルを構成するオブジェクトのインスタンスを識別する識別子は、整数のみではありません。MIB-II中には、ipアドレスを識別子とするオブジェクトも定義されています。

インスタンス	Object ID
5番目のインタフェースのifDescr	ifDescr.5 {1.3.6.1.2.1.2.2.1.2.5}

B.3.3 MIB の特徴

- 1) MIB は、以下の特徴を持ちます。
- 2) 各nodeに割り当てられる識別オクテットに (0)を用いることはない。
- 3) 木構造の葉にあたるオブジェクトのみ、意味のある値を持つ。それ以外は、オブジェクトの集合を示す識別子となる。
- 4) 意味のあるオブジェクトは、ホスト(システム) にただ一つのオブジェクトと複数存在するオブジェクトとがある。
前者は、単体でオブジェクト名を持つ[例： sysName, sysLocation]。後者は、二次元の表の形で表現される [例： ifTable, atTable]。
- 5) MIB 中には、三次元以上の表は存在しない。

B.4 認証 / 権限機構

SNMPプロトコルを使った通信における認証 / 権限機構としては、SNMPエージェントでローカルに実施するIPアドレスをキーとした機構、およびプロトコル中に規定されたCOMMUNITY NAMEを用いた機構の二種類があります。

a) IPアドレスをキーとした機構

特定のIPアドレスからの get, get-nextのみ受け入れる、特定のIPアドレスから全ての要求を受け入れる、あるいは特定のIPアドレスにのみTrapを送信する等のIPアドレスをキーにして許可する操作を決定する機構です。

b) COMMUNITY NAMEを用いた機構

PDUに含まれるCOMMUNITY NAMEを識別し、特定のCOMMUNITY NAMEを持つ get, get-next要求のみ許可する、あるいは全ての要求を許可するという機構です。

これらの機構は二種類ともSNMPエージェント側で実装されます。エージェントは許可されていないオペレーションを要求された場合、authentication failure Trapを通知します。

SNMPプロトコルは上述の認証 / 権限機構しか持ちません。また、PDUの暗号化も行われていません。このため、マネージャとエージェントの間の通信を盗聴すれば COMMUNITY名も全て知る事ができます。マネージャになりすまして行われるハッキングには対抗できません。

上述の機構のための定義は、例えば、富士通製のルータであるLRではSNMPマネージャの定義として行われます。SNMPマネージャの定義では、以下の定義情報を設定します。

	定義
IPアドレス	SNMPマネージャのIPアドレス。ワイルドカード指定可。
COMMUNITY名	使用されるコミュニティ名。
SET	ACCEPT or REJECT。SET要求を受け入れるか否か。
TRAP	ON or OFF。TRAPを送出するか否か。

一台のLRで上記の定義を複数行う事ができます。これらの定義を組み合わせる事によりたとえば以下のような運用ができます。

一般のマネージャからは、get系の要求のみしか受け入れない。
特定のマネージャからは、全ての要求を受け入れ、Trapも通知する。

B.5 MIB定義ファイルの読み方

ここでは、MIB定義ファイルの読み方を例を挙げて解説します。以降が解説のコメントです。

a) 全体例 (mib-2 RFC1213 より、全体は同梱の"rfc1213-mib11.txt"参照)

```
-- "--で始まる行はコメントです。
-- MIB定義の始まりを示します。
RFC1213-MIB DEFINITIONS ::= BEGIN

-- 他の定義ファイルから型やオブジェクトIDをインポートしています。
IMPORTS
    mgmt, NetworkAddress, IpAddress, Counter, Gauge,
        TimeTicks
    FROM RFC1155-SMI
    OBJECT-TYPE
        FROM RFC-1212;

-- This MIB module uses the extended OBJECT-TYPE macro as
-- defined in [14];
-- MIB-II (same prefix as MIB-I)

-- MIB2 OBJECT IDの定義です。mgmt.1であることを示しています。
mib-2      OBJECT IDENTIFIER ::= { mgmt 1 }
-- textual conventions

DisplayString ::=
    OCTET STRING
-- This data type is used to model textual information taken
-- from the NVT ASCII character set. By convention, objects
-- with this syntax are declared as having
--
--     SIZE (0..255)

-- この定義ファイルの中で使用される型(DisplayStringとPhysAddress)の定義を行なっています。
-- ここでは両方ともOCTET STRINGから派生した型であるとして、コメントで特徴を述べています。
PhysAddress ::=
    OCTET STRING
-- This data type is used to model media addresses. For many
-- types of media, this will be in a binary representation.
-- For example, an ethernet address would be represented as
-- a string of 6 octets.
```

```

-- この定義ファイルの中で定義されるオブジェクトをグループ分けするため、
-- mib-2配下にOBJECT IDENTIFIERを定義しています。
-- groups in MIB-II
system      OBJECT IDENTIFIER ::= { mib-2 1 }
interfaces  OBJECT IDENTIFIER ::= { mib-2 2 }
at          OBJECT IDENTIFIER ::= { mib-2 3 }
ip          OBJECT IDENTIFIER ::= { mib-2 4 }
icmp        OBJECT IDENTIFIER ::= { mib-2 5 }
tcp         OBJECT IDENTIFIER ::= { mib-2 6 }
udp         OBJECT IDENTIFIER ::= { mib-2 7 }
egp         OBJECT IDENTIFIER ::= { mib-2 8 }

-- historical (some say hysterical)
-- cmot      OBJECT IDENTIFIER ::= { mib-2 9 }

transmission OBJECT IDENTIFIER ::= { mib-2 10 }
SNMP         OBJECT IDENTIFIER ::= { mib-2 11 }

-- 以下、systemグループのオブジェクト定義です。
-- the System group

-- Implementation of the System group is mandatory for all
-- systems. If an agent is not configured to have a value
-- for any of these variables, a string of length 0 is
-- returned.

-- キーワードOBJECT-TYPEはオブジェクト定義の始まりを示します。
-- キーワードの前の単語は、オブジェクト名です。
sysDescr OBJECT-TYPE
-- オブジェクトの型を示します。ここでは個々のオクテットのサイズも規定されています。
-- その他に実際に取り得る値が規定されている場合もあります。
    SYNTAX DisplayString (SIZE (0..255))
-- アクセスレベルが定義されています。以下のいずれかが指定されます。
--   - read-only: 読み込みのみ可
--   - read-write: 読み書き可
--   - write-only: 書き込み可
--   - not-accessible: アクセス不可
    ACCESS read-only
-- 実装状態が定義されています。以下のいずれかが指定されます。
--   - mandatory: 必須
--   - optional: 選択
--   - obsolete: 廃止
    STATUS mandatory
-- 説明文です。
    DESCRIPTION
        "A textual description of the entity. This value
        should include the full name and version
        identification of the system's hardware type,

```

```
software operating-system, and networking
software. It is mandatory that this only contain
printable ASCII characters."
-- OIDが定義されています。
-- ここでは、system.1が指定されています。これがこのオブジェクトのクラスIDとなります。
::= { system 1 }

sysObjectID OBJECT-TYPE
    SYNTAX  OBJECT IDENTIFIER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The vendor's authoritative identification of the
        network management subsystem contained in the
        entity. This value is allocated within the SMI
        enterprises subtree (1.3.6.1.4.1) and provides an
        easy and unambiguous means for determining `what
        kind of box' is being managed. For example, if
        vendor `Flintstones, Inc.' was assigned the
        subtree 1.3.6.1.4.1.4242, it could assign the
        identifier 1.3.6.1.4.1.4242.1.1 to its `Fred
        Router'."
    ::= { system 2 }

-- ~ ~ ~ 一部省略 ~ ~ ~

-- the Interfaces group

-- Implementation of the Interfaces group is mandatory for
-- all systems.

ifNumber OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "The number of network interfaces (regardless of
        their current state) present on this system."
    ::= { interfaces 1 }

-- 以下、テーブルの定義が始まります。
-- the Interfaces table

-- The Interfaces table contains information on the entity's
-- interfaces. Each interface is thought of as being
-- attached to a `subnetwork'. Note that this term should
-- not be confused with `subnet' which refers to an
-- addressing partitioning scheme used in the Internet suite
```

```
-- of protocols.
```

```
-- ifTableというテーブルの型の定義です。
```

```
-- ifTableは、ifEntryの並びであると定義されています。
```

```
ifTable OBJECT-TYPE
    SYNTAX      SEQUENCE OF ifEntry
    ACCESS      not-accessible
    STATUS      mandatory
    DESCRIPTION
        "A list of interface entries. The number of
         entries is given by the value of ifNumber."
    ::= { interfaces 2 }
```

```
-- テーブル中のエントリの定義です。エントリの型をifEntryと定義しています。
```

```
ifEntry OBJECT-TYPE
    SYNTAX      ifEntry
    ACCESS      not-accessible
    STATUS      mandatory
    DESCRIPTION
        "An interface entry containing objects at the
         subnetwork layer and below for a particular
         interface."
    INDEX       { ifIndex }
    ::= { ifTable 1 }
```

```
-- ifEntryの定義です。ifIndex等のオブジェクトの並びであると定義されています。
```

```
ifEntry ::=
    SEQUENCE {
        ifIndex
            INTEGER,
        ifDescr
            DisplayString,
        ifType
            INTEGER,
        ifMtu
            INTEGER,
        ifSpeed
            Gauge,
        ifPhysAddress
            PhysAddress,
        ifAdminStatus
            INTEGER,
        ifOperStatus
            INTEGER,
        ifLastChange
            TimeTicks,
        ifInOctets
            Counter,
        ifInUcastPkts
```



```
        Counter,
    ifInNUcastPkts
        Counter,
    ifInDiscards
        Counter,
    ifInErrors
        Counter,
    ifInUnknownProtos
        Counter,
    ifOutOctets
        Counter,
    ifOutUcastPkts
        Counter,
    ifOutNUcastPkts
        Counter,
    ifOutDiscards
        Counter,
    ifOutErrors
        Counter,
    ifOutQLen
        Gauge,
    ifSpecific
        OBJECT IDENTIFIER
}
```

-- テーブル中のエントリの定義です。

```
ifIndex OBJECT-TYPE
    SYNTAX  INTEGER
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "A unique value for each interface. Its value
        ranges between 1 and the value of ifNumber. The
        value for each interface must remain constant at
        least from one re-initialization of the entity's
        network management system to the next re-
        initialization."
    ::= { ifEntry 1 }
```



```
ifDescr OBJECT-TYPE
    SYNTAX  DisplayString (SIZE (0..255))
    ACCESS  read-only
    STATUS  mandatory
    DESCRIPTION
        "A textual string containing information about the
        interface. This string should include the name of
        the manufacturer, the product name and the version"
```

```

        of the hardware interface."
 ::= { ifEntry 2 }

-- ~~~ 一部省略 ~~~
-- MIB定義の終わりを示しています。
-- BEGINと対応しています。
END

```

b) Trap定義 (rmon-mib RFC1757 より. 全体は同梱の"rfc1757.my"参照)

```

-- キーワードTRAP-TYPEはTrap定義の始まりを示します。
-- TRAP-TYPEの前にある文字列がTrap名です。
risingAlarm TRAP-TYPE
-- Enterprise-0IDを定義しています。ここで指定されたrmonという名前は
-- 別のところでOBJECT IDENTIFIERとして定義されています。
    ENTERPRISE rmon
-- Trapに付加されるオブジェクト(Varbind)を定義します。
-- ここでは、合計5個のVarbindが指定されています。
-- VARIABLESは省略されることがあります。
    VARIABLES { alarmIndex, alarmVariable, alarmSampleType,
        alarmValue, alarmRisingThreshold }
-- Trapについての説明です。
    DESCRIPTION
        "The SNMP Trap that is generated when an alarm
        entry crosses its rising threshold and generates
        an event that is configured for sending SNMP
        Traps."
-- 最後にspecific codeが定義されています。
    ::= 1

```

付録C T2E変換機能の概要

Systemwalker Centric Managerは、受信したTrapをEventメッセージの形に変換して、監視イベント一覧画面に表示します。

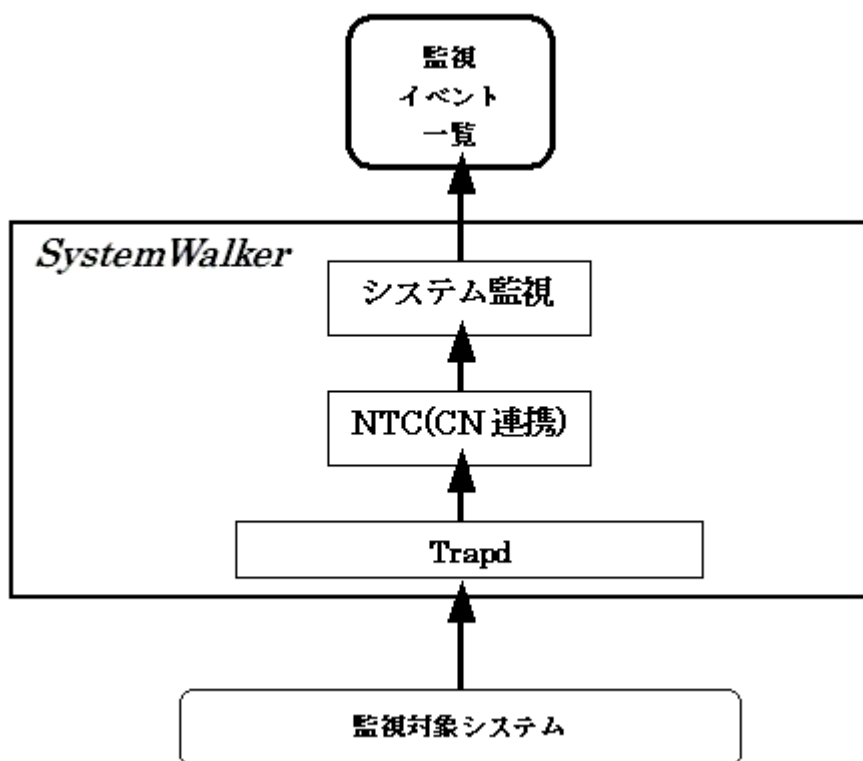
この文書では、Systemwalker Centric ManagerのT2E変換機能の概要とT2E変換定義の適用機能について説明します。

C.1 T2E変換

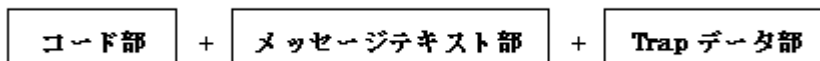
Systemwalker Centric Managerでは、トラップデーモン (Trapd)と呼ばれるコンポーネントが、監視対象システム (サーバや通信機器) から送られてくるTrapを最初に受信します。

Trapdは、受信したTrapをNTC(Network Trap Converterの略、V4.xまではCN連携)と呼ばれるコンポーネントに通知します。

NTCは、Trapdから受け取ったTrapをEventメッセージの形に変換し、システム監視機能に通知します。この結果、Trapを監視イベント一覧画面で監視する事が可能となります。



変換後のEventメッセージは以下のような形をしています。



例えば、RepeaterMIB (RFC1368により規定された拡張MIB) に定義された rptrHealth-Trapを受信した場合、Systemwalker Centric Managerは以下のようなメッセージを表示します。

コード部

: "AP:MpCNaapl: ERROR: 106: "

メッセージテキスト部

: " ネットワークで事象が発生しました. "

Trap データ部

:

```

"(TRAP      agent:10.34.195.134      community:public      generic:6
enterprise:mib-2.22
specific:1 timestamp:1234566 varbind:(mib-2.22.1.1.2 [1 1 0] 3))"

```

コード部には、Eventメッセージの識別情報が表示されます。この例では、以下のような意味になります。

AP	アプリケーションレベルのイベントである事を示します。Solaris OE版 / Linux版の場合は、UXとなります。
MpCNaapl	NTCがシステム監視機能に通知した事示します。監視イベント種別は、" ネットワーク " になります。なお、T2Eツールキットで作成したT2E変換定義ファイルで出力したTrapでは"MpApLink"、監視イベント種別は " アプリ連携 " になります。
ERROR	イベントの重要度を示します。ERRORの他にWARNINGが存在します。V4.x系では、" エラー " などの日本語表記でしたが、V5.x系から英語版との統一を図りました
106	イベント番号を示す。TrapのGenericCode(0~6)+100の番号になります。V4.x系では、3 固定となります。

メッセージテキスト部の文字列には、受信したTrapの種別により予め規定された文字列が設定されます。上記の場合、Enterprise-Trapのデフォルトメッセージテキストである " ネットワークで事象が発生しました. " が設定されました。(V4.x系では、" ネットワークで障害が発生しました. ") Generic-Trapに対応するデフォルトのメッセージテキストとしては、以下の文字列が予め設定されています。

Trap種別	メッセージテキスト
cold start	ネットワークで"ColdStart"が発生しました。
warm start	ネットワークで"WarmStart"が発生しました。
link down	ネットワークで"LinkDown"が発生しました。(IF Index.[番号])"
link up	ネットワークで"LinkUp"が発生しました。(IF Index.[番号])"
authentication failure	ネットワークで"AuthenticationFailure"が発生しました。
egp neighbor loss	ネットワークで"EgpNeighborLoss"が発生しました。

Trapデータ部のパラメタの意味は次の通りです。

TRAP	Trapである事を示します。
agent	Trap通知元のエージェントのIPアドレスを示します。ここでは、10.34.195.134のIPアドレスを持つノードからのTrapである事がわかります。
community	Trap通知に用いられたcommunity名を示します。
generic	受信したTrapの Generic-Code(Trap種別)を示します。ここでは、"6" Enterprise-Specific Trapである事が分かります。
enterprise	受信したTrapの Enterprise-0IDを示します。ここでは、mib-2配下の22番の0IDである事が分かります。
specific	受信したTrapの Specific-Codeを示します。
timestamp	受信したTrapのTimestampフィールドに設定された値を表示します。
varbind	受信したTrapに設定されたVarBind情報を表示します。複数のVarBind情報が存在する場合は、複数繰り返して表示します。

Trapデータ部の VarBind 情報の表示形式は以下の通りです。

(オブジェクト名 [X Y -] 値)	
オブジェクト名	MIBオブジェクトの識別情報が、名前表現またはドット表記の形で表示されます。
X	オブジェクトの型を示す以下のコードが表示されます。
	Integer 1
	Octet String 2
	Object Identifier 3
	Null 4
	IPAddress 5
	Counter 6
	Gauge 7
	Timeticks 8
	Opaque 9
Y	オブジェクトの大きさがByte単位で表示されます。
値	オブジェクトの値を示します。

前述の例 “ (mib-2.22.1.1.2 [1 1 0] 1) “ で説明すると、VarBind情報として "mib-2.22.1.1.2"というMIB名、整数型、長さ 1byte、および値 "1" のオブジェクトが設定されていた事がわかります。

C.2 MIB拡張操作とT2E変換機能

MIB拡張操作を行うと、拡張MIBに含まれるObjectIDの定義（OIDの値と名前）が、Systemwalker Centric Managerに登録されます。拡張MIB登録後は、ドット形式ではなく名前を使ったMIBの操作が可能になります。

MIB拡張操作は、T2E変換機能の処理結果（つまり変換後のEventメッセージ）にも影響を与えます。

例えば、前項で説明した `rpTrHealth-Trap` 受信の結果表示されるメッセージは、MIB拡張操作前と後で以下のように変わります。

MIB拡張操作前	AP:MpCNAppl: ERROR: 106:ネットワークで事象が発生しました . (TRAP agent:10.34.195.134 community:public generic:6 enterprise: <u>mib-2.22</u> specific:1 timestamp:1234566 varbind:(<u>mib-2.22.1.1.2</u> [1 1 0] 3))
MIB拡張操作後	AP:MpCNAppl: ERROR: 106 :ネットワークで事象が発生しました . (TRAP agent:10.34.195.134 community:public generic:6 enterprise: <u>snmpDot3RptrMgt</u> specific:1 timestamp:1234566 varbind:(<u>rpTrOperStatus</u> [1 1 0] 3))

上記のように、拡張前はドット形式で表示されていたTrapデータ部の情報（Enterprises-OIDとVarBind情報のMIBオブジェクト識別）が、拡張後は名前形式で表示されるようになります。

このように、拡張MIBに登録すれば、Trapデータ部に表示される情報を、より人間に分かりやすい形にすることができます。

C.3 メッセージのカスタマイズ

NTCでは、T2E変換定義ファイルと呼ぶファイルに格納された変換方法にしたがってTrapをEventメッセージに変換します。

T2E変換定義ファイルを拡張することにより、Trap受信により表示されるEventメッセージのメッセージテキスト部とコード部の一部をカスタマイズする事が可能です。

カスタマイズを行うため追加するT2E変換定義ファイルの形式は、以下の通りです。

T2E変換定義ファイルのコード系はSJISである。Solaris OE / Linux上のSystemwalker Centric Managerに適用する場合も、SJISコードで適用する。

T2E変換定義ファイルは、Enterprise-01D単位に作成されている。

ファイルの先頭レコードには、以下のファイル制御レコードが存在する。

"# enterprises=(ドット形式のEnterprise-01D値)"

ファイル制御レコード以降は、変換定義エントリが続く。

先頭が # で始まるレコードはコメントとして無視される。

変換定義エントリは、次のような形式をしています。

以下の変更定義エントリ制御レコードで始まる。

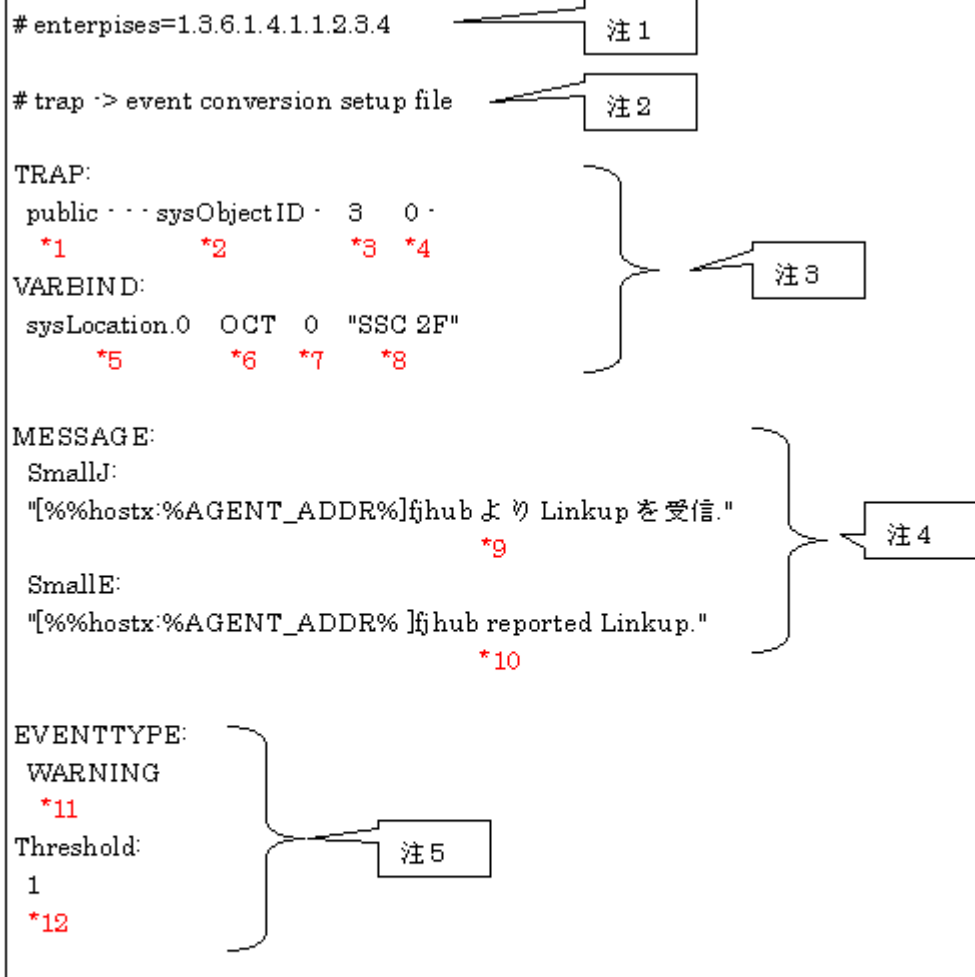
"# Trap -> event conversion setup file"

変換対象Trapを識別する情報

メッセージ変換定義

重要度変換定義

T2E変換定義ファイルの例を以下に示します。



- 注 1 : ファイル制御レコード
- 注 2 : 変更定義エントリ 制御レコード
- 注 3 : 変換対象 Trap 識別情報
- 注 4 : メッセージ変換定義
- 注 5 : 重要度変換定義

“ *数字 ” で示した項目については次の表に示します。

1	コミュニティ名： 文字列またはワイルドカード("-") を指定。
2	EnterpriseOID： Enterprise-OIDをドット記法で指定。
3	Generic-Code： 以下のいずれかを指定。 0: Cold_start 1: Warm_start 2: Linkdown 3: Linkup 4: Authenticationfailure 5: Egpneighborloss 6: EnterpriseTrap -: 全てのTrapにヒット
4	Specific-Code： Generic-Codeが6の場合、数字（例 1、125、etc）またはワイルドカード("-") を指定。Generic-Codeが6以外の場合は、"- " を必ず指定する。
5	Varbind名： ドット記法またはオブジェクト名を指定する。ワイルドカード("-")指定の場合、Varbindの名前、タイプ、属性、値（6～8番）全てに - を指定する。
6	Varbindタイプ： 以下のいずれかを指定。 INT: Integer OCT: Octet string OID: Object Identifier IPA: IPaddress CNT: Counter GAG: Gauge TIM: TimeTicks OPA: Opaque
7	Varbind値属性： VarbindタイプにOctet String又はOpaqueを指定した場合、Varbind値の属性として0(: ASCII)をかならず指定する。
8	Varbind値： 以下のようにVarbindタイプにより指定方法が異なる。 INT: 数字（例 1、125、etc） OCT: 文字列 OID: ドット記法またはオブジェクト名 IPA: ドット記法またはDNS名 CNT: 数字 GAG: 数字 TIM: 数字 OPA: 文字列

9

日本語メッセージ文：

英語メッセージ文：

日本語以外のOSで動作する場合には英語メッセージが出力される。

〔メッセージ文中での変数〕

メッセージ文中で以下の変数を使用すると、Trapの内容を表示することができる。

```

%COMMUNITY% ..... community
%ERROR_STATUS% ..... error-status
%ERROR_INDEX% ..... error-index
%ENTERPRISE% ..... enterprise
%AGENT_ADDR% ..... agent-addr
%GENERIC_TRAP% ..... generic-trap
%SPECIFIC_TRAP% ..... specific-trap
%TIME_STAMP% ..... time-stamp
%TRAPD_HOST% ..... trapd-host
%VARBIND_NAME% ..... varbind-name
%VARBIND_TYPE% ..... varbind-type
%VARBIND_VALUE% ..... varbind-value
%VARBIND_LENGTH% ..... varbind-value length(実際のデータ長)
%VARBIND_VIEW_LENGTH% ... varbind-value display length(ディスプレイ上での
長さ)
%VARBIND_BIN_FLAG% ..... varbind-value binary flag
%NEXT_V% ..... next varbind
(指定後、これ以降次のVarbindの内容を参照できる)
%BACK_V% ..... backward varbind
(指定後、これ以降一つ前のVarbindの内容を参照できる)

```

〔メッセージ文の記述〕

[%hostx:%AGENT_ADDR%] は 必 ず 記 述 す る .
 この記述は " システム監視 " にTrap送信先ノードのIPアドレスを通知するために必要 .

メッセージ文は" "で囲み、1行で記述する .

メッセージ文中に%を指定したい場合は、%と記述する .

Varbindは1番目から順番に表示し、%NEXT_V%が指定されると2番目のVarbindの表示が可能になる . もし、2番目がない場合は1番目が表示される .

2番目のVarbindを表示しているときに、%BACK_V%を指定すると、その直後から1番目のVarbindの表示が可能になる . 1番目のときに指定しても1番目のままである .

%NEXT_V%、%BACK_V%はメッセージ文では" "に置き換えられる .

%NEXT_V%、%BACK_V%を使用するときは、%が2つ重ならないように注意する .

例)

Value: %VARBIND_VALUE% %NEXT_V%Vname: %VARBIND_NAME%
 1 番目のvalue この空白は重要 2 番目のname

1 0	英語メッセージ文： Systemwalker Centric Managerが日本語以外のOSで動作する場合には英語メッセージが出力される。メッセージの記述方法は日本語メッセージと同様。
1 1	通知イベントタイプ： 以下のいずれかを指定する。 ERROR：エラー ... 監視イベント一覧に重要と表示される。 WARNING：警告 ... 監視イベント一覧に警告と表示される。 INFORMATION：情報 ... 監視イベント一覧には表示されない。
1 2	しきい値： 以下のいずれかを指定する。（ただし，V5.0L10,V5.0L20では使用不可） 0：イベント発行せず 1：イベント発行する（推奨）

C.4 変換メッセージの決定論理

NTCでは、TrapをEventメッセージに変換する際、以下の論理で変換定義の検索を行います。

- 1) TrapのEnterprisesOIDを参照し、OIDが一致するT2E変換定義ファイルを検索する。見つからない場合は、4)へ。
- 2) T2E変換定義ファイルの変換定義エントリを先頭から順に検索する。
- 3) 受け取ったTrap内に設定されている情報がTrap識別情報とマッチすれば、そのエントリの変換定義情報に従いEventメッセージを作成する。7)へ
- 4) デフォルトのT2E変換定義ファイルを検索。
- 5) デフォルトT2E変換定義ファイルの変換定義エントリを先頭から順に検索する。
- 6) 受け取ったTrap内に設定されている情報がTrap識別情報とマッチすれば、そのエントリの変換定義情報に従いEventメッセージを作成する。
- 7) システム監視機能にEventメッセージを通知する。

デフォルトのT2E変換定義ファイルは、Systemwalker Centric Managerインストール時にシステムに導入されます。このファイルを変更する事はできません。また、Systemwalker Centric ManagerのVLによって変更される事がありますが、他の定義に影響がでるようなことはありません。

Enterprise-OID単位のT2E変換定義ファイルは、T2Eツールキットを用いて作成し、システムに適用する事ができます。

T2E変換定義ファイルの作成に当たっては、変換定義エントリが先頭から順に検索される事に注目してください。この処理論理を活用すれば、ワイルドカード指定と組み合わせる事により、より柔軟なメッセージ変換方法を指定できます。

T2E変換定義ファイルの追加は、Enterprise-OID単位に行われ検索もまずファイル単位におこなわれます。このため、T2E変換定義ファイルの数そのものは、T2E変換処理の性能に大きな影響を与える事はありません。

C.5 T2E変換定義ファイルの適用

NTCは、各部門管理サーバで動作します。このため、T2E変換定義は運用管理サーバと全ての部門管理サーバに必要となります。Systemwalker Centric Managerでは、T2E変換定義をポリシー情報として扱う事により、全管理サーバへの情報配付を行います。

具体的な手順は、以下の通りです。

- 1) T2E変換定義ファイルを作成します。T2Eツールキットで作成したT2E変換定義ファイルは、Windowsのテキスト形式(SJIS、LR+LF)です。Solaris OE版 / Linux版の運用管理サーバで適用する場合も、同じファイルを使用してください。漢字コードや改行を変換する必要はありません。
- 2) 作成した変換定義ファイルを適用コマンド (CNSetCnfMg.exeコマンド)を運用管理サーバ上で実行し、ポリシーに適用します。
- 3) ポリシー情報の配付を行います。

CNSetCnfMg.exeコマンドによる変換定義ファイルの適用では、追加モードと置換モードを選択する事ができます。

置換モード

既存の同一OIDの変換定義ファイルを新たに適用する変換定義ファイルで置き換える。



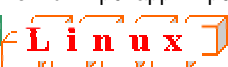
追加モード

新たに適用する変換定義ファイル内の変換定義エントリを既存の同一OIDの変換定義ファイルの末尾に追加する。

追加モードを用いる事により、既存の変換定義に手を加える事なく新たな変換定義を追加する事が可能になります。

CNSetCnfMg.exeコマンドは、Systemwalker Centric Managerにインストールされています。存在するディレクトリとコマンド仕様は以下の通りです。T2EツールキットのCNSetCnfMg.batとはパラメタが異なりますので注意してください。

ディレクトリ





 [install-dir]¥MpwWalker.dm¥MpcNappl¥MpcNmgr¥bin
 
 /opt/FJSVfwntc/MpcNmgr/bin

コマンド仕様

```
CNSetCnfMg.exe -f 配置対象のファイルのフルパス { -a | -c }
    -f 配置対象のファイルのフルパス
        適用する定義ファイルをフルパスで指定する。
    -a
        追加モードで定義ファイルを適用する。
    -c
        上書きモードで定義ファイルを適用する。
    復帰値
```

0 : 正常終了
0 以外 : 異常終了

変換定義情報を何時有効とするかは、ポリシー配付時の指定に従います。
ポリシー配付の結果、適用したT2E変換定義ファイルの情報は、運用管理サーバおよび全部門管理サーバの以下のディレクトリに格納されます。

 Systemwalker インストール `dir¥MplWalker.dm¥MplNappI¥MplNagt¥etc¥master`
  `/opt/FJSVfwntc/MplNagt/etc/master`

情報は、“B” + 0IDのドット表記 *+ “.cnf” という名前のファイルに Enterprise-0ID単位に格納されます。このファイルおよびその内容から、現状のT2E変換定義の適用状況を知る事ができます。

なお、適用したT2E変換定義を元に戻すのは以下のようにします。

既にT2E変換定義が適用されていた場合、元の変換定義ファイルを置換モードで再適用します。

あらたにT2E変換定義を適用した場合、ファイル制御レコード [“# enterprises=(ドット形式のEnterprise-0ID値)”] のみが存在するT2E変換定義ファイルを置換モードで適用します。

CNSetCnfMg.exeで出力される主なエラーメッセージは以下のとおりです。

現象	原因	対処
err:CNSetCnfMg:Failed to prepare file. code = 3 LastError = 86031	指定したT2E変換定義ファイルが存在しない。	ファイル名やパスを確認する。なお、T2E変換定義ファイルは絶対パスで指定する。
err:CNSetCnfMg:Failed to prepare file. code = 3 LastError = 86032	本コマンドを実行するユーザにシステム権限がない(=スーパーユーザでない)。	WindowsではAdministrator、Solaris OE / Linuxではスーパーユーザで実行する。
err:CNSetCnfMg:Failed to prepare file. code = 3 LastError = 69639	T2E変換定義ファイルとして正しくないファイル(もしくはディレクトリ)を指定している。または、T2E変換定義ファイルの先頭に、# enterprises=1.3.6... がない。	T2E変換定義ファイルを修正する。

