



BOM for Windows Ver.6.0
カスタム監視 ガイドライン

免責事項

本書に記載された情報は、予告無しに変更される場合があります。セイ・テクノロジーズ株式会社は、本書に関していかなる種類の保証（商用性および特定の目的への適合性の黙示の保証を含みますが、これに限定されません）もいたしません。

セイ・テクノロジーズ株式会社は、本書に含まれた誤謬に関しての責任や、本書の提供、履行および使用に関して偶発的または間接的に起こる損害に対して、責任を負わないものとします。

著作権

本書のいかなる部分も、セイ・テクノロジーズ株式会社からの文書による事前の許可なしには、形態または手段を問わず決して複製・配布してはなりません。

Copyright © 2013-2014 SAY Technologies, Inc. All rights reserved.

本ガイドラインに記載されている BOM はセイ・テクノロジーズ株式会社の登録商標です。Microsoft, Windows は、米国 Microsoft Corporation の米国及びその他の国における登録商標です。その他会社名、製品名およびサービス名は各社の商標または登録商標です。

なお、本文および図表中では、「™ (Trademark)」、「® (Registered Trademark)」は明記しておりません。

■ 目次

本ガイドラインについて	1
製品表記	1
使用方法	1
表記規則	1
環境説明	1
第 1 章 カスタム監視とは	2
第 2 章 監視項目の作成・設定	3
2.1 カスタム監視の作成	3
2.2 カスタム監視の設定	4
第 3 章 外部コマンドの実例	6
3.1 コンソールアプリケーションの場合	6
3.1.1 コンソールアプリケーション作成時の指針	6
3.1.2 コンソールアプリケーションのラッピング	6
3.2 バッチファイル(コマンドラインスクリプト)の場合	8
3.2.1 バッチファイル作成時の指針	8
3.2.2 バッチファイル実行時の挙動	9
3.3 WSH(Windows Script Host)の場合	10
3.3.1 WSH スクリプト作成時の指針	10
3.3.2 WSH スクリプト実行時の挙動	11
3.4 Windows PowerShell の場合	13
3.4.1 PowerShell によるカスタム監視の設定例	13
3.4.2 PowerShell 実行時の挙動	14
第 4 章 FAQ	15

本ガイドラインについて

製品表記

本ガイドラインでは、以下の製品について略称を使用しております。

正式名称	本マニュアルでの呼称(略称)
BOM for Windows Ver.6.0	BOM 6.0

使用方法

本ガイドラインには、BOM 6.0 で作成可能な監視項目のひとつである“カスタム監視”を使用する際に必要となる、詳細な情報と手順が記載されています。

本ガイドラインを使用するには、Microsoft Windows オペレーティングシステムについての実際的な知識、BOM for Windows Ver.6.0 の基本的な知識、使用するプログラム・スクリプト言語の実際的な知識が必要です。

本ガイドラインに掲載されているサンプルスクリプトは、カスタム監視の挙動を解説するためのものです。実際の監視項目として使用するには適さない場合もありますのでご注意ください。

表記規則

本ガイドラインでは、以下の表記規則を使用しています。

表記	説明
‘参照先’	シングルクォート内(‘ ’)は本マニュアル内、あるいは別のマニュアルの参照を示します。
“ラベル”	ダブルクォート内(“ ”)はラベル名を示します。
「タブ」	鉤括弧内(「 」)はプロパティシート等でのタブ名を示します。
[ボタン]	角括弧内([])はボタン名を示します。
<キー>	山括弧(不等号記号)内(< >)はキーボード入力を示します。
(補足説明)	丸括弧内(「(」と「) 」)は補足説明を示します。

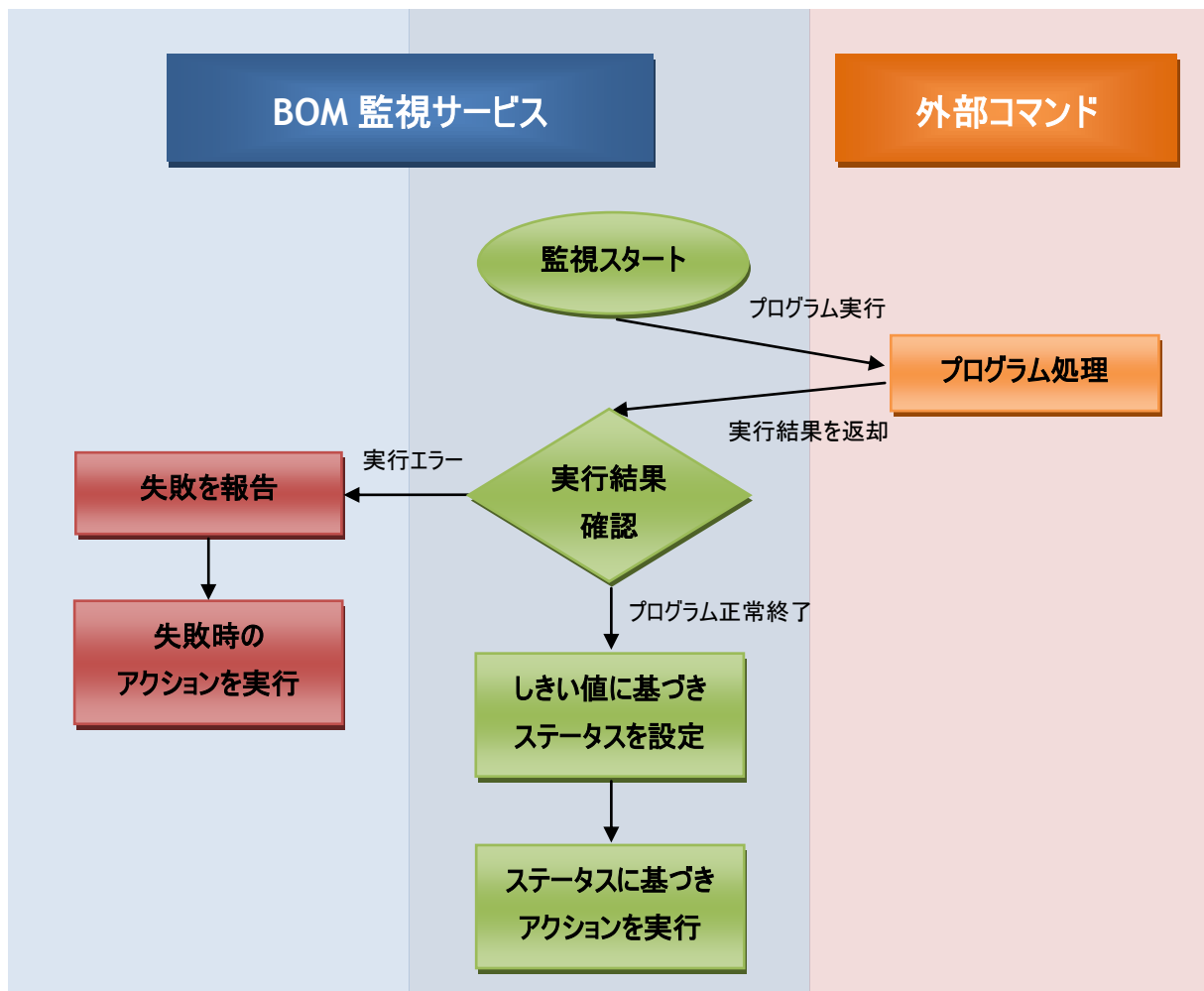
環境説明

本ガイドラインでは、コンピューターの操作には Windows Server 2008 R2 と Windows Server 2012 R2 の画面を主に使用しております。お使いの OS によっては、操作方法が若干異なる場合がございますが、適宜読み替えてお使いください。

第1章 カスタム監視とは

BOM 6.0 のカスタム監視は、コンソールアプリケーションや OS のスクリプトであるバッチファイル(コマンドラインスクリプト)、WSH (Windows Script Host)、Windows PowerShell といった外部コマンドを使用して、任意の監視を行う監視項目です。

カスタム監視を使用することで、BOM 6.0 に標準では搭載されていない監視機能を作成し、監視することができます。



カスタム監視は、監視用にご用意いただいた外部コマンドを BOM 監視サービスから実行し、外部コマンドの実行結果を元に通知・リカバリー等のアクションを実行します。

本ガイドラインでは、カスタム監視の使用方法与外部コマンドの作成指針をご案内いたします。

※ コンソールアプリケーションやスクリプトの作成方法等はサポート対象外となります。

プログラム・スクリプト言語の仕様は、MSDN をはじめとした言語提供元のサポート情報をご確認ください。

第2章 監視項目の作成・設定

2.1 カスタム監視の作成

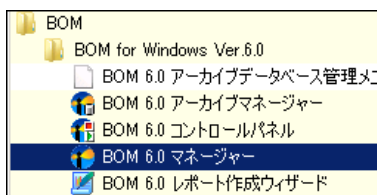
カスタム監視は BOM 6.0 の監視項目として作成します。以下の手順に沿って作業してください。

なお、作業にあたっては管理者権限が必要となりますので、管理者権限を持つアカウントにてログオンの上、作業を行ってください。

※ 以降の手順は必要な作業項目の概要のみを抽出した概略手順となります。

詳細な手順については、‘BOM for Windows Ver.6.0 ユーザーズ マニュアル’をご参照ください。

1. スタートメニューより、“BOM”→“BOM for Windows Ver.6.0”→“BOM 6.0 マネージャー”を選択します



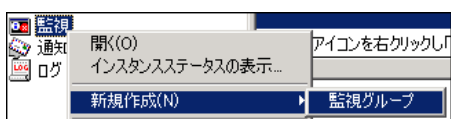
2. BOM マネージャーにて、“接続”をクリックし、監視コンピューターに接続します



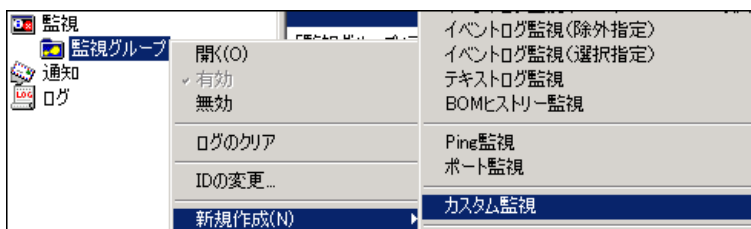
3. スcopeペインより“BOM for Windows Ver.6.0 (ローカル)” → “インスタンス名” → “監視”を選択します



4. 右クリックメニューから“新規作成” → “監視グループ”を選択し、監視グループを作成します



5. 作成した監視グループを選択し、右クリックメニューから“新規作成”→“カスタム監視”を選択します



6. 作成したカスタム監視を選択し、右クリックメニューから“プロパティ”を選択します



2.2 カスタム監視の設定

作成したカスタム監視は、スクリプトの配置場所や引数など、各種設定を行っていただく必要があります。

以下の内容を参考に設定を実施してください。

※ 以降の内容は概要のみを抽出した簡易情報となります。

項目の詳細な内容については、「BOM for Windows Ver.6.0 ユーザーズ マニュアル」をご参照ください。

1. 「全般」タブでは、監視項目名や監視間隔などを設定します

2. 「設定」タブでは、カスタム監視から実行する外部コマンドの名称や、外部コマンドに渡す引数などを設定します

3. 「拡張設定」タブでは、タイムアウト時間や実行アカウントの指定など、外部コマンドを実行するための環境情報を設定します

The screenshot shows the 'カスタム監視のプロパティ' (Custom Monitoring Properties) dialog box with the '拡張設定' (Extension Settings) tab selected. The dialog has four tabs: '全般' (General), '設定' (Settings), '拡張設定' (Extension Settings), and 'しきい値' (Thresholds). The '拡張設定' tab contains the following fields and controls:

- タイムアウト時間(T):** A text box with '30' and a unit dropdown set to '秒' (seconds).
- リトライ:** A group box containing:
 - リトライ回数(R):** A dropdown menu set to '0' and a unit dropdown set to '回' (times).
 - リトライ間隔(I):** A text box with '60' and a unit dropdown set to '秒' (seconds).
- 実行ユーザーアカウント:** A group box containing:
 - ユーザー名(U):** A text box.
 - ドメイン名(M):** A text box.
 - パスワード(P):** A text box.
- 作業フォルダー(C):** A text box with a '参照(B)...' (Browse...) button next to it.
- 環境変数:** A table with two columns: '変数' (Variable) and '値' (Value). To the right of the table are three buttons: '新規(N)...' (New...), '編集(E)...' (Edit...), and '削除(L)' (Delete).

At the bottom of the dialog are three buttons: 'OK', 'キャンセル' (Cancel), and '適用(A)' (Apply).

4. 「しきい値」タブでは、外部プログラムから返却された値に対するしきい値を設定します

The screenshot shows the 'カスタム監視のプロパティ' (Custom Monitoring Properties) dialog box with the 'しきい値' (Thresholds) tab selected. The dialog has four tabs: '全般' (General), '設定' (Settings), '拡張設定' (Extension Settings), and 'しきい値' (Thresholds). The 'しきい値' tab contains the following fields and controls:

- 注意(W):** A group box containing:
 - A text box with '0'.
 - A dropdown menu set to '以上' (Above).
- 危険(C):** A group box containing:
 - A text box with '0'.
 - A dropdown menu set to '以上' (Above).

At the bottom of the dialog are three buttons: 'OK', 'キャンセル' (Cancel), and '適用(A)' (Apply).

第3章 外部コマンドの実例

カスタム監視では、外部コマンドを実行し、その実行結果を標準出力から数値で取得し、監視結果(値)とします。

また、外部コマンドが正しく実行されたか否かを、終了コード(リターンコード)で確認します。

そのため、外部コマンドでは、カスタム監視に対し 0 以上の整数を戻り値として標準出力に出力する必要があります。

また、処理に失敗した場合には、カスタム監視に対し 0 以外の終了コードを返す必要があります。

なお、外部コマンドが実行されるコンピューターは、BOM 監視サービスをインストールしているコンピューターになります。

代理監視インスタンスでカスタム監視を使用する場合には、外部コマンドが代理監視元コンピューターで動くことに注意して下さい。

3.1 コンソールアプリケーションの場合

3.1.1 コンソールアプリケーション作成時の指針

コンソールアプリケーションを開発し、BOM 6.0 のカスタム監視に使用する場合には、数値が返却されるようにアプリケーションを作成してください。

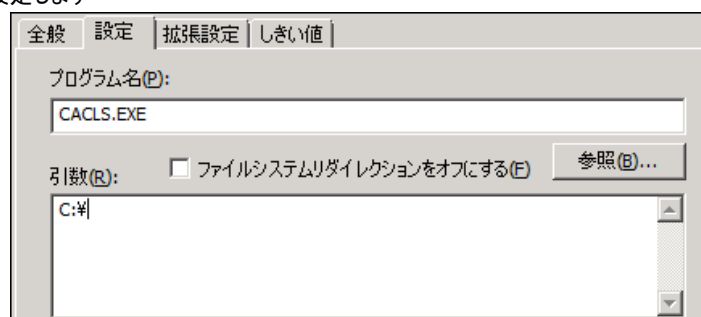
3.1.2 コンソールアプリケーションのラッピング

市販のコンソールアプリケーションなどをカスタム監視で使いたい場合には、コンソールアプリケーションをバッチファイルにラッピングして、数値以外の値を標準出力に出力しないようにします。

以下に、コンソールアプリケーション“CACLS.EXE”を使用して、C ドライブ直下に一般ユーザーのアクセス権があることを監視する場合について解説します。

1. 以下の設定でカスタム監視を実行した場合、外部プログラム“CACLS.EXE”はアクセス権の一覧を文字列で返却するため、カスタム監視では処理できず“(N/A)”という値を設定します

設定項目名	設定値
プログラム名	CACLS.EXE
引数	C:\¥



2. 以下は、アクセスコントロールリストの中に一般ユーザーが表示されていれば“1”を、表示がなければ“2”を返却し、合致したリストをログファイルへ出力するように“CACLS.EXE”をラッピングしたバッチファイルです

1	@ECHO OFF
2	CACLS.EXE %1 Find "BUILTIN\Users:" >"%~dpn0.log" 2>&1
3	IF "%ERRORLEVEL%"=="0" (ECHO 1) ELSE (ECHO 2)

3. 以下の設定でカスタム監視を実行した場合、外部プログラムであるバッチファイルは、アクセス権の一覧に一般ユーザーの文字列が存在する場合は“1”を、存在しない場合には“2”をカスタム監視へ返却します

その後、カスタム監視では“1”を受け取れば“正常”、“2”を受け取れば“危険”というステータスを設定します

設定項目名	設定値
プログラム名	バッチファイルのパス
引数	C:¥
注意しきい値	2 と等しい
危険しきい値	2 と等しい

4. 必要に応じて、メール送信アクションを作成し、アクセス権の一覧を記録したログファイルを送付できます

作成アクション	設定項目	設定値
メール送信	作成場所	カスタム監視 配下
	実行ステータス	正常
	実行頻度	毎回
	添付ファイル	ログファイル名

3.2 バッチファイル(コマンドラインスクリプト)の場合

Windows のコマンドライン用スクリプトである“バッチファイル”を使用して、カスタム監視用のスクリプトを作成することができます。

カスタム監視の外部コマンドとして、バッチファイルを使用する場合の手順を以下に示します。

3.2.1 バッチファイル作成時の指針

バッチファイルを BOM 6.0 のカスタム監視に使用する場合には、数値のみが返却されるように作成します。

以下のバッチファイルは、“CHKDSK”コマンドを実行し、ディスクが正常であることを確認するサンプルスクリプトです。

以下のサンプルスクリプトを元に、バッチファイルを作成する上での注意点について解説いたします。

1	@ECHO OFF
2	SET LogFile="%~dpn0. log"
3	ECHO 開始時間 %Date% %Time% 1>"%LogFile%" 2>&1
4	CHKDSK %1 1>>"%LogFile%" 2>&1
5	IF ERRORLEVEL 1 EXIT -1
6	ECHO 終了時間 %Date% %Time% 1>>"%LogFile%" 2>&1
7	FIND "問題は見つかりませんでした。" %LogFile% 1>NUL 2>&1
8	IF "%ERRORLEVEL%"=="0" (ECHO 1) ELSE (ECHO 2)

1. コマンド実行文字列が標準出力に出力されてしまわないように、バッチファイルの先頭行には“@ECHO OFF”コマンドを記載します

```
@ECHO OFF
```

2. 標準出力およびエラー出力に不要なメッセージが出力されないように、“NUL”またはファイルにメッセージをリダイレクトします
メッセージのリダイレクトには“>”(上書き)または“>>”(追記)を使用します

```
FIND "問題は見つかりませんでした。" %LogFile% 1>NUL 2>&1
```

```
CHKDSK %1 1>>"%LogFile%" 2>&1
```

3. “IF”コマンドで実行結果を判定し、結果を“ECHO”コマンドで標準出力に出力します

※ このサンプルスクリプトでは、正常に実行された場合には“1”を、エラーが発生した場合には“2”を標準出力に出力します

```
IF "%ERRORLEVEL%"=="0" (ECHO 1) ELSE (ECHO 2)
```

4. メイン処理に失敗した場合には“EXIT”コマンドに終了コードを付与して終了します

```
IF ERRORLEVEL 1 EXIT 1
```

5. バッチファイルが完成したら、カスタム監視のプロパティにてバッチファイル、引数などを指定します

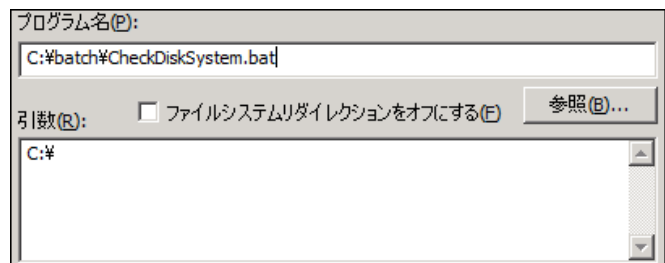
設定項目名	設定値
プログラム名	バッチファイルのパス
引数	バッチファイルの引数
注意しきい値	“ECHO”コマンドの値
危険しきい値	“ECHO”コマンドの値

3.2.2 バッチファイル実行時の挙動

カスタム監視で先のサンプルスクリプトを設定し、C ドライブのディスク状態をチェックする挙動について確認します。

1. サンプルスクリプトを“CheckDiskSystem.bat”という名称で保存します
2. カスタム監視にて、作成したバッチファイルを設定します

設定項目名	設定値
監視間隔	1 日
プログラム名	作成バッチファイルのパス
引数	C:
タイムアウト	600 秒
注意しきい値	2 と等しい
危険しきい値	2 と等しい



※ “CHKDSK”コマンドは環境によっては完了までに 600 秒以上かかる場合があります

その場合には、タイムアウト時間を増やして確認してください

3. BOM 監視サービスを起動して、カスタム監視を実行します
4. カスタム監視がバッチファイルを起動し、バッチファイル内のコマンドを順次処理します
5. 4 行目で“CHKDSK”コマンドが実行されます
“CHKDSK”コマンドは、実行結果に応じて以下のメッセージをログファイルに出力します

- ディスクの状態が正常な場合のメッセージ:

ファイル システムのチェックが終了しました。問題は見つかりませんでした。

- ディスクの状態に異常が見つかった場合のメッセージ:

ファイル システムに問題が見つかりました。

CHKDSK を /F オプションを使って実行して、問題を修正してください。

6. なんらかの理由で“CHKDSK”コマンドの実行に失敗した場合は、5 行目で“EXIT -1”が実行されて終了します
カスタム監視はバッチファイルから “-1”を終了コードとして受け取り、監視結果を“失敗”に設定します
7. 7 行目で“FIND”コマンドが実行されます
“FIND”コマンドは、ディスクの状態が正常な場合に出力されるメッセージをログファイルから検索します
メッセージを検出した場合には正常と判定し、“IF”構文で“ECHO 1”コマンドを実行します
メッセージを検出なかった場合にはエラーと判定し、“IF”構文で“ECHO 2”コマンドを実行します
8. バッチファイルが正常に終了し、カスタム監視は“1”または“2”を、バッチファイルから結果として受け取ります
カスタム監視は、“1”を受け取った時は“正常”、“2”を受け取った時には“危険”にステータスを設定します

3.3 WSH(Windows Script Host)の場合

Windows のスクリプトホストである WSH(Windows Script Host)を使用して、カスタム監視用のスクリプトを作成することができます。

カスタム監視では、コマンドラインホストである CScript を指定します。

3.3.1 WSH スクリプト作成時の指針

WSH スクリプトを BOM 6.0 のカスタム監視に使用する場合には、数値のみが返却されるように作成します。

以下の VBS は、システム稼働時間を取得するサンプルスクリプトです。

以下のサンプルスクリプトを元に、WSH スクリプトを作成する上での注意点について解説いたします。

1	On Error Resume Next
2	Dim colOS ' WMI コレクション
3	Dim objOS ' WMI オブジェクト
4	Dim intUpTime ' システム稼働時間 (秒)
5	
6	If Ucase(right(WScript.FullName, 11)) <> "CSCRIPT.EXE" Then WScript.Quit
7	
8	Set colOS = GetObject("winmgmts:root\cimv2").ExecQuery _
9	("Select * From Win32_PerfFormattedData_PerfOS_System")
10	If Err Then WScript.Quit(Err.Number) ' WMI コレクション取得失敗
11	
12	For Each objOS in colOS
13	intUpTime = objOS.SystemUpTime
14	Next
15	If Err Then WScript.Quit(Err.Number) ' WMI オブジェクト取得失敗
16	
17	If IsNumeric(intUpTime) Then
18	WScript.Echo intUpTime
19	Else
20	WScript.Quit(-1) ' SystemUpTime 取得失敗
21	End If

1. スクリプトエラーのエラーメッセージが標準出力およびエラー出力に返さないように、VBS の先頭には“On Error Resume Next”構文を記載します

```
On Error Resume Next
```

2. GUI ホストである WScript から起動しないように、CScript から起動しているかを確認します

```
If Ucase(right(WScript.FullName, 11)) <> "CSCRIPT.EXE" Then WScript.Quit
```

3. 標準出力およびエラー出力に数値以外が出力されないように、“IsNumeric”関数を使用して出力する値が数値であることを確認します

```
If IsNumeric(intUpTime) Then
    WScript.Echo intUpTime
```

4. エラー発生時には、“Quit”メソッドに終了コードを渡してスクリプトを終了します

```
If Err Then WScript.Quit(Err.Number) ' WMI コレクション取得失敗
WScript.Quit(-1) ' SystemUpTime 取得失敗
```

5. VBS ファイルが完成したら、カスタム監視のプロパティにてプログラム名に“CScript.exe”を、引数にスクリプトファイル、スクリプトファイルへの引数を指定します

また、CScript に渡す引数として、ログを表示しないための“//nologo”オプションが必要になることに注意して下さい

設定項目名	設定値
プログラム名	CScript.exe
引数	//nologo スクリプトのパス スクリプトの引数
注意しきい値	“Echo”メソッドの値
危険しきい値	“Echo”メソッドの値

3.3.2 WSH スクリプト実行時の挙動

カスタム監視で先のサンプルスクリプトを設定し、システム稼働時間をチェックする挙動について確認します。

1. サンプルスクリプトを“GetSystemUpTime.vbs”という名称で保存します
2. カスタム監視にて、作成した VBS ファイルを設定します

設定項目名	設定値
プログラム名	CScript.exe
引数	//nologo 作成 VBS ファイルのパス
注意しきい値	86400 以上
危険しきい値	604800 以上

※ しきい値は、注意が 1 日 (86,400 秒)、危険が 1 週間 (604,800 秒) を意味します

3. BOM 監視サービスを起動して、カスタム監視を実行します
4. カスタム監視が CScript.exe を起動し、CScript.exe が引数で指定された VBS ファイルを実行します
5. 8～9 行目で WMI コレクションを取得します
6. なんらかの理由で WMI コレクションの取得に失敗した場合は、10 行目で“Quit”メソッドが実行されて終了します
カスタム監視は CScript.exe からエラーコードを終了コードとして受け取り、監視結果を“失敗”に設定します
7. 12～14 行目で WMI オブジェクトを取得し、システム稼働時間を変数“intUpTime”に代入します

-
8. なんらかの理由で WMI オブジェクトの取得に失敗した場合は、15 行目で“Quit”メソッドが実行されて終了します
カスタム監視は CScript.exe からエラーコードを終了コードとして受け取り、監視結果を“失敗”に設定します
 9. 17 行目で変数“intUpTime”の値が数値か否かを判定します
 10. 変数“intUpTime”の値が数値であった場合は、18 行目でシステム稼働時間を出力します
CScript.exe が正常に終了し、カスタム監視はシステム稼働時間を、結果として受け取ります
カスタム監視のステータスは、“86400”未満の値を受け取った時は“正常”、“86400”以上の値を受け取った時は“注意”、“604800”以上の値を受け取った時には“危険”になります
 11. なんらかの理由で変数“intUpTime”の値が数値以外の場合は、20 行目で“Quit”メソッドが実行されて終了します
カスタム監視は CScript.exe からエラーコードを終了コードとして受け取り、監視結果を“失敗”に設定します

3.4 Windows PowerShell の場合

Windows PowerShell (以下、PowerShell と記)は、Windows 用の高機能なコマンドライン方式のシェル及びスクリプト実行環境です。
この PowerShell を使用して、カスタム監視用のスクリプトを作成することができます。

3.4.1 PowerShell によるカスタム監視の設定例

PowerShell を BOM 6.0 のカスタム監視に使用する場合には、数値のみが返却されるように作成します。

以下は、カスタム監視に実際に適用したサンプルとなります。これを元に PowerShell を使った実例を紹介いたします。

このサンプルは直近の“Application”イベントログの EventRecordID を監視値として取得するもので、ローカル監視・代理監視の双方に対応したものとなっています。

① “プログラム名”欄には“PowerShell.exe”と指定します

② “引数”欄には直接 PowerShell のスクリプトをダブルクォートで囲んで指定します。

Get-WinEvent イベントログを取得するコマンドレット(要 PowerShell Ver 2.0 以上)
 -Cn \$(TargetComputer) . . . イベントログを取得する対象のコンピューターを指定
 -logname Application . . . 取得する対象のイベントログを“Application”に指定
 -MaxEvent 1 取得するイベントログの件数を最新の 1 件に指定
 fw 取得したイベントログオブジェクトの表示を行うコマンドレット
 -Property RecordId . . . 表示するプロパティを“EventRecordID”に指定

③ “EventRecordID”は非常の大きくなる値なので“表示スケール”欄にて 1000 分の 1 を指定し、“表示単位”を“K records”と指定しています。

④ 各しきい値の設定を含めて以下の様にしています。

設定項目名	設定値
プログラム名	PowerShell.exe
引数	スクリプト自身
注意しきい値	3000000 K records 以上
危険しきい値	3800000 K records 以上

3.4.2 PowerShell 実行時の挙動

カスタム監視で先のサンプルスクリプトを設定し、システム稼働時間をチェックする挙動について確認します。

1. カスタム監視に前項の様に設定します

※ しきい値は、注意が 3000000 K レコード、危険が 3800000 K レコードを意味します

2. BOM 監視サービスを起動して、カスタム監視を実行します
3. カスタム監視が PowerShell.exe を起動し、PowerShell.exe が引数で指定されたスクリプトを実行します
4. Get-WinEvent コマンドレットで最新の Application ログを 1 件取得します
5. 取得したイベントログデータの内、EventRecordID のプロパティ値を表示します
6. なんらかの理由で指定イベントログの取得に失敗した場合は、コマンドレットのエラーコードを返して終了します
カスタム監視は PowerShell.exe からエラーコードを終了コードとして受け取り、監視結果を“失敗”に設定します
7. なんらかの理由で指定プロパティ値の表示に失敗した場合は、コマンドレットのエラーコードを返して終了します
カスタム監視は PowerShell.exe からエラーコードを終了コードとして受け取り、監視結果を“失敗”に設定します

第4章 FAQ

Q カスタム監視を実行した際、監視結果は正常とありますが、値が“(N/A)”という表示になり、監視結果が確認できません。

A カスタム監視に対し、外部コマンドが文字列等を返却した場合には、結果の取得に失敗し“(N/A)”という値になります。
外部コマンドが 0 以上の整数を返却するように修正してください。

Q カスタム監視で処理できる戻り値に制限はありますか。

A カスタム監視は、符号なし 64bit (0~18446744073709551615) の範囲であれば正常に処理できます。
範囲外の値がカスタム監視に返却された場合、監視に失敗します。

Q カスタム監視で処理できる終了コードに制限はありますか。

A カスタム監視は、符号つき 32bit (-2147483648~2147483647) の範囲で処理します。
範囲外の値がカスタム監視に返却された場合、カスタム監視側で値を切り捨てるため、元の終了コードを確認できません。

Q カスタム監視の結果が失敗であった時に、外部コマンドの終了コードを確認する方法はありますか。

A BOM マネージャにて、“BOM for Windows Ver.6.0 (ローカル)” → “インスタンス名” → “ログ” → “ヒストリー” → “監視” を選択し、該当するカスタム監視の失敗メッセージを開くことで、外部プログラムからの終了コードが確認できます。

BOM for Windows Ver.6.0

カスタム監視 ガイドライン

2013 年 12 月 22 日 初版

2014 年 7 月 7 日 改訂

著者 セイ・テクノロジーズ株式会社

発行者 セイ・テクノロジーズ株式会社

発行 セイ・テクノロジーズ株式会社

バージョン Ver.6.0.0.0

Copyright © 2013-2014 SAY Technologies, Inc. All rights reserved.
