Tech Fielders セミナー

第2部 応用編①
WMI による Windows Server の 監視

マイクロソフト株式会社
エバンジェリスト
安納 順一
http://blogs.technet.com/junichia/

本日の目的

WMIを使用して
Windows Server を監視する方法を学びます

※今回は VBScript を使用します

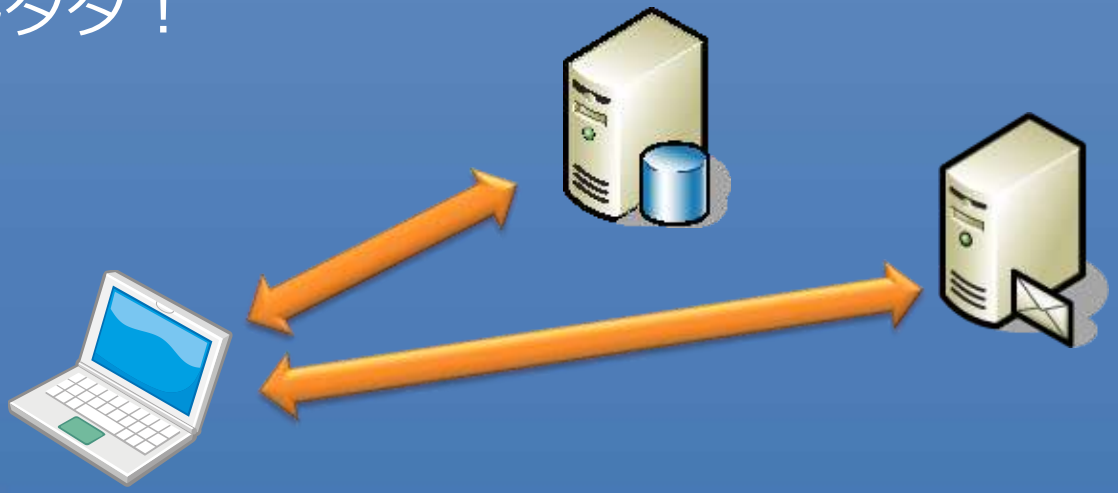
- WMIとは何か？
- WMIを使用したスクリプト作成の基礎
- WMIを使用したイベントの待ち合わせ
- WMIを使用した永続的監視の実装

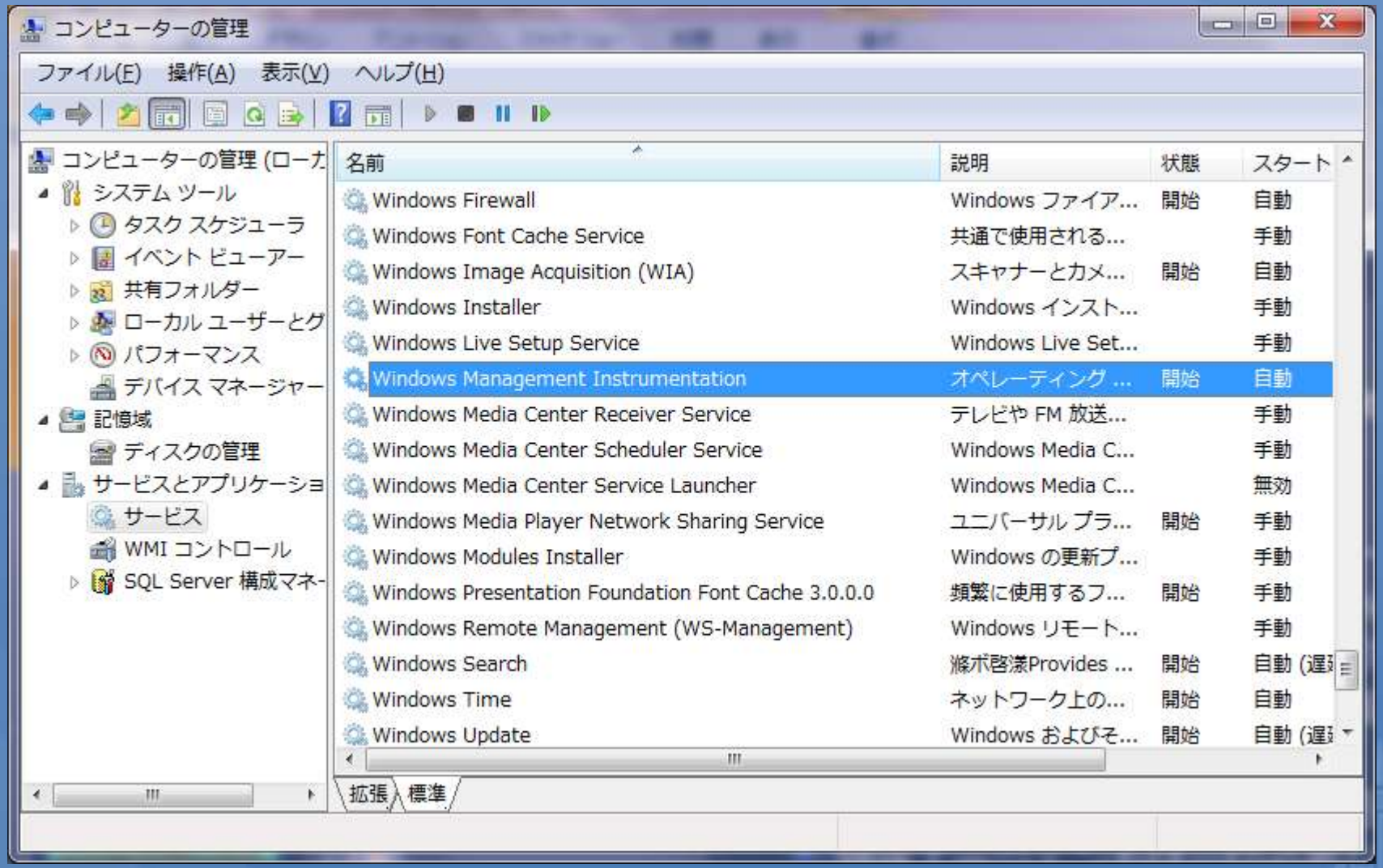
スクリプトの奥深さに感動していただけるはずです

WMI とは

Windows Management Instrumentation

- WBEM/CIMに準拠
- 各種管理情報への共通インターフェース
- Windows Management Serviceとして実装
- Windows 2000以上のOSに標準搭載
- WMIプロバイダはベンダーが拡張可能
- 別途エージェントは一切必要なし
- もちろんタダ！

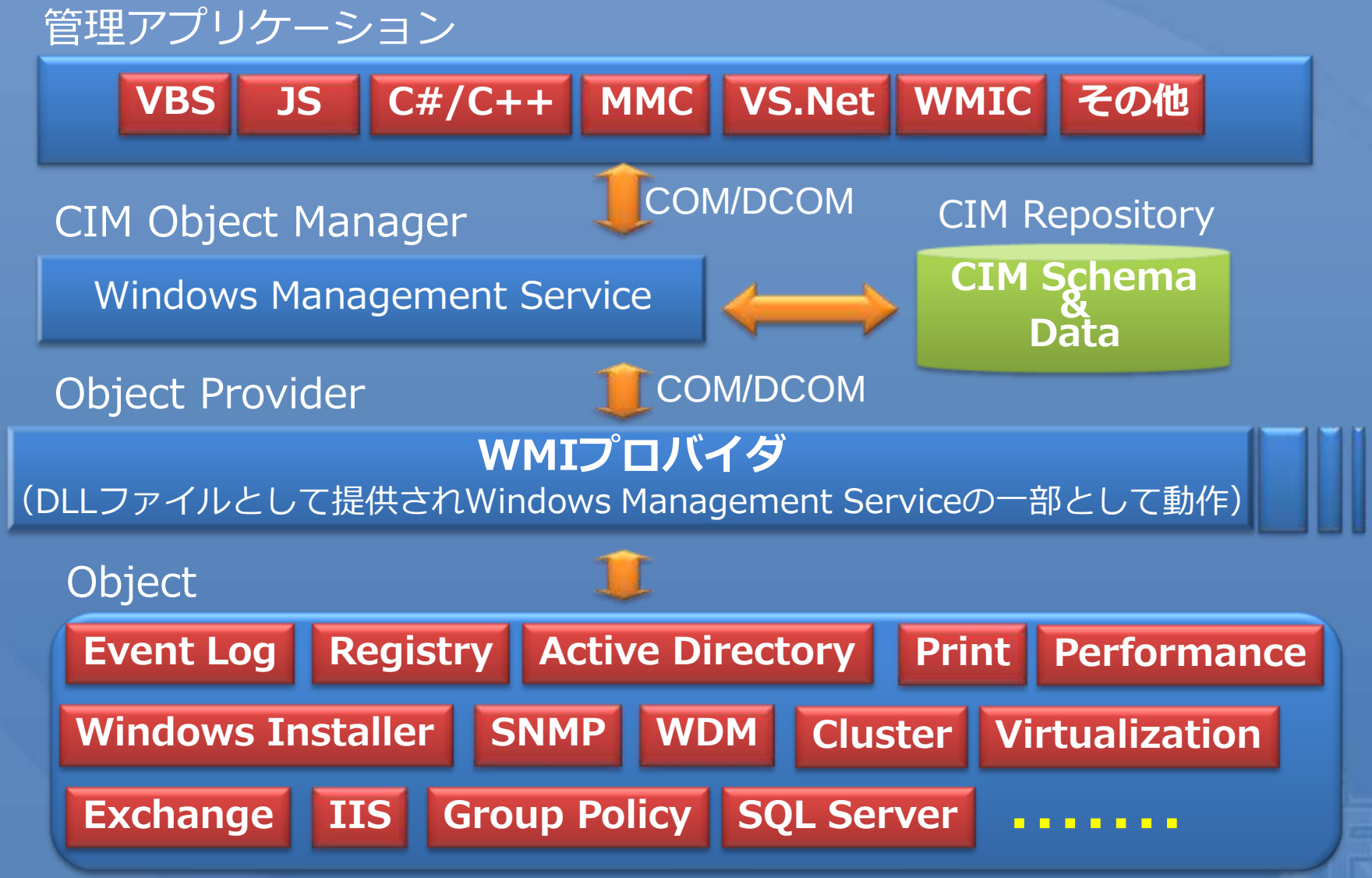




WMI にアクセスするには

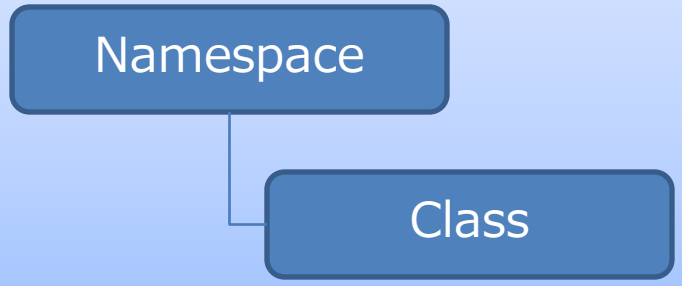
- バッチファイルから
  - WMICコマンド（Windows XP以降）
- Windows Script Host から
- PowerShell から
- Visual Studio から
- GUIツールから
  - WMI Admin Tools など

WMIのアーキテクチャ



WMIの構造

%WinDir%¥System32¥wbem 配下の mof （Managed Object Format）ファイルに定義されている



Namespace によって使える Class が異なることに注意

規定のNamespaceは **Root¥CIMV2**

（例）

ROOT¥COMV2
- Win32_OperatingSystem
- Win32_NTLogEventLog
- ・
- ・

ROOT¥Default
- STDREGPROV

ROOT¥microsoft¥SqlServer¥ComputerManagement10
- SqlService
- ・
- ・

WMIの主役 Win32_クラス

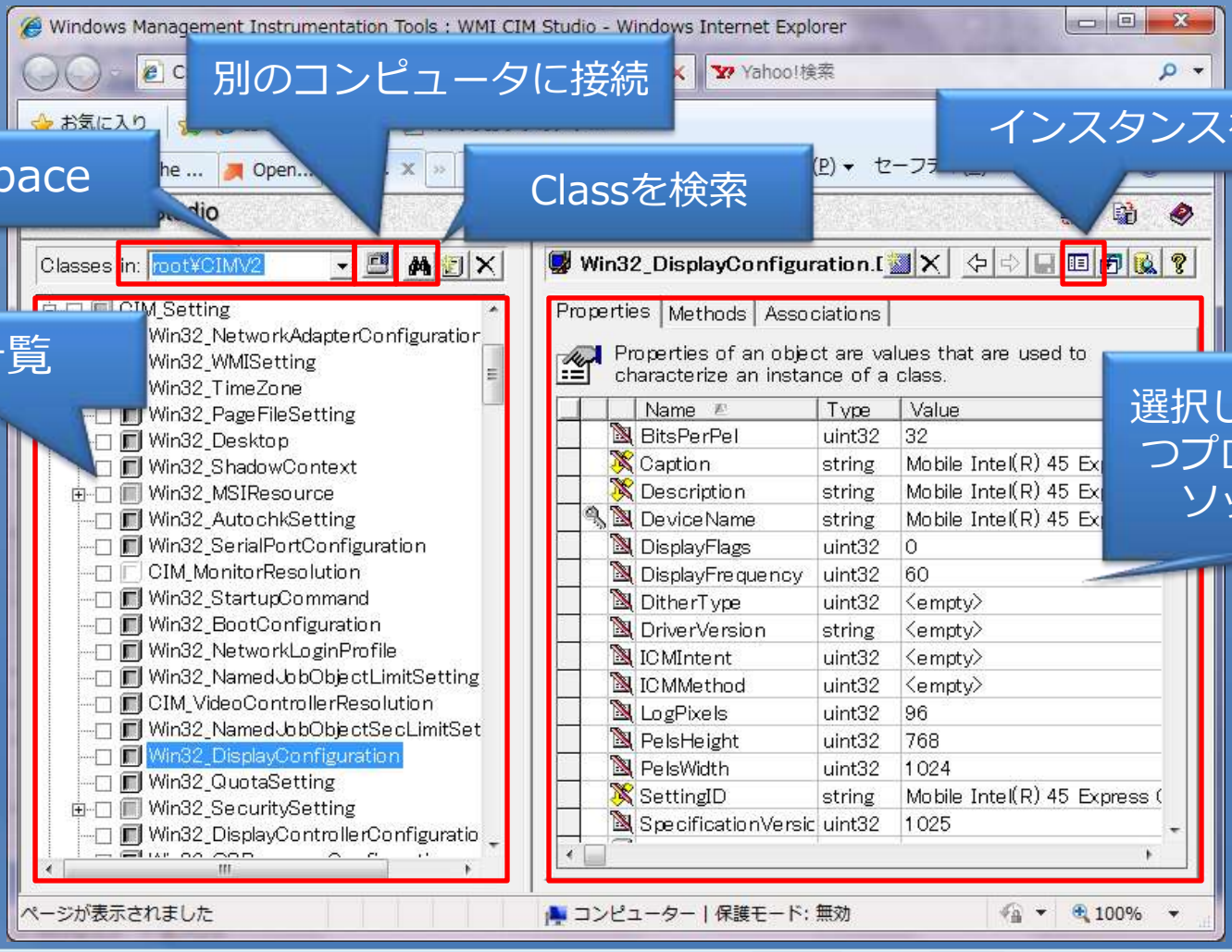
一部抜粋

Win32_ActiveRoute
Win32_ComputerShutdownEvent
Win32_ComputerSystemEvent
Win32_ConnectionShare
Win32_CurrentTime
Win32_DeviceChangeEvent
Win32_DiskQuota
Win32_GroupInDomain
Win32_IP4PersistedRouteTable
Win32_IP4RouteTable
Win32_IP4RouteTableEvent
Win32_JobObjectStatus
Win32_LoggedOnUser
Win32_LogonSession
Win32_LogonSessionMappedDisk
Win32_NetworkAdapter
Win32_ComputerSystem
・
・

Win32_ModuleTrace
Win32_NamedJobObject
Win32_NTDomain
Win32_PingStatus
Win32_ProcessStartTrace
Win32_ProcessStopTrace
Win32_Proxy
Win32_QuotaSetting
Win32_ServerConnection
Win32_SessionConnection
Win32_TokenGroups
Win32_TokenPrivileges
Win32_VolumeChangeEvent
Win32_WindowsProductActivation
Win32_ControllerHasHub
・
・
・

WMI Tools ～ CIM Studio

ブラウザを使用して、WMIの構造、実際のインスタンスを確認
クエリを発行して戻り値を確認
リモートコンピュータに接続も可能



WMIにVBScriptから接続

WMI Scriptingの基本形

wmisample01.vbs

```
'SWbemLocator オブジェクトの作成
Set Locator = CreateObject("WbemScripting.SWbemLocator")

‘ローカルコンピュータへの接続
Set Service = Locator.ConnectServer("", "root¥cimv2", "", "")

‘クエリーの定義（WQL : WMI Query Language）
strQuery  = "Select * from Win32_NetworkAdapterConfiguration " & _
            "where IPEnabled = True"

‘クエリーの実行（インスタンスを取得する）
Set objNet  = Service.ExecQuery(strQuery)

'結果の参照
For each n in objNet
        WScript.Echo n.caption
        WScript.Echo n.MACAddress
Next
```

Namespace

Class

多くの場合、戻り値はアレイ値

WMIスクリプトの実行権限

リモートコンピュータに対するアクセス権の取得

```
Set Service = Locator.ConnectServer(RemoteHost,Namespace,User,Password)
```

IPアドレスでも可だが、ドメインに参加しているとコンピュータ名でないとアクセスできないことがある

特殊権限の取得

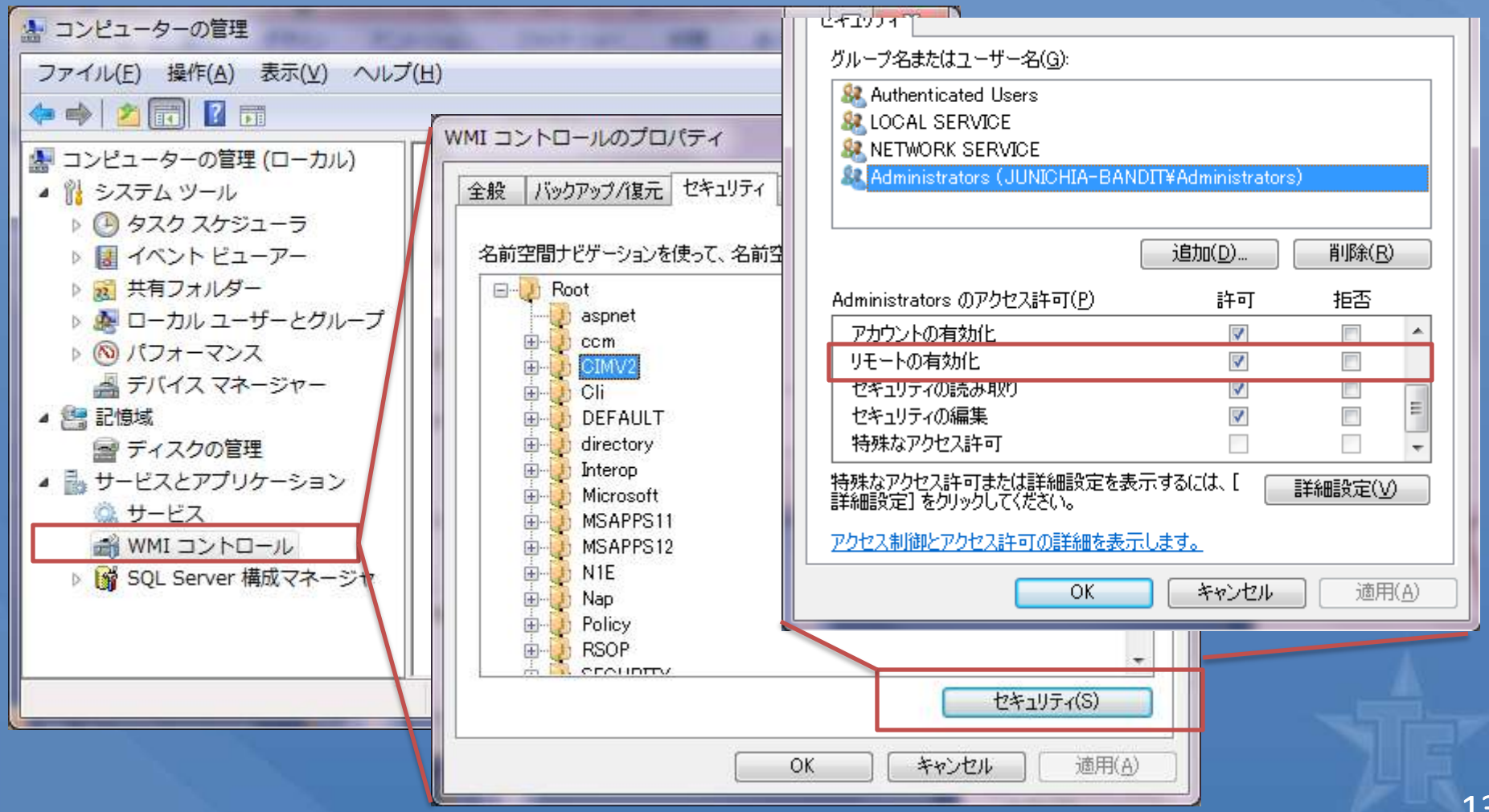
wmisample02.vbs

```
Set Locator = CreateObject("WbemScripting.SWbemLocator")

Set Service = Locator.ConnectServer("DC01", "root¥cimv2", "Dom¥administrator", "pass")

Service.Security_.Privileges.AddAsString "SeBackupPrivilege", True
Service.Security_.Privileges.AddAsString "SeSecurityPrivilege", True

strQuery = "Select * from Win32_NTEventlogFile" & _
           " Where LogfileName = 'Security' "

Set obj = Service.ExecQuery(strQuery)

For each n in obj
        r = n.BackupEventLog("C:¥tmp¥Security.evt")
Next
```

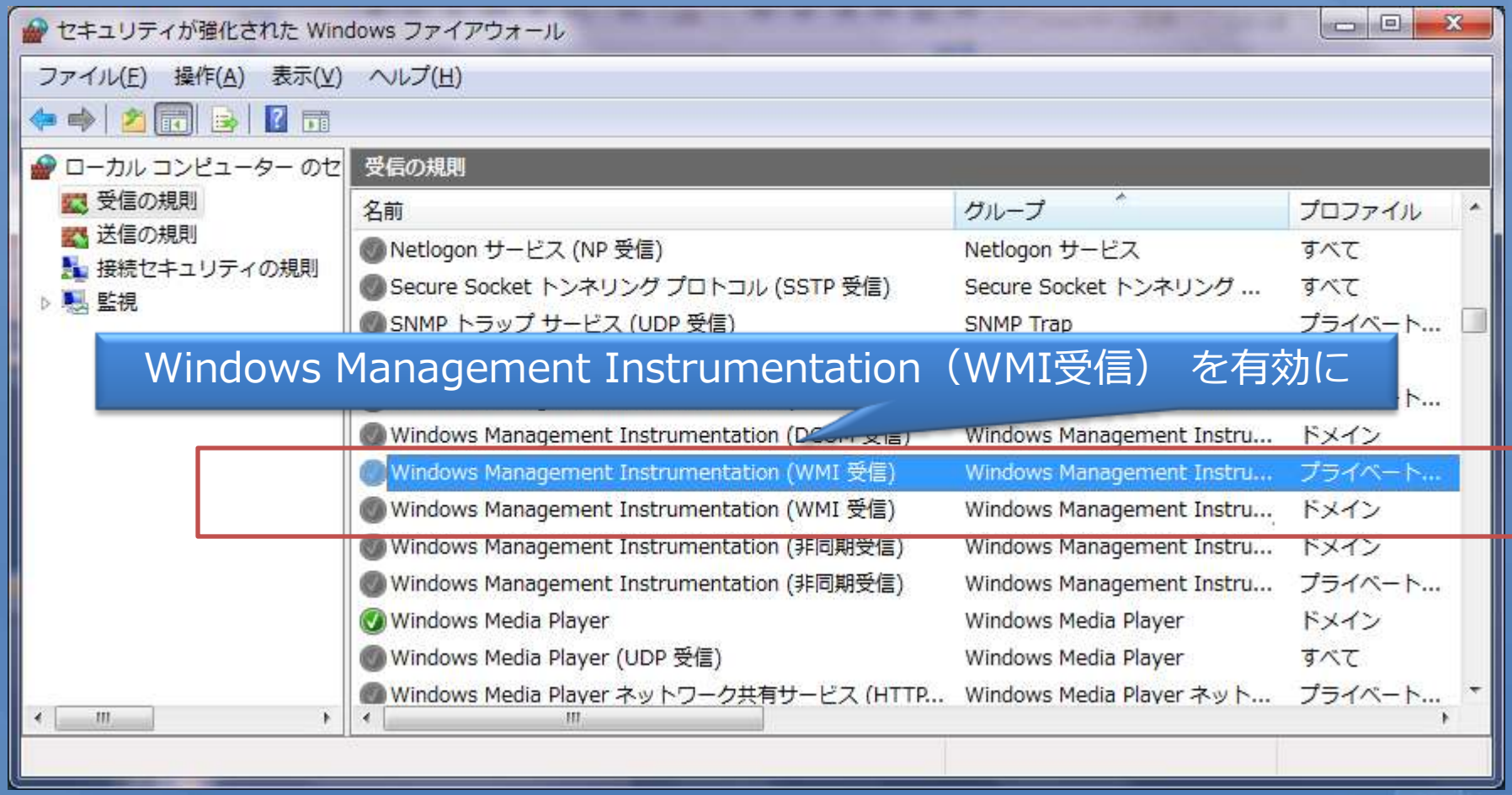
セキュリティログへのアクセスには特殊な権限が必要

（注意）リモートコンピュータにアクセスする前に①

リモートからの接続が許可されていることを確認
規定ではシステム管理者にはアクセスが許可されている



（注意）リモートコンピュータにアクセスする前に②



サーバーを監視する

イベントの監視

- システムで発生したイベントをトリガーとして処理を実行
- 一時監視と永続的監視
- 以下のイベント監視用クラスが用意されている
  - __InstanceCreationEvent
    - あたらしいインスタンスが作成された
  - __InstanceModificationEvent
    - インスタンスの属性が変更された
  - __InstanceDeletionEvent
    - 既存のインスタンスが削除された

アンダースコア2つ

監視のイメージ

WORDが起動

イベントログにイベントが書かれた

USBデバイスが挿された

ユーザーがログオン

タイムゾーンが変更

シャットダウン

メモ帳が終了

IPアドレス変更

WMIのClassでフィルタ

検出

一時的な監視例①

wmisample03.vbs

__InstanceCreationEventによる新たに作成されたインスタンスの監視

```
Set objLocator   = CreateObject("WbemScripting.SWbemLocator")
Set objService = objLocator.ConnectServer(TargetComputer, _
                                          "", AdminUser, AdminPass)

strQueryCreate = "Select  *  " & _
                 "FROM __InstanceCreationEvent  WITHIN 5 " & _
                 "WHERE TargetInstance  ISA 'Win32_Process' "

Set objEventsCreation = objService.ExecNotificationQuery(strQueryCreate)

Do
        Set CreationEvent = objEventsCreation.Nextevent
        ProcessName = CreationEvent.TargetInstance.Name
        Wscript.Echo ProcessName
Loop
```

監視したいクラス

Win32_Procassのプロパティ

一時的な監視例②

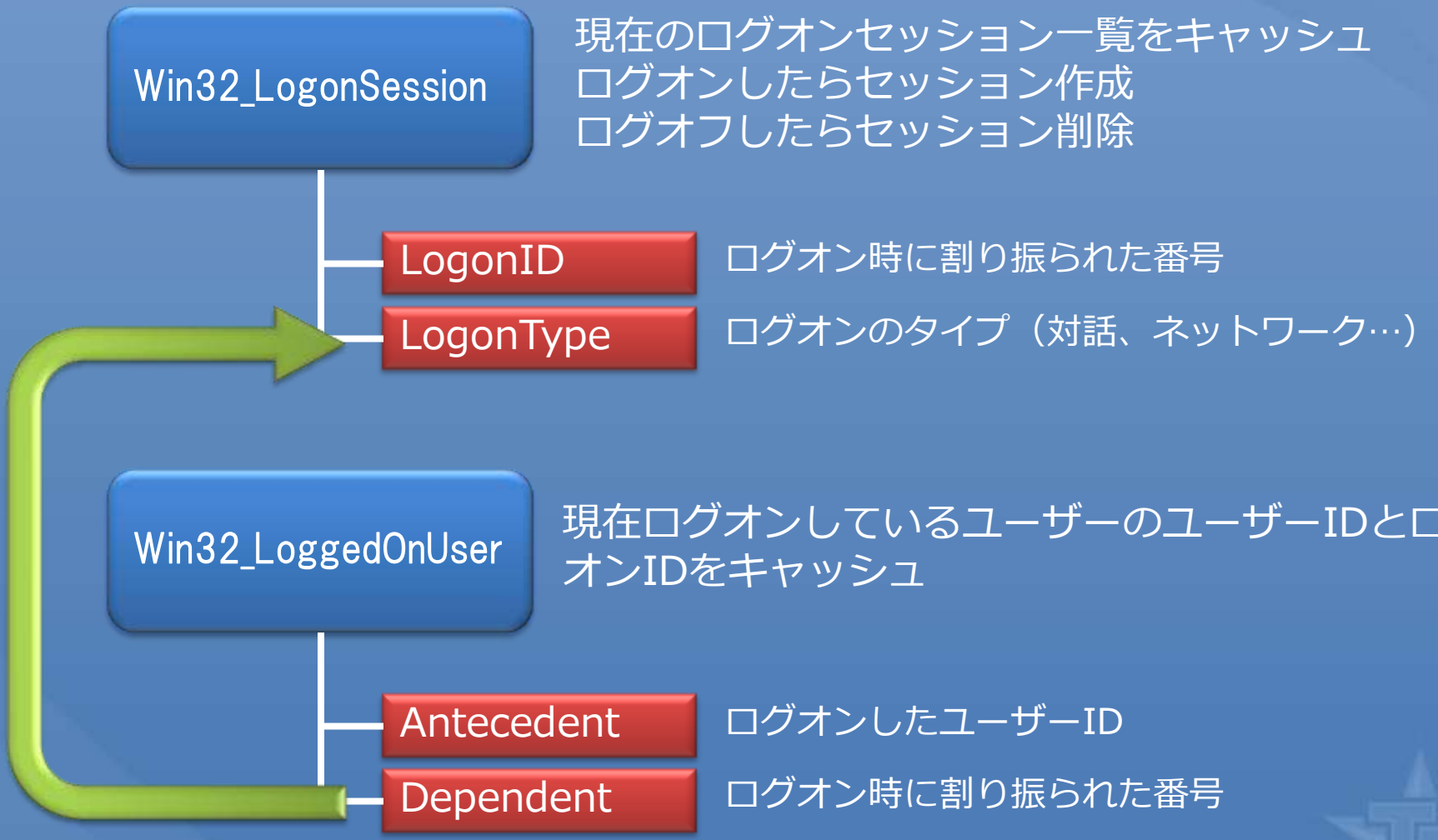
wmisample04.vbs

__InstanceModificationEventによるユーザーログオンの監視

```
Set objLocator   = CreateObject("WbemScripting.SWbemLocator")
Set objService = objLocator.ConnectServer(TargetComputer, _
                              "ROOT¥CIMV2", AdminUser, AdminPass)

strQuery= "Select * " & _
          "FROM __InstanceModificationEvent WITHIN 5 " & _
          "WHERE TargetInstance ISA 'Win32_ComputerSystem' "

Set objEventsModification = objService.ExecNotificationQuery(strQuery)

Do
          Set ModificationEvent = objEventsModification.Nextevent
          UserName = ModificationEvent.TargetInstance.UserName
          Wscript.Echo UserName
Loop
```

いまいち信頼性が無い…

そこでこんな方法…



一時的な監視例③

wmisample05.vbs

__InstanceCreationEventによるユーザーログオンの監視

```
Set objLocator   = CreateObject("WbemScripting.SWbemLocator")
Set objService = objLocator.ConnectServer("demo2008", "ROOT¥CIMV2", "", "")
Wscript.Echo "接続が完了しました"
strQueryCreate = "Select  *  FROM __InstanceCreationEvent WITHIN 5 " & _
                        "WHERE TargetInstance  ISA 'Win32_LogonSession' "
Set objEventsCreation = objService.ExecNotificationQuery(strQueryCreate)
Do
        Set CreationEvent = objEventsCreation.Nextevent
        LogonId= CreationEvent.TargetInstance.LogonID
        LogonType = CreationEvent.TargetInstance.LogonType
        strQueryLU = "Select * " & _
                "FROM Win32_LoggedOnUser"  ' Where Dependent like '%" & LogonId & "%' "
        Set objLoggedOnUser = objService.ExecQuery(strQueryLU)
        For Each u in objLoggedOnUser
                If instr(u.Dependent, LogonId) Then
                        Wscript.Echo u.Antecedent
                        Wscript.Echo u.Dependent

                End If
        Next
Loop
```

ユーザーID

ログオンID

一時的な監視例③‘ （③をﾌﾞﾗｯｼｭｱｯﾌﾟ）

wmisample06.vbs

```
Set objLocator   = CreateObject("WbemScripting.SWbemLocator")
Set objService = objLocator.ConnectServer("demo2008", "ROOT¥CIMV2", "", "")

Wscript.Echo "接続が完了しました"
strQueryCreate = "Select  * FROM _InstanceCreationEvent WITHIN 5 " & _
                        "WHERE TargetInstance  ISA 'Win32_LogonSession' "
Set objEventsCreation = objService.ExecNotificationQuery(strQueryCreate)
Do
        Set CreationEvent = objEventsCreation.Nextevent
        LogonId= CreationEvent.TargetInstance.LogonID
        LogonType = CreationEvent.TargetInstance.LogonType
        Select Case LogonType
                Case 0 strLogonType = "System"
                Case 2 strLogonType = "Interactive"
                Case 3 strLogonType = "Network"
                Case 4 strLogonType = "Batch"
                Case 5 strLogonType = "Service"
                Case 6 strLogonType = "Proxy"
                Case 7 strLogonType = "Unlock"
                Case 8 strLogonType = "NetworkClearText"
                Case 9 strLogonType = "NewCredentials"
                Case 10 strLogonType = "RemoteInteractive(TS)"
                Case 11 strLogonType = "CachedInteractive"
                Case 12 strLogonType = "CachedRemoteInteractive"
                Case 13 strLogonType = "CachedUnlock"
        End Select
```

```
        strQueryLoggedOnUser = "Select * " & _
                    "FROM Win32_LoggedOnUser"
        Set objLoggedOnUser = objService.ExecQuery(strQueryLoggedOnUser)

        For Each u in objLoggedOnUser
                If instr(u.Dependent, LogonId) Then
                   arrAntecedent = Split(u.Antecedent,".")
                   Wscript.Echo Date & "," & Time & "," & _
                                LogonId & "," & arrAntecedent(2) & "," & strLogonType
                            Exit For
                End If
        Next

Loop
```

一時的な監視例④

wmisample07.vbs

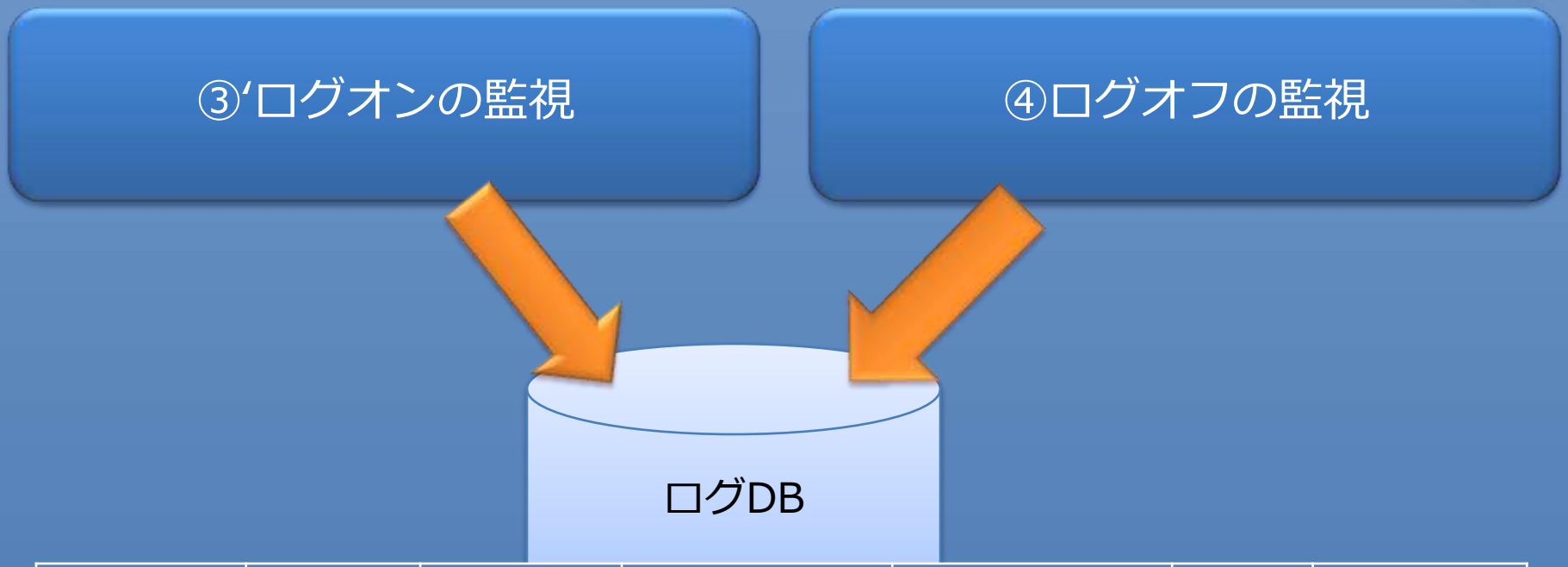
__InstanceDeletionEventによるユーザーログオフの監視

```
Set objLocator   = CreateObject("WbemScripting.SWbemLocator")
Set objService = objLocator.ConnectServer("demo2008", "ROOT¥CIMV2", "", "")

Wscript.Echo "接続が完了しました"
strQueryCreate = "Select  *  " & _
                        "FROM __InstanceDeletionEvent WITHIN 5 " & _
                        "WHERE TargetInstance  ISA 'Win32_LogonSession' "

Set objEventsDeletion = objService.ExecNotificationQuery(strQueryCreate)

Do
            Set DeletionEvent = objEventsDeletion.Nextevent
            LogonId= DeletionEvent.TargetInstance.LogonID
            LogonType = DeletionEvent.TargetInstance.LogonType

            Wscript.Echo Date & "," & Time & "," & LogonId & "," & strLogonType

Loop
```

セッションの削除はログオフしてから1分程度を要する

③’と④を組み合わせると

ログオンID をキーにしてログデータベースに書き込むことでサーバーを使用したユーザーの履歴を管理できる



ServerName	LogonID	LogonType	LogonDateTime	LogoffDateTime	Domain	UserID
demo2008	5443965	Interactive	2009/03/01 10:00:00	2009/03/01 12:15:30	dom	anno
demo2008	6220879	TS	2009/03/01 12:00:00	2009/03/01 13:10:01	dom	administrator

スクリプトをサービス化

イベントコンシューマ

イベントコンシューマとは….

システムで発生したイベントをトリガーに特定のアクションを実行する機構

- **ActiveScriptEventConsumer**
  - **イベントが発生したらスクリプトを実行**
- LogFileEventConsumer
  - イベントが発生したらテキストファイルに書き込み
- NTEventLogEventConsumer
  - イベントが発生したらイベント
- SMTPEventConsumer
  - イベントが発生したらメール送信
- CommandLineEventConsumer
  - イベントが発生したらコマンドを実行

システムに登録されるのでログオンする必要が無い
ただし、監視できるのはローカルコンピュータ

MOFファイルによる永続的監視

ActiveScriptEventConsumerによるmofファイルの例

```
#pragma namespace("¥¥¥¥.¥¥root¥¥subscription")

instance of ActiveScriptEventConsumer as $Cons
{
    Name = "LogonUserLogging";
    ScriptingEngine = "VBScript";
    ScriptFileName = "c:¥¥tmp¥¥demoscript¥¥wmisample08.vbs";
};
```

1

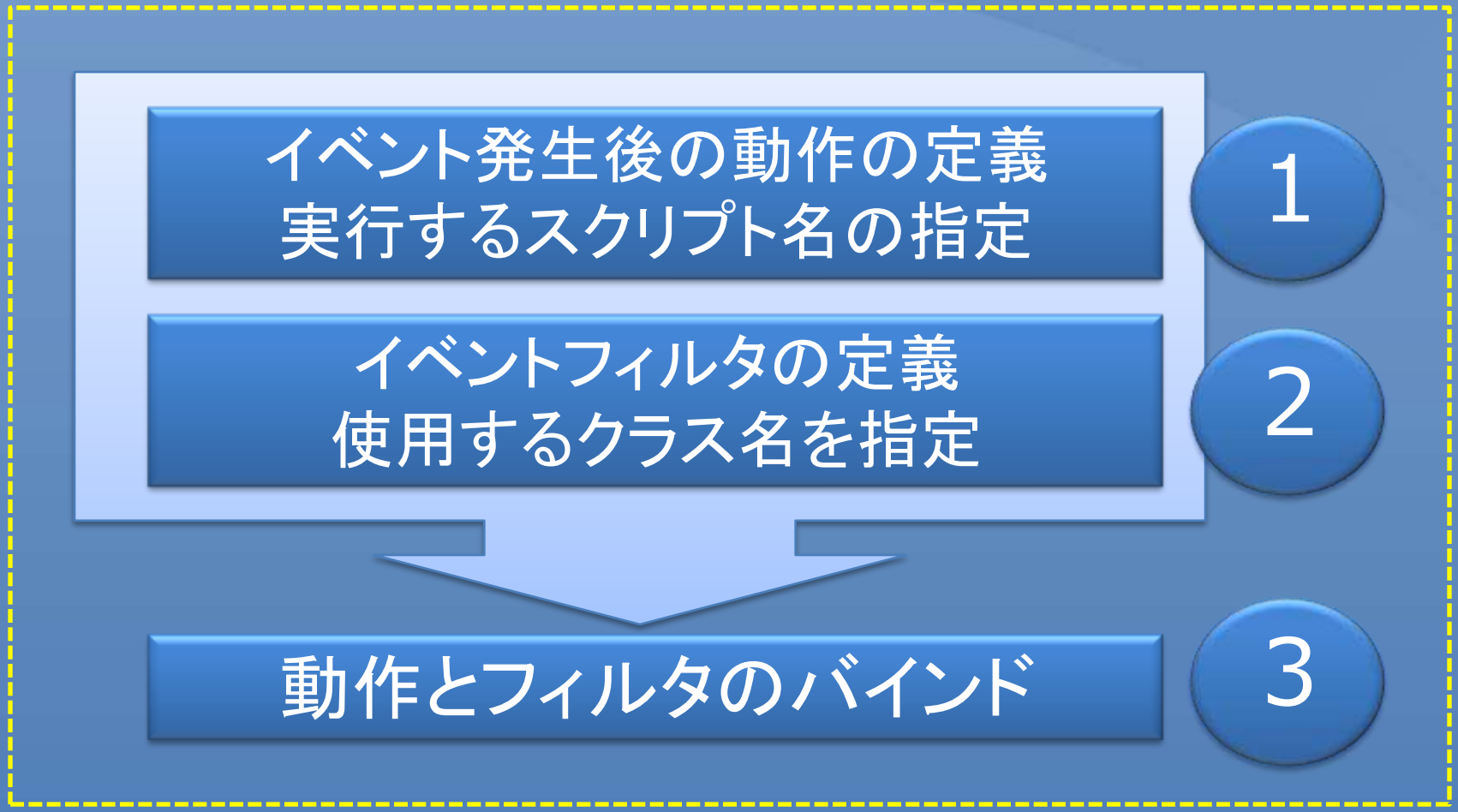
```
instance of _EventFilter as $Filt
{
    Name = "LogonUser";
    Query = "SELECT * FROM _InstanceCreationEvent WITHIN 5 "
          "WHERE TargetInstance ISA ¥"Win32_LogonSession¥" ";
    QueryLanguage = "WQL";
    EventNamespace = "root¥¥cimv2";
};
```

2

```
instance of _FilterToConsumerBinding
{
    Filter = $Filt;
    Consumer = $Cons;
};
```

3

拡張子 mof で保存



コマンドプロンプトからコンパイル

C:¥>mofcomp.exe <mofファイル名>

呼び出されるスクリプト（③”）

wmisample08.vbs

もとのスクリプトを
ちょっとだけ修正

```
Set objLocator   = CreateObject("WbemScripting.SWbemLocator")
Set objService = objLocator.ConnectServer("demo2008", "ROOT¥CIMV2", "", "")
Set objFS = CreateObject("Scripting.FileSystemObject")

Wscript.Echo "接続が完了しました"
strQueryCreate = "Select * FROM __InstanceCreationEvent WITHIN 5 " & _
                 "WHERE TargetInstance ISA 'Win32_LogonSession' "
Set objEventsCreation = objService.ExecNotificationQuery(strQueryCreate)
Do
        Set CreationEvent = objEventsCreation.Nextevent
        LogonId= CreationEventTargetEvvent.TargetInstance.LogonID
        LogonType = CreationEventTargetEvent.TargetInstance.LogonType
        Select Case LogonType
                Case 0 strLogonType = "System"
                Case 2 strLogonType = "Interactive"
                Case 3 strLogonType = "Network"
                Case 4 strLogonType = "Batch"
                Case 5 strLogonType = "Service"
                Case 6 strLogonType = "Proxy"
                Case 7 strLogonType = "Unlock"
                Case 8 strLogonType = "NetworkClearText"
                Case 9 strLogonType = "NewCredentials"
                Case 10 strLogonType = "RemoteInteractive(TS)"
                Case 11 strLogonType = "CachedInteractive"
                Case 12 strLogonType = "CachedRemoteInteractive"
                Case 13 strLogonType = "CachedUnlock"
        End Select
```

```
        strQueryLoggedOnUser = "Select * FROM Win32_LoggedOnUser"
        Set objLoggedOnUser = objService.ExecQuery(strQueryLoggedOnUser)

        For Each u in objLoggedOnUser
            If instr(u.Dependent, LogonId) Then
           arrAntecedent = Split(u.Antecedent,".")
           Set objLogFile = objFS.OpenTextFile("C:¥tmp¥demoscript¥userlog.txt",8,True)
           objLogFile.WriteLine Date & "," & Time & "," & LogonId & "," & _
                                arrAntecedent(2) & "," & strLogonType
           objLogFile.Close
           Wscript.Echo Date & "," & Time & "," & _
                                LogonId & "," & arrAntecedent(2) & "," & strLogonType
        Exit For
                    End If
        Next
Loop
```

まとめ

WMIはパターンを覚えれば簡単です

あたらしい使い方で作業が劇的に変化します！

Yes, we can !

リソース

MSDN - Windows Management Instrumentation
http://msdn.microsoft.com/en-us/library/aa394582(VS.85).aspx

Script Guy!
http://www.microsoft.com/japan/technet/scriptcenter/resources/qanda/default.mspx

検索！検索！検索！

本日使用したスクリプト
http://blogs.technet.com/junichia/pages/3221367.aspx