

CLUSTERPRO MC ProcessSaver 2.11 for Windows テンプレートガイド

第 14 版

2026 年 4 月

日本電気株式会社

版数	改版	内容
1.0	2015.03	新規作成
2.0	2016.03	バージョンアップに伴い改版
3.0	2017.04	バージョンアップに伴い改版
4.0	2018.04	バージョンアップに伴い改版
5.0	2018.06	商標の記載の修正
6.0	2019.04	バージョンアップに伴い改版
7.0	2020.04	バージョンアップに伴い改版
8.0	2021.04	監視テンプレートを最新化 WebOTX Application Server 監視テンプレートを新規作成 WebSAM JobCenter WebSAM SystemManager G ESMPRO/ServerAgentService
9.0	2021.04	監視テンプレートを更新 WebSAM JobCenter - 再起動スクリプトに JobCenter 管理者を追加 監視テンプレートガイドの情報を更新 WebSAM SystemManager G - 概要の表に「常駐プロセス」を記載 ESMPRO/ServerAgentService - SNMP サービスの「回復機能」を訂正
10.0	2022.04	バージョンアップに伴い改版
11.0	2023.04	バージョンアップに伴い改版
12.0	2024.04	バージョンアップに伴い改版 64bitプロセス変更に伴い、インストール先のデフォルトフォルダーを変更
13.0	2025.04	バージョンアップに伴い改版
14.0	2026.04	バージョンアップに伴い改版

(1) マニュアルについて

「CLUSTERPRO MC ProcessSaver 2.11 for Windows テンプレートガイド」は、ProcessSaver によるプロセス/サービス監視ソリューションを支援するための導入事例をご紹介します。

本資料でご紹介する事例は、監視対象となる製品の設定ファイルの設定・運用スタイル・バージョンの差異、さらにバッチなどによって、監視手順に変更が発生する場合があります。実際に導入する場合は、十分な検証と動作確認の実施をお願いします。

(2) 商標および著作権について

- Apache、Tomcat は、Apache Software Foundation の登録商標または商標です。
- WebOTX、ESMPRO、WebSAM は、日本電気株式会社の商標または登録商標です。
- Oracle は、Oracle Corporation およびその子会社、関連会社の米国およびその他の国における登録商標です。文中の社名、商品名等は各社の商標または登録商標である場合があります。
- その他記載の製品名および会社名は、すべて各社の商標または登録商標です。

なお、本書では®、TM マークを明記しておりません。

1. はじめに	P.15
2. Tomcat 6 の監視事例	P.19
1. 概要	
2. Pfile	
3. 再起動スクリプト	
3. Tomcat 7 の監視事例	P.22
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. WebOTX 9.1 の監視事例	P.25
1. 概要	
2. Pfile	
3. 再起動スクリプト	
5. WebOTX 10.3 の監視事例	P.28
1. 概要	
2. Pfile	
3. 再起動スクリプト	

6. WebLogic 11g の監視事例	P.31
1. 概要	
2. Pfile	
3. 再起動スクリプト	
7. WebLogic 12c の監視事例	P.34
1. 概要	
2. Pfile	
3. 再起動スクリプト	
8. ESMPRO/ServerAgentService Ver.2 の監視事例	P.37
1. 概要	
2. Pfile	
3. 再起動スクリプト	

9. WebSAM JobCenter 15.4 の監視事例	P.41
1. 概要	
2. Pfile	
3. 再起動スクリプト	
10. WebSAM SystemManager G 10.1 の監視事例	P.44
1. 概要	
2. Pfile	
3. 再起動スクリプト	

11. Oracle 10g の監視事例	P.47
1. システム一意	
1-1. 概要	
1-2. Pfile	
1-3. 再起動スクリプト	
2. インスタンス	
2-1. 概要	
2-2. Pfile	
2-3. 再起動スクリプト	
12. Oracle 11g の監視事例	P.59
1. システム一意	
1-1. 概要	
1-2. Pfile	
1-3. 再起動スクリプト	
2. インスタンス	
2-1. 概要	
2-2. Pfile	
2-3. 再起動スクリプト	

13. Oracle 12c の監視事例	P.68
1. システム一意	
1-1. 概要	
1-2. Pfile	
1-3. 再起動スクリプト	
2. インスタンス	
2-1. 概要	
2-2. Pfile	
2-3. 再起動スクリプト	
14. IIS6.0 の監視事例	P.77
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. Pfile 書き換え	
4-1. IIS の状態確認	
4-2. Pfile の読み込み	
4-3. 監視しないサービスの削除	
4-4. 監視サービスの並び替え	
4-5. Pfile の保存	

15. IIS7.0 の監視事例	P.105
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. Pfile 書き換え	
4-1. Pfile の直接編集	
4-2. Pfile の読み込み	
4-3. 監視しないサービスの削除	
4-4. Pfile の保存	
16. IIS7.5 の監視事例	P.126
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. Pfile 書き換え	
4-1. Pfile の直接編集	
4-2. Pfile の読み込み	
4-3. 監視しないサービスの削除	
4-4. Pfile の保存	

17. IIS8.0 の監視事例	P.142
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. Pfile 書き換え	
4-1. Pfile の直接編集	
4-2. Pfile の読み込み	
4-3. 監視しないサービスの削除	
4-4. Pfile の保存	

18. Microsoft SQL Server 2008 の監視事例	P.158
1. システム一意	
1-1. 概要	
1-2. Pfile	
1-3. 再起動スクリプト	
2. インスタンス	
2-1. 概要	
2-2. Pfile	
2-3. 再起動スクリプト	
3. Pfile 書き換え	
3-1. Pfile の直接編集	
3-2. Pfile の読み込み	
3-3. 監視しないサービスの削除	
3-4. Pfile の保存	

19. Microsoft SQL Server 2008 R2 の監視事例	P.183
1. システム一意	
1-1. 概要	
1-2. Pfile	
1-3. 再起動スクリプト	
2. インスタンス	
2-1. 概要	
2-2. Pfile	
2-3. 再起動スクリプト	
3. Pfile 書き換え	
3-1. Pfile の直接編集	
3-2. Pfile の読み込み	
3-3. 監視しないサービスの削除	
3-4. Pfile の保存	

20. Microsoft SQL Server 2012 の監視事例	P.207
1. システム一意	
1-1. 概要	
1-2. Pfile	
1-3. 再起動スクリプト	
2. インスタンス	
2-1. 概要	
2-2. Pfile	
2-3. 再起動スクリプト	
3. Pfile 書き換え	
3-1. Pfile の直接編集	
3-2. Pfile の読み込み	
3-3. 監視しないサービスの削除	
3-4. Pfile の保存	

21. StarOffice の監視事例	P.233
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. Pfile 書き換え	
4-1. Pfile の直接編集	
4-2. Pfile の読み込み	
4-3. 監視しないサービスの削除	
4-4. Pfile の保存	
22. IIS のストール監視事例	P.248
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. ストール監視スクリプトの設定	
23. Microsoft SQL Server のストール監視事例	P.261
1. 概要	
2. Pfile	
3. 再起動スクリプト	
4. ストール監視スクリプトの設定	

1. はじめに

表記規則

本書での表記規則に関しては、下記のように定義します。

記号表記	使用方法	例
『』	画面名の前後	『Create Pfile』 は、Pfile を作成する画面です。
「」	参照するマニュアル名の前後 参照する章および章のタイトル名の前後	「CLUSTERPRO MC ProcessSaver 2.10 for Windows ユーザーズガイド（コンソール編）」を参照してください。
【】	環境によって読み換えるファイル名および フォルダー名の前後	【HOME_NAME】は、Oracle のインストール時に指定するホーム名です。
< >	ファイル名およびフォルダー名の前後	<C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_iis6.0>
[]	項目名の前後・タブ名の前後	[Process/ServiceName] を変更してください。
斜体、太字	ボタン名・チェックボックス名	<i>Read</i> を押してください。

1. (続き)

用語の定義

本書での用語に関しては、下記のように定義します。

用語	説明
ProcessSaver コンソール	ProcessSaver の運用管理のための GUI (グラフィカルユーザーインターフェース) 画面のことを、ProcessSaver コンソールと表記します。
Pfile	プロセス/サービスの監視定義情報を記述する設定ファイルのことを、Pfile と表記します。
Pcheck	Pfile で指定された定義情報に基づき、プロセス/サービス消滅監視および再開を行うプロセスのことを、Pcheck と表記します。
再起動スクリプト	監視対象プロセス/サービスの消滅を検知した場合に、監視対象プロセス/サービスを自動的に再起動するためのコマンド処理を記述したファイルのことを、再起動スクリプトと表記します。
PcheckRunList	OS 起動時に自動的に監視を開始するための Pcheck の起動リストです。監視を行う Pfile の一覧が記述されています。
インストールフォルダー	ProcessSaver をインストールしたフォルダーです。本資料では、<C:¥Program Files> にインストールしたものとします。
Pcheck 自動起動	OS 起動時に ProcessSaver サービスから PcheckRunList にしたがって Pcheck を起動することです。
Pcheck 手動起動	OS 起動後に ProcessSaver コンソールや コマンドから Pcheck を起動することです。

1. (続き)

参考ドキュメント

Pcheck 自動起動の設定方法については、「CLUSTERPRO MC ProcessSaver 2.10 for Windows ユーザーズガイド (コンソール編)」を参照してください。

Pcheck の起動方法については、「CLUSTERPRO MC ProcessSaver 2.10 for Windows ユーザーズガイド (コンソール編)」および「CLUSTERPRO MC ProcessSaver 2.10 for Windows ユーザーズガイド (コマンド編)」を参照してください。

1. (続き)

作業を始める前に

テンプレートの各 Pfile の使い方については、テンプレート集に同梱の <Readme.txt> を参照してください。

2. Tomcat 6 の監視事例

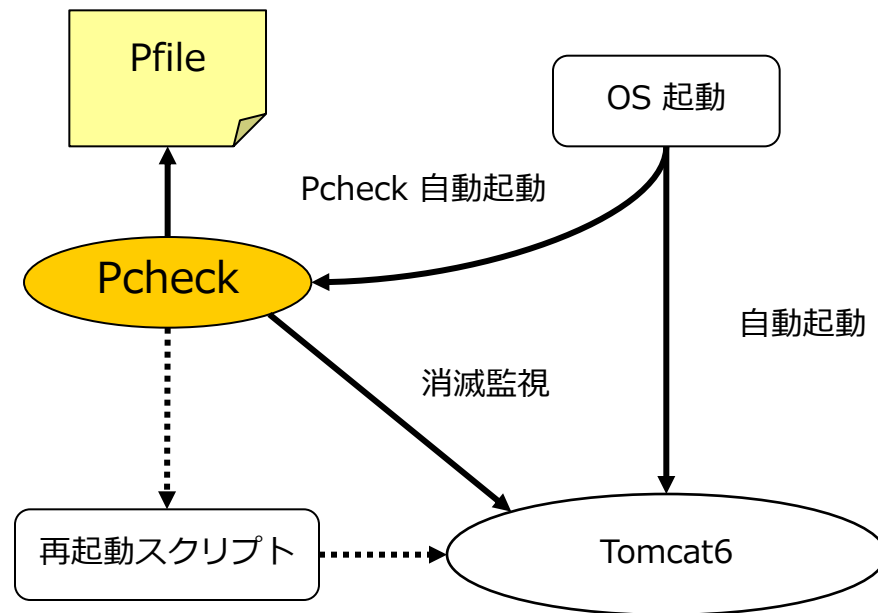
1. 概要

- 本事例は Tomcat 6.0.18 を使用しています。

タイプ	Service
表示名	Apache Tomcat
サービス名	Tomcat6
常駐プロセス	Tomcat6.exe
スタートアップ	自動
回復機能	何もしない
依存関係	なし

監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開



2. (続き)

2. Pfile

Pfile <sample_tomcat6>

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥tomcat6.dmp"  
UpMessageReduceMode   disable  
  
##### PENT #####  
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;  
ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
Tomcat6,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_tomcat6.bat",86400,3,Continue,Service
```

2. (続き)

3. 再起動スクリプト

再起動スクリプト <restart_tomcat6.bat>

```
@echo off
```

```
rem Tomcat 6.0.18 の再起動スクリプトです  
sc start tomcat6
```

```
}
```

sc コマンドを使用してサービスの起動を行います。

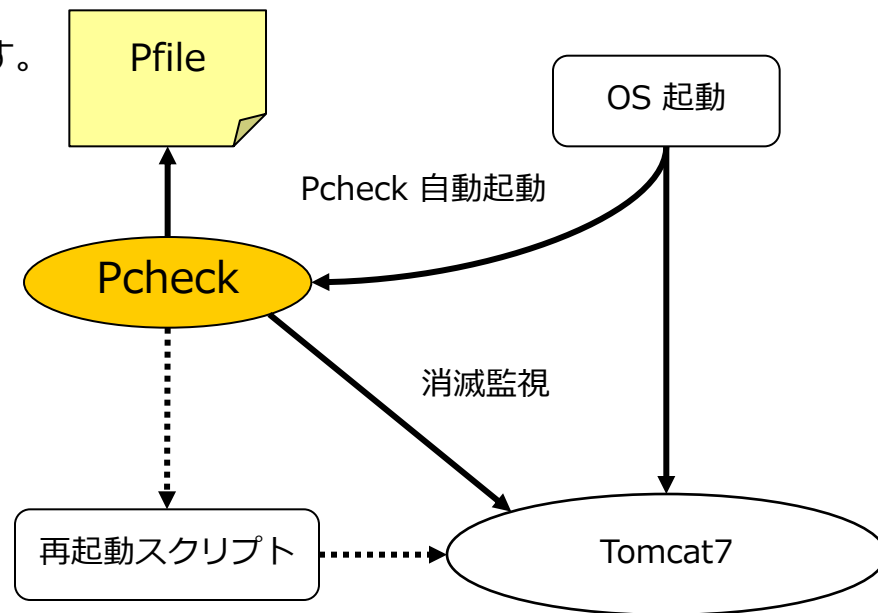
```
exit
```

3. Tomcat 7 の監視事例

1. 概要

- 本事例は Tomcat 7.0.42 を使用しています。

タイプ	Service
表示名	Apache Tomcat 7.0 Tomcat7
サービス名	Tomcat7
常駐プロセス	Tomcat7.exe
スタートアップ	自動
回復機能	何もしない
依存関係	なし



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

3. (続き)

2. Pfile

Pfile <sample_tomcat7>

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥tomcat7.dmp"  
UpMessageReduceMode   disable  
  
##### PENT #####  
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;  
ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
Tomcat7,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_tomcat7.bat",86400,3,Continue,Service
```

3. (続き)

3. 再起動スクリプト

再起動スクリプト <restart_tomcat7.bat>

```
@echo off
```

```
rem Tomcat 7.0.42 の再起動スクリプトです  
sc start tomcat7
```

} sc コマンドを使用してサービスの起動を行います。

```
exit
```

4. WebOTX 9.1 の監視事例

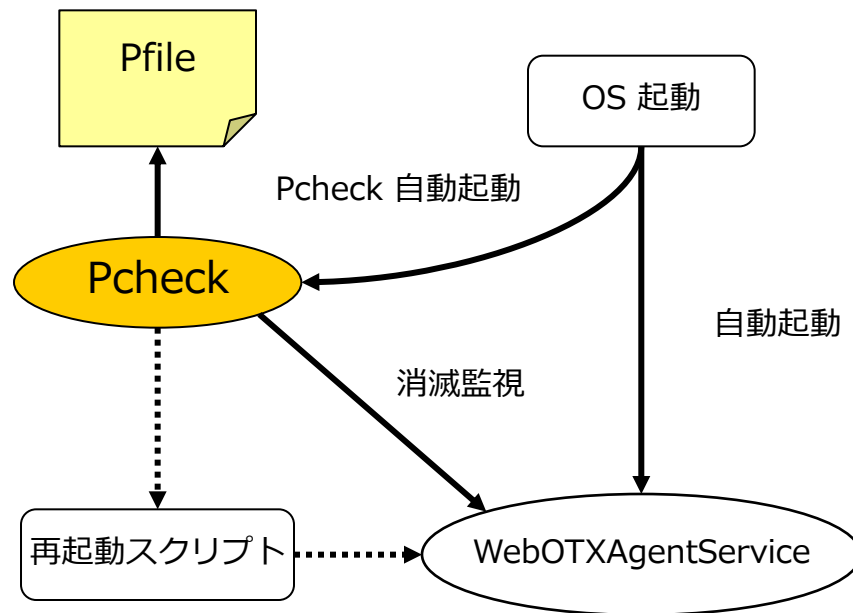
1. 概要

・本事例は WebOTX 9.11 を使用しています。

タイプ	Service
表示名	WebOTX AS Agent Service
サービス名	WebOTXAgentService
常駐プロセス	WOAgentSvc.exe
スタートアップ	自動
回復機能	何もしない
依存関係	なし

監視手順

- ・ OS 起動時にサービス自動起動
- ・ Pcheck による消滅監視
- ・ 消滅検出時には、再起動スクリプトで再開



※ WebOTX 8.21 と WebOTX 9.11 の構成は同じため、本テンプレートを WebOTX 8.21 でも同様に使用することができます。

4. (続き)

2. Pfile

Pfile <sample_webotxas>

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥webotxas.dmp"  
UpMessageReduceMode   disable  
  
##### PENT #####  
##  
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;  
ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
WebOTXAgentService,"C:¥Program  
Files¥HA¥ProcessSaver¥scriptfile¥restart_webotxas.bat",86400,3,Continue,Service
```

4. (続き)

3. 再起動スクリプト

再起動スクリプト <restart_webotxas.bat>

```
@echo off
```

```
rem WebOTX の再起動スクリプトです  
sc start WebOTXAgentService
```

} sc コマンドを使用してサービスの起動を行います。

```
exit
```

5. WebOTX 10.3 の監視事例

1. 概要

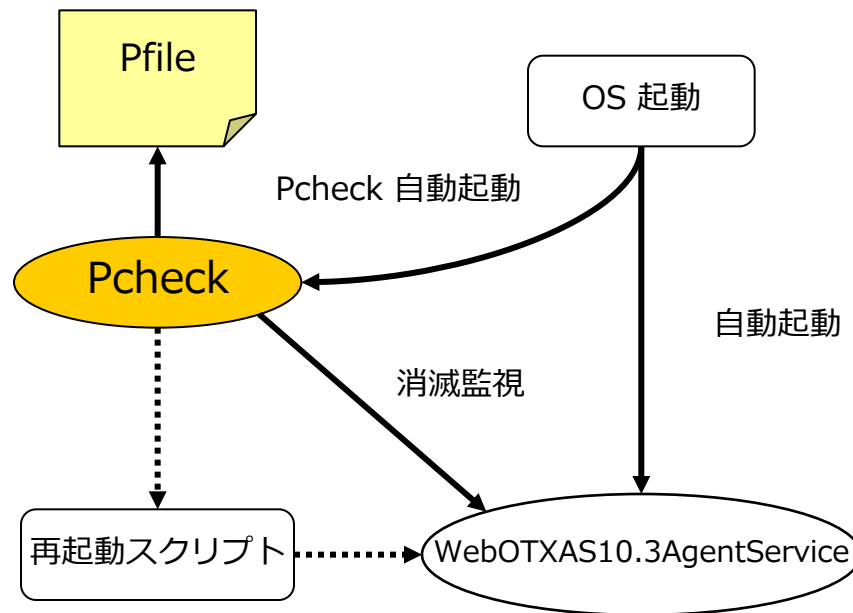
・ 本事例は WebOTX 10.31 を使用しています。

タイプ	Service
表示名	WebOTX AS 10.3 Agent Service
サービス名	WebOTXAS10.3AgentService
常駐プロセス	WOAgentSvc.exe
スタートアップ	自動
回復機能	何もしない
依存関係	なし

監視手順

- ・ OS 起動時にサービス自動起動
- ・ Pcheck による消滅監視
- ・ 消滅検出時には、再起動スクリプトで再開

※ WebOTX 10.3 からサービス名に製品バージョン番号が含まれるようになりましたので、本テンプレートは製品バージョン間の互換性はありません。



5. (続き)

2. Pfile

Pfile <sample_webotxas>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥webotxas.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch;StallCheck
WebOTXAS10.3AgentService,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_webotxas.bat",86400,3,Continue,Service
```

5. (続き)

3. 再起動スクリプト

再起動スクリプト <restart_webotxas.bat>

```
@echo off
```

```
rem WebOTX の再起動スクリプトです  
sc start WebOTXAS10.3AgentService }
```

sc コマンドを使用してサービスの起動を行います。

```
exit
```

6. WebLogic 11g の監視事例

1. 概要

- 本事例は WebLogic 11g (10.3) を使用しています。WebLogic のインストール Dir は <C:¥bea> です。

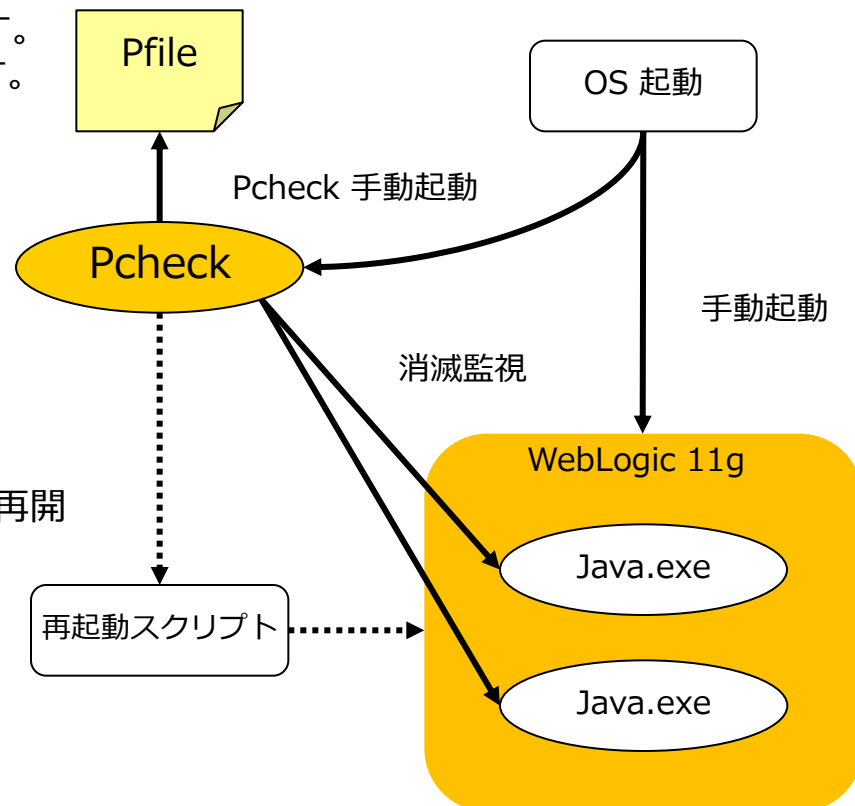
タイプ	Process
常駐プロセス	Java.exe

監視手順

- OS 起動後、手動にて起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

※ 注意事項

手動起動を前提に作成しています。
サービスに登録している場合は、
別途作成してください。



6. (続き)

2. Pfile

Pfile <sample_weblogic11g>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:\Program Files\HA\ProcessSaver\log\weblogic11g.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCm
d;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
java_exe,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_weblogic12c.bat",86400,3,Continue,Process;GroupTag=web;
IncludeStrings=-jrockit -Xdebug -Xnoagent -Xrunjdwp:transport=dt_socket;RestartWaitTime=60;PnameFullMatch=disable
java_exe,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_weblogic12c.bat",86400,3,Continue,Process;GroupTag=web;
IncludeStrings=com.pointbase.net.netServer /noconsole /port:9093 /d:0
/pointbase.ini-C:\bea\WORKSH~1.3\samples\domains\workshop\pointbase.ini;RestartWaitTime=60;PnameFullMatch=disabl
e
```

依存関係をもたせるために
GroupTag を指定して
グループ監視を行います。

ほかの Java プロセスと誤認しないように
IncludeStrings を指定して Java プロセスを
識別します。

6. (続き)

3. 再起動スクリプト

再起動スクリプト <restart_weblogic11g.bat>

```
@echo off

rem WebLogic11g の再起動スクリプトです

rem WebLogic の停止コマンドを実行します
start call "C:¥bea¥workshop_10.3¥samples¥domains¥workshop¥bin¥stopWebLogic.cmd"

rem 任意の秒数の間待ちます(指定範囲 : 1 ~ 600)
"C:¥Program Files¥HA¥ProcessSaver¥bin¥PSSleep.exe" 30 }

rem WebLogic の起動コマンドを実行します
start call "C:¥bea¥workshop_10.3¥samples¥domains¥workshop¥startWebLogic.cmd"

exit
```

停止コマンドが終了するまで起動コマンドを待ち合わせます。
環境によって待ち合わせ時間が異なりますので調整してください。

7. WebLogic 12c の監視事例

1. 概要

- 本事例は WebLogic 12c (12.1.2) を使用しています。
WebLogic のインストール Dir は <C:¥Oracle¥Middleware¥Oracle_Home> です。

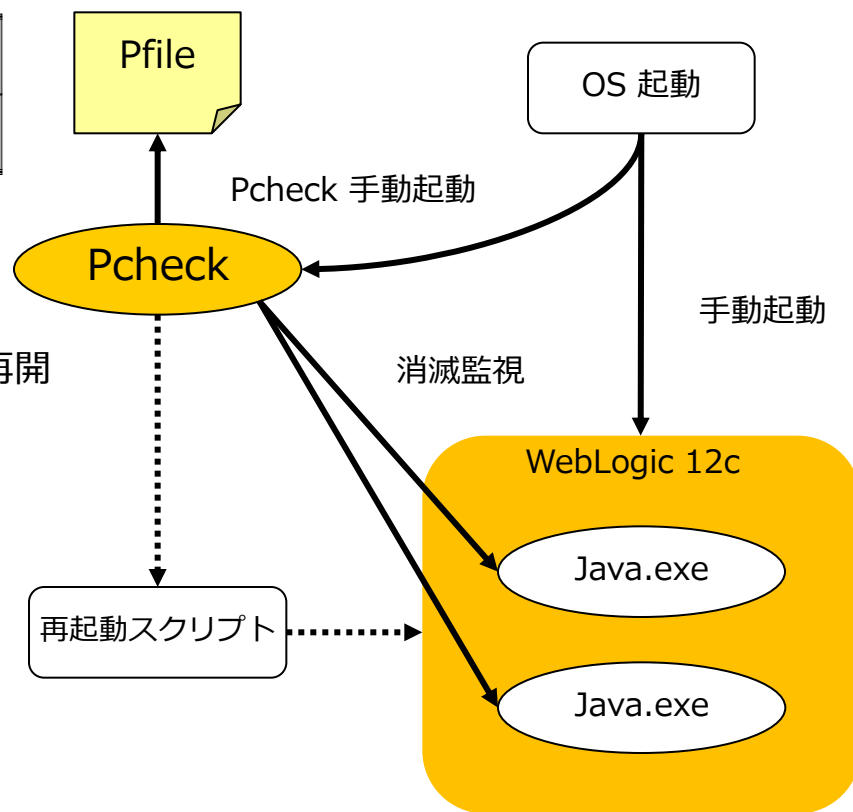
タイプ	Process
常駐プロセス	Java.exe

監視手順

- OS 起動後、手動にて起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

※ 注意事項

手動起動を前提に作成しています。
サービスに登録している場合は、
別途作成してください。



2. Pfile

Pfile <sample_weblogic12c>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval      10
DumpFileName          "C:\Program Files\HA\ProcessSaver\log\weblogic12c.dmp"
UpMessageReduceMode  disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;Group Tag;ClearCm
d;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
java.exe,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_weblogic12c.bat",86400,3,Continue,Process,Group Tag=web;
IncludeStrings=-
Djava.security.policy=C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\server\lib\weblogic.policy;RestartWaitTime=60;PnameFullM
atch=disable
java.exe,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_weblogic12c.bat",86400,3,Continue,Process,Group Tag=web;
IncludeStrings=-classpath
"C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\common\derby/lib/derby.jar;C:\Oracle\MIDDLE~1\ORACLE~1\wlserver\commo
n\derby/lib/derbynet.jar;RestartWaitTime=60;PnameFullMatch=disable
```

依存関係をもたせるために
GroupTag を指定して
グループ監視を行います。

ほかの Java プロセスと誤認しないように
IncludeStrings を指定して Java プロセスを識別します。

※ IncludeStrings に指定する値は、環境により適宜修正してください。

3. 再起動スクリプト

再起動スクリプト <restart_weblogic12c.bat>

```
@echo off

rem WebLogic12c の再起動スクリプトです

rem WebLogic の停止コマンドを実行します
start call
"C:¥Oracle¥Middleware¥Oracle_Home¥user_projects¥domains¥base_domain¥bin¥stopWebLogic.cmd"

rem 任意の秒数の間待ちます(指定範囲：1 ～ 600)
"C:¥Program Files¥HA¥ProcessSaver¥bin¥PSSsleep.exe" 30 }

rem WebLogic の起動コマンドを実行します
start call
"C:¥Oracle¥Middleware¥Oracle_Home¥user_projects¥domains¥base_domain¥bin¥startWebLogic.cmd"

exit
```

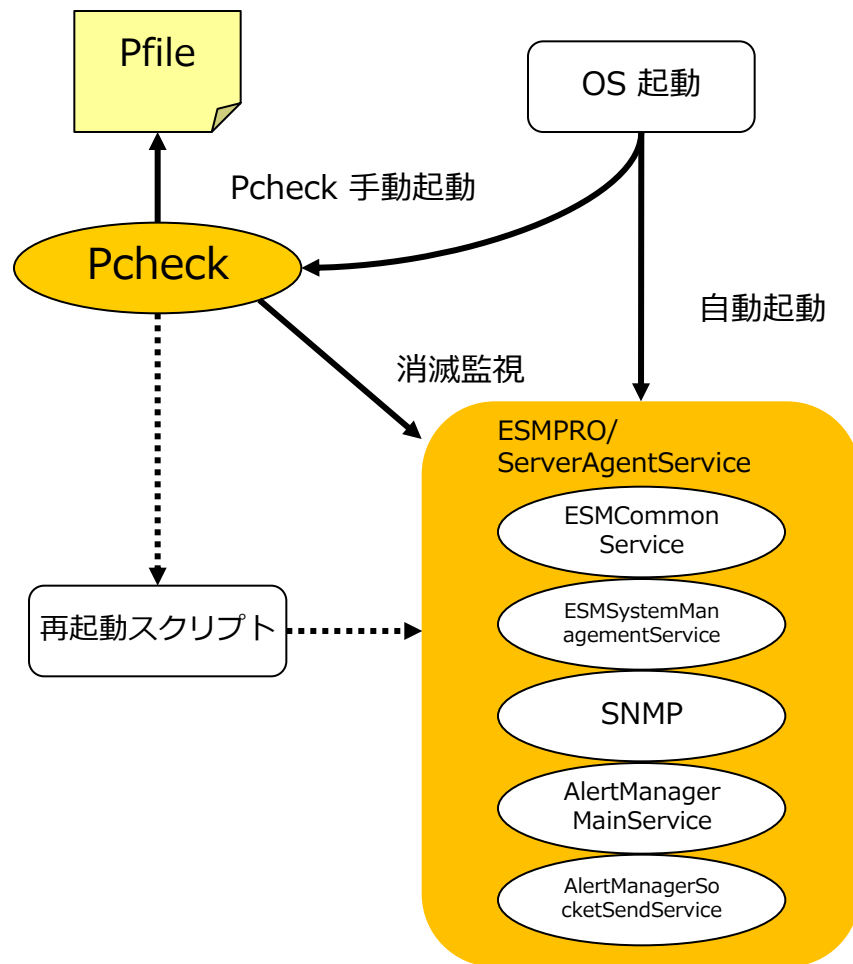
停止コマンドが終了するまで起動コマンドを待ち合わせます。
環境によって待ち合わせ時間が異なりますので調整してください。

※ WebLogic の停止/起動コマンドに指定するフォルダーは、環境により適宜修正してください。

8. ESMPRO/ServerAgentService Ver.2 の監視事例

1. 概要

- ・ 本事例は ESMPRO/ServerAgentService Ver.2 を使用しています。



監視手順

- ・ OS 起動時にサービス自動起動
- ・ Pcheck による消滅監視
- ・ 消滅検出時には、再起動スクリプトで再開

8. (続き)

タイプ	Service	Service	Service	Service	Service
表示名	ESMCommonService	ESM System Management Service	SNMP サービス	Alert Manager Main Service	Alert Manager Socket(S) Service
サービス名	ESMCommonService	ESMSystemManagementService ※1	SNMP	AlertManagerMainService	AlertManagerSocketSendService
常駐プロセス	ESMCommon.exe	esmsmsrv.exe	snmp.exe	AMVMMain.exe	amvscks.exe
スタートアップ	自動(遅延開始)	自動	自動	自動(遅延開始)	手動 ※2
回復機能	サービスを再起動する	サービスを再起動する※3 何もしない ※4	サービスを再起動する	サービスを再起動する	サービスを再起動する
依存関係	なし	なし ※3 Winmgmt ※4	なし	なし	なし

※1 IPMI、iLO 対応機種の場合にインストールされます。

※2 マネージャ通報(TCP/IP In-Band)、マネージャ通報(TCP/IP Out-of-Band)のどちらかが有効の場合に起動されます。

※3 iLO 対象装置の場合です。

※4 BMC 対象装置の場合です。

8. (続き)

2. Pfile

Pfile <sample_esmprosas>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥esmprosas.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch;StallCheck
ESMCommonService,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_esmprosas.bat",86400,3,Continue,Service;GroupTag=esmprosas
#ESMSystemManagementService,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_esmprosas.bat",86400,3,Continue,Service;GroupTag=esmprosas
SNMP,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_esmprosas.bat",86400,3,Continue,Service;GroupTag=esmprosas
AlertManagerMainService,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_esmprosas.bat",86400,3,Continue,Service;GroupTag=esmprosas
#AlertManagerSocketSendService,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_esmprosas.bat",86400,3,Continue,Service;GroupTag=esmprosas
```

依存関係をもたせるために
GroupTag を指定して
グループ監視を行います。

ESMSystemManagementService、AlertManagerSocketSendService
を監視する場合は # を削除してコメントインしてください。

8. (続き)

3. 再起動スクリプト

再起動スクリプト <restart_esmprosas.bat>

```
@echo off

rem ESMPRO/ServerAgentService の再起動スクリプトです

rem ESMPRO/ServerAgentService を停止します
net stop ESMCommonService
rem net stop ESMSystemManagementService
net stop AlertManagerMainService
rem net stop AlertManagerSocketSendService
net stop SNMP

rem ESMPRO/ServerAgentService を起動します
net start ESMCommonService
rem net start ESMSystemManagementService
net start SNMP
net start AlertManagerMainService
rem net start AlertManagerSocketSendService

exit
```

ESMSystemManagementService、AlertManagerSocketSendService
を監視する場合は rem を削除してコメントインしてください。

9. WebSAM JobCenter 15.4 の監視事例

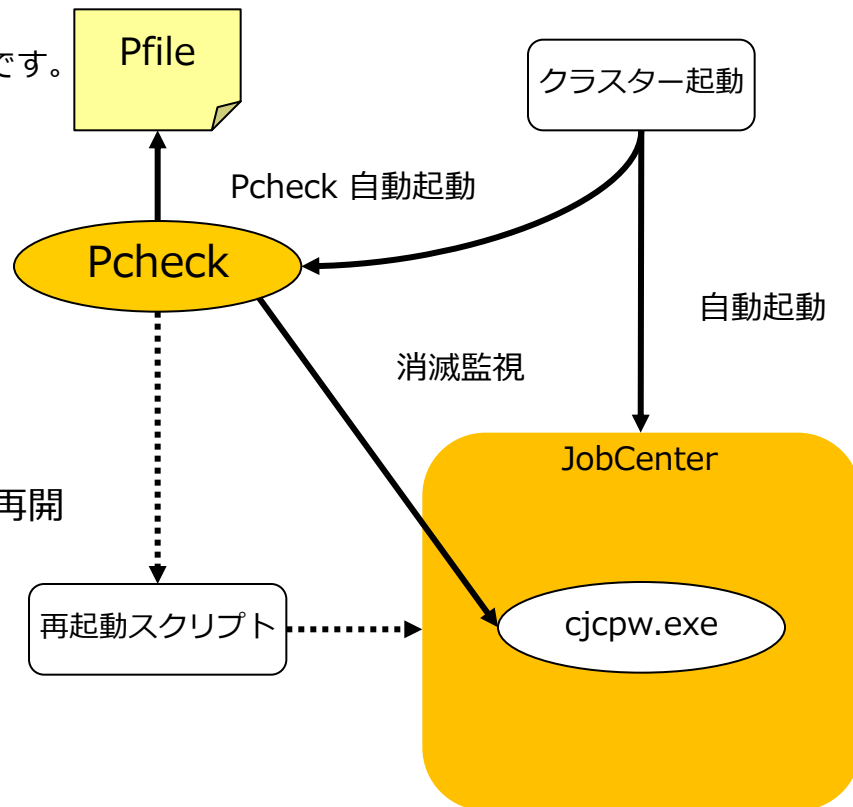
1. 概要

- 本事例は WebSAM JobCenter 15.4 を使用しています。
JobCenter のインストール Dir は <C:¥JobCenter¥SV> です。

タイプ	Process
常駐プロセス	cjcpw.exe

監視手順

- ・ クラスター起動時にプロセス自動起動
- ・ Pcheck による消滅監視
- ・ 消滅検出時には、再起動スクリプトで再開



9. (続き)

2. Pfile

Pfile <sample_websamjc>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥websamjc.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch;StallCheck
cjcpw.exe,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_websamjc.bat",86400,3,Shutdown,Process
```

3. 再起動スクリプト

再起動スクリプト <restart_websamjc.bat>

```
@echo off

rem WebSAM JobCenter の再起動スクリプトです

rem WebSAM JobCenter を停止します
C:¥JobCenter¥SV¥bin¥cluster¥cjcpcw -stop site-name
TIMEOUT /T 10 /NOBREAK>nul

rem WebSAM JobCenter を起動します
start C:¥JobCenter¥SV¥bin¥cluster¥cjcpcw -u Administrator site-name db-directory
TIMEOUT /T 10 /NOBREAK>nul

exit
```

停止コマンドが終了するまで起動コマンドを待ち合わせます。
環境によって待ち合わせ時間が異なりますので調整してください。

JobCenter 管理者を環境に合わせて指定してください。

サイト名とデータベースディレクトリ名は環境に合わせて指定してください。

10. WebSAM SystemManager G 10.1 の監視事例

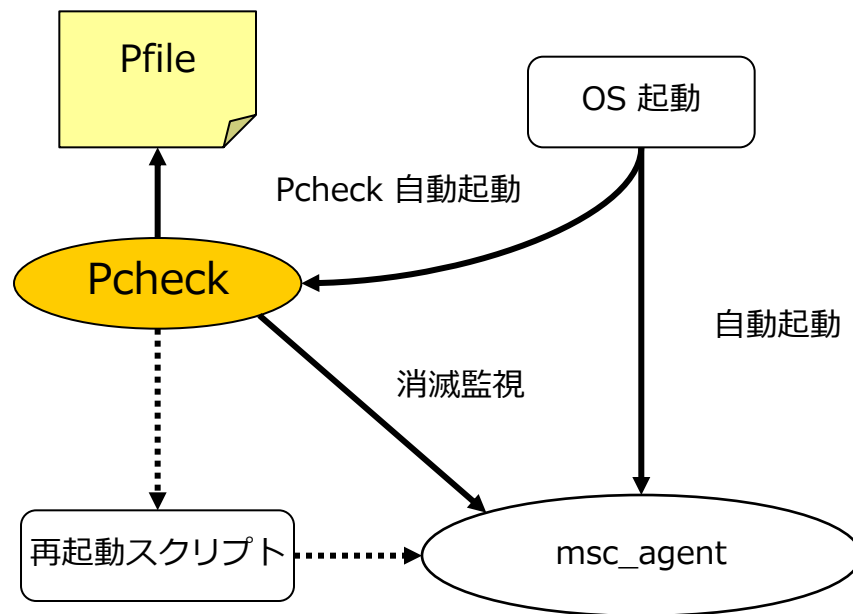
1. 概要

- ・本事例は WebSAM SystemManager G 10.1 エージェントを使用しています。

タイプ	Service
表示名	SystemManager G Agent Service
サービス名	mhc_agent
常駐プロセス	mhc_opm.exe
スタートアップ	自動
回復機能	何もしない
依存関係	なし

監視手順

- ・ OS 起動時にサービス自動起動
- ・ Pcheck による消滅監視
- ・ 消滅検出時には、再起動スクリプトで再開



10. (続き)

2. Pfile

Pfile <sample_websamsmg_agent>

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥websamsmg_agent.dmp"  
UpMessageReduceMode   disable  
  
##### PENT #####  
##  
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch;StallCheck  
msc_agent,"C:¥Program  
Files¥HA¥ProcessSaver¥scriptfile¥restart_websamsmg_agent.bat",86400,3,Continue,Service
```

10. (続き)

3. 再起動スクリプト

再起動スクリプト <restart_websamsmg_agent.bat>

```
@echo off
```

```
rem WebSAM SystemManager G Agent の再起動スクリプトです
```

```
net start msc_agent
```

} net コマンドを使用してサービスの起動を行います。

```
exit
```

11. Oracle 10g の監視事例

- ・ 本事例は Oracle 10g R2 を使用しています。
Oracle にて提供されている以下のサービスを監視する場合を紹介します。

(1) システム一意

Oracle をインストールした際に登録されるサービスです。

Oracle 一意のサービスのため、複数登録されません。

【HOME_NAME】は、Oracle のインストール時に指定するホーム名です。

- ・ Oracle 【HOME_NAME】 iSQL*Plus (OracleOraDb10g_home1iSQL*Plus)
- ・ Oracle 【HOME_NAME】 TNSListener (OracleOraDb10g_home1TNSListener)

*本書では、【HOME_NAME】 (ホーム名) を “OraDb10g_home1” とします。

(2) インスタンス

DB を作成した際に登録されるサービスです。

DB 数分サービスが登録されます。

【SID】は、Oracle のデータベース作成時に指定するデータベース名です。

- ・ OracleDBConsole 【SID 名】 (OracleDBConsoleorcl)
- ・ OracleService 【SID 名】 (OracleServiceORCL)

*本書では、【SID 名】 (データベース名) を “orcl” とします。

11. (続き)

1. システム一意

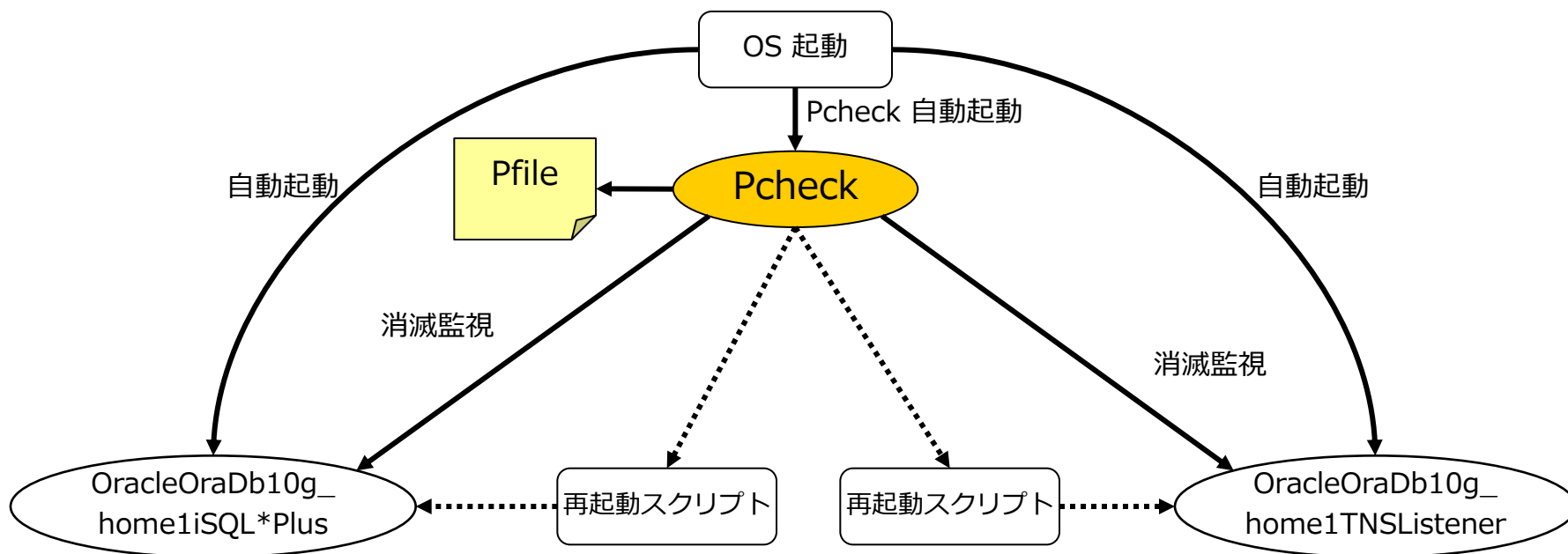
1-1. 概要

- OracleOraDb10g_home1iSQL*Plus
- OracleOraDb10g_home1TNSListener

タイプ	Service	Service
表示名	OracleOraDb10g_home1iSQL*Plus	OracleOraDb10g_home1TNSListener
サービス名	OracleOraDb10g_home1iSQL*Plus	OracleOraDb10g_home1TNSListener
常駐プロセス	isqlplussvc.exe	TNSLSNR.EXE
スタートアップ	自動	自動
回復機能	何もしない	何もしない
依存関係	なし	なし

11. (続き)

1-1. (続き)



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

11. (続き)

1-2. Pfile

Pfile <sample_oracle10g_system>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥oracle10g_system.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;Group
Tag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
OracleOraDb10g_home1iSQL*Plus,"C:¥ProgramFiles¥HA¥ProcessSaver¥scriptfile¥restart_sqlplus_10g.bat",86400,3,
Continue,Service
OracleOraDb10g_home1TNSListener,"C:¥ProgramFiles¥HA¥ProcessSaver¥scriptfile¥restart_tnslistener_10g.bat",864
00,3,Continue,Service
```

Oracle インストール時に指定したホーム名に変更してください。

OracleOraDb10g_home1iSQL*Plus → Oracle 【HOME_NAME】 iSQL*Plus

OracleOraDb10g_home1TNSListener → Oracle 【HOME_NAME】 TNSListener

11. (続き)

1-3. 再起動スクリプト

- OracleOraDb10g_home1iSQL*Plus <restart_sqlplus_10g.bat>

```
@echo off
rem iSQL*Plus の再起動スクリプトです。
sc start OracleOraDb10g_home1iSQL*Plus
exit
```

} sc コマンドを使用してサービスの起動を行います。

- OracleOraDb10g_home1TNSListener <restart_tnslistener_10g.bat>

```
@echo off
rem TNSListener の再起動スクリプトです。
sc start OracleOraDb10g_home1TNSListener
exit
```

} sc コマンドを使用してサービスの起動を行います。

Oracle インストール時に指定したホーム名に変更してください。

OracleOraDb10g_home1iSQL*Plus	→	Oracle 【HOME_NAME】 iSQL*Plus
OracleOraDb10g_home1TNSListener	→	Oracle 【HOME_NAME】 TNSListener

11. (続き)

2. インスタンス

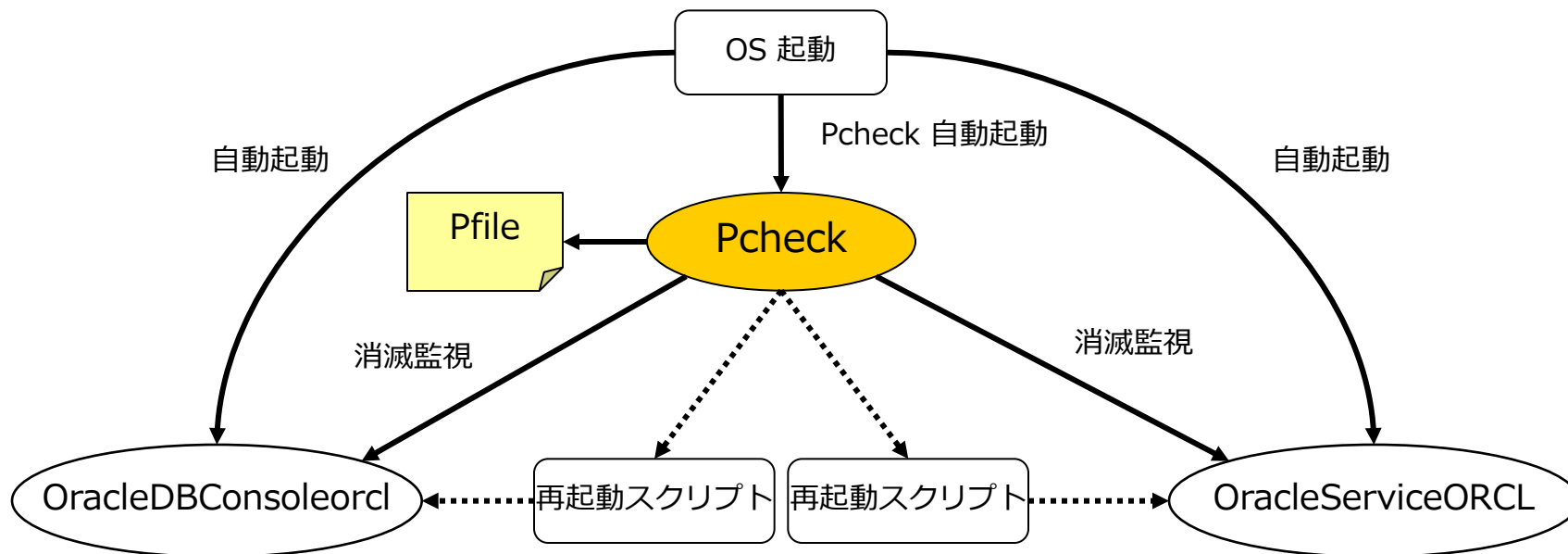
2-1. 概要

- OracleDBConsoleorcl
- OracleServiceORCL

タイプ	Service	Service
表示名	OracleDBConsoleorcl	OracleServiceORCL
サービス名	OracleDBConsoleorcl	OracleServiceORCL
常駐プロセス	nmesrvc.exe emagent.exe perl.exe java.exe cmd.exe	oracle.exe
スタートアップ	自動	自動
回復機能	何もしない	何もしない
依存関係	なし	なし

11. (続き)

2-1. (続き)



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

11. (続き)

2-2. Pfile

Pfile <sample_oracle10g_instance>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥oracle10g_instance.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
OracleDBConsoleorcl,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_oracle10gdbconsole.bat",86400,3,Continue,Service;RestartWaitTime=600
OracleServiceORCL,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_oracle10gsvc.bat",86400,3,Continue,Service
```

データベース作成時に指定したデータベース名に変更してください。

OracleDBConsoleorcl	→	OracleDBConsole 【SID名】
OracleServiceORCL	→	OracleService 【SID名】

11. (続き)

2-3. 再起動スクリプト

- OracleDBConsoleorcl <restart_oracle10gdbconsole_base.bat>

```
@echo off

rem OracleDBConsoleの再起動スクリプトです。
rem 起動の前処理
cd C:¥WINDOWS¥system32
wmic process where "name = ¥"java.exe¥" and commandline = ¥"【引数】 ¥" delete

rem 起動処理
sc start OracleDBConsoleorcl
exit
```

Java プロセスを終了します。

sc コマンドを使用してサービスの起動を行います。

【引数】の取得方法については次ページを参照してください。
※ この再起動スクリプトの内容は環境によって異なります。

データベース作成時に指定したデータベース名に変更してください。
OracleDBConsoleorcl → OracleDBConsole 【SID名】

11. (続き)

2-3. (続き)

- OracleDBConsoleorcl

wmic process where "name = ¥"java.exe¥"" get commandline の実行結果にて下記の引数が取得されます。

```
【インストール Dir】 ¥jdk/bin/java -server -Xmx【全体の最大ヒープサイズ】 -XX:MaxPermSize=
【Permanent 領域Max サイズ】 -XX:MinHeapFreeRatio=【Gorbage Collector 後に拡張されるまでに消費
されるヒープ比率】 -XX:MaxHeapFreeRatio=【Gorbage Collector 後にシュリンクされるまでに消費され
るヒープ比率】 -DORACLE_HOME=【インストール Dir】 -Doracle.home=【インストール Dir】 /oc4j -
Doracle.oc4j.localhome=【インストール Dir】 ¥【コンピューター名】 _【SID 名】 /sysman -
DEMSTATE=【インストール Dir】 ¥¥【コンピューター名】 _【SID 名】 -
Doracle.j2ee.dont.use.memory.archive=true -Djava.protocol.handler.pkgs=HTTPClient -
Doracle.security.jazn.config=【インストール Dir】 /oc4j/j2ee/OC4J_DBCConsole_【コンピューター名】
_【SID 名】 /config/jazn.xml -Doracle.security.policy=【インストール Dir】
/oc4j/j2ee/OC4J_DBCConsole_【コンピューター名】 _【SID 名】 /config/java2.policy -
Doracle.security.properties=【インストール Dir】 /oc4j/j2ee/home/config/jazn.security.props -
DEMROOT=【インストール Dir】 ¥¥【コンピューター名】 _【SID 名】 -Dsysman.md5password=true
-Drepapi.oracle.home=【インストール Dir】 -Ddisable.checkForUpdate=true -
Djava.awt.headless=true -jar 【インストール Dir】 /oc4j/j2ee/home/oc4j.jar -config 【インストール
Dir】 oc4j/j2ee/OC4J_DBCConsole_【コンピューター名】 _【SID 名】 /config/server.xml
```

※ 注意事項

【インストール Dir】 oc4j/j2ee/OC4J_DBCConsole_【コンピューター名】 _【SID 名】 /config/server.xml
があるものを記載してください。

11. (続き)

2-3. (続き)

- OracleDBConsoleorcl

<restart_oracle10gdbconsole.bat>

```
@echo off
```

```
rem OracleDBConsoleの再起動スクリプトです。  
rem 起動の前処理
```

```
cd C:%WINDOVS%system32
```

```
wmic process where "name = %"java.exe%" and commandline = %"C:%oracle%product%10.2.0%db_1%jdk/bin/java -server -  
Xmx256M -XX:MaxPermSize=96m -XX:MinHeapFreeRatio=20 -XX:MaxHeapFreeRatio=40 -  
DORACLE_HOME=C:%oracle%product%10.2.0%db_1 -Doracle.home=C:%oracle%product%10.2.0%db_1/oc4j -  
Doracle.oc4j.localhome=C:%oracle%product%10.2.0%db_1%Win2003Oracle_orcl/sysman -  
DEMSTATE=C:%oracle%product%10.2.0%db_1%Win2003Oracle_orcl -Doracle.j2ee.dont.use.memory.archive=true -  
Djava.protocol.handler.pkgs=HTTPClient -  
Doracle.security.jazn.config=C:%oracle%product%10.2.0%db_1/oc4j/j2ee/OC4J_DBConsole_Win2003Oracle_orcl/config/jazn.x  
ml -  
Djava.security.policy=C:%oracle%product%10.2.0%db_1/oc4j/j2ee/OC4J_DBConsole_Win2003Oracle_orcl/config/java2.policy -  
-Djava.security.properties=C:%oracle%product%10.2.0%db_1/oc4j/j2ee/home/config/jazn.security.props -  
DEMDROOT=C:%oracle%product%10.2.0%db_1%Win2003Oracle_orcl -Dsysman.md5password=true -  
Drepapi.oracle.home=C:%oracle%product%10.2.0%db_1 -Ddisable.checkForUpdate=true -Djava.awt.headless=true -jar  
C:%oracle%product%10.2.0%db_1/oc4j/j2ee/home/oc4j.jar -config  
C:%oracle%product%10.2.0%db_1/oc4j/j2ee/OC4J_DBConsole_Win2003Oracle_orcl/config/server.xml%" delete
```

```
rem 起動処理
```

```
sc start OracleDBConsoleorcl  
exit
```

sc コマンドを使用してサービスの起動を行います。

例) 環境が以下の場合の再起動スクリプト

コンピューター名	: Win2003Oracle
インストール Dir	: C:%oracle%product%10.2.0%db_1
SID 名	: orcl
Xmx	: 256M
MaxPermSize	: 96M
MinHeapFreeRatio	: 20
MaxHeapFreeRatio	: 40

データベース作成時に指定したデータベース名に変更してください。
OracleDBConsoleorcl → OracleDBConsole 【SID名】

11. (続き)

2-3. (続き)

- OracleServiceORCL <restart_oracle10gsvc.bat>

```
@echo off
rem OracleService の再起動スクリプトです。 }
sc start OracleServiceORCL
exit
```

sc コマンドを使用してサービスの起動を行います。

データベース作成時に指定したデータベース名に変更してください。
OracleServiceORCL → OracleService 【SID名】

12. Oracle 11g の監視事例

- ・ 本事例は Oracle 11g R2 を使用しています。
Oracle にて提供されている以下のサービスを監視する場合を紹介します。

(1) システム一意

Oracle をインストールした際に登録されるサービスです。

Oracle 一意のサービスのため、複数登録されません。

【HOME_NAME】は、Oracle のインストール時に指定するホーム名です。

- ・ Oracle 【HOME_NAME】 TNSListener (OracleOraDb11g_home1TNSListener)

*本書では、【HOME_NAME】 (ホーム名) を “OraDb11g_home1” とします。

(2) インスタンス

DB を作成した際に登録されるサービスです。

DB 数分サービスが登録されます。

【SID】は、Oracle のデータベース作成時に指定するデータベース名です。

- ・ OracleDBConsole 【SID 名】 (OracleDBConsoleorcl)
- ・ OracleService 【SID 名】 (OracleServiceORCL)

*本書では、【SID 名】 (データベース名) を “orcl” とします。

12. (続き)

1. システム一意

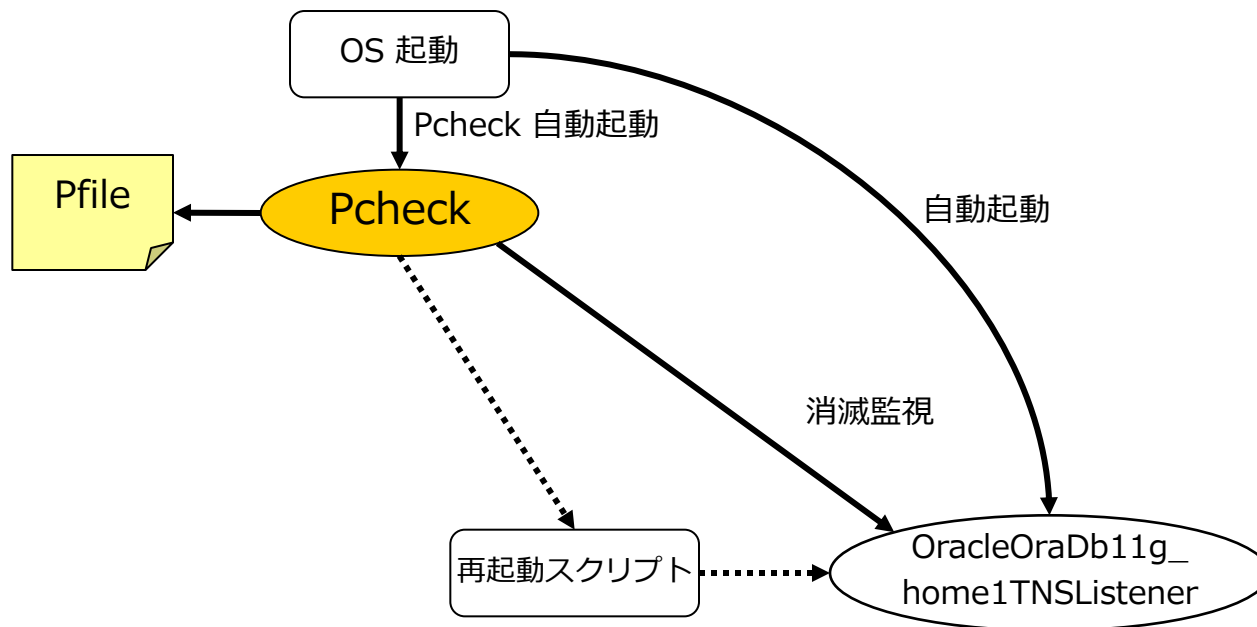
1-1. 概要

- OracleOraDb11g_home1TNSListener

タイプ	Service
表示名	OracleOraDb11g_home1TNSListener
サービス名	OracleOraDb11g_home1TNSListener
常駐プロセス	TNSLNR.EXE
スタートアップ	自動
回復機能	何もしない
依存関係	なし

12. (続き)

1-1. (続き)



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

12. (続き)

1-2. Pfile

Pfile <sample_oracle11g_system>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥oracle11g_system.dmp"
UpMessageReduceMode   disable

##### PENT #####
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;Group
Tag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
OracleOraDb11g_home1TNSListener, "C:¥ProgramFiles¥HA¥ProcessSaver¥scriptfile¥restart_tnslistener_11g.bat",86400,3,C
ontinue,Service
```

Oracle インストール時に指定したホーム名に変更してください。
OracleOraDb11g_home1TNSListener → Oracle 【HOME_NAME】 TNSListener

12. (続き)

1-3. 再起動スクリプト

- OracleOraDb11g_home1TNSListener <restart_tnslistener_11g.bat>

```
@echo off  
  
rem TNSListener の再起動スクリプトです。  
sc start OracleOraDb11g_home1TNSListener }  
  
exit
```

sc コマンドを使用してサービスの起動を行います。

Oracle インストール時に指定したホーム名に変更してください。
OracleOraDb11g_home1TNSListener → Oracle 【HOME_NAME】 TNSListener

12. (続き)

2. インスタンス

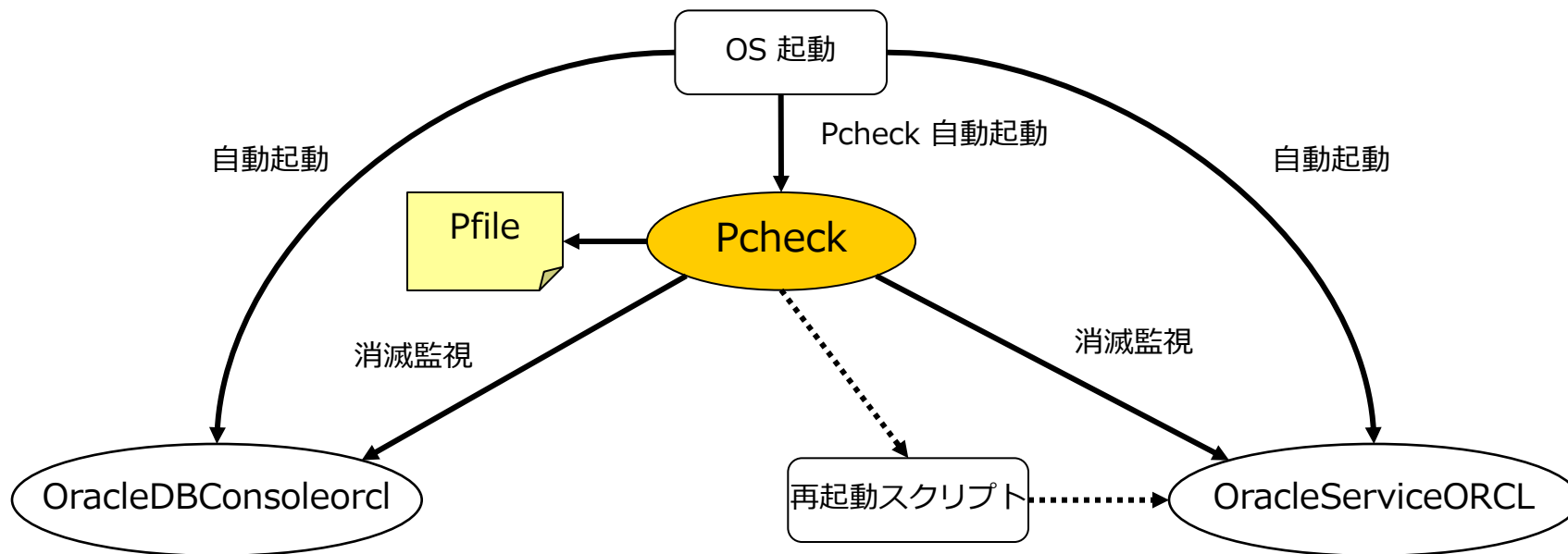
2-1. 概要

- OracleDBConsoleorcl
- OracleServiceORCL

タイプ	Service	Service
表示名	OracleDBConsoleorcl	OracleServiceORCL
サービス名	OracleDBConsoleorcl	OracleServiceORCL
常駐プロセス	nmesrvc.exe emagent.exe perl.exe java.exe cmd.exe	oracle.exe
スタートアップ	自動	自動
回復機能	何もしない	何もしない
依存関係	なし	なし

12. (続き)

2-1. (続き)



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- OracleServiceORCL のみ消滅検出時には、再起動スクリプトで再開

12. (続き)

2-2. Pfile

Pfile <sample_oracle11g_instance>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥oracle11g_instance.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
OracleDBConsoleorcl,86400,0,Continue,Service
OracleServiceORCL,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_oracle11gsvc.bat",86400,3,Continue,Service
```

データベース作成時に指定したデータベース名に変更してください。

OracleDBConsoleorcl → OracleDBConsole 【SID名】

OracleServiceORCL → OracleService 【SID名】

12. (続き)

2-3. 再起動スクリプト

- OracleServiceORCL <restart_oracle11gsvc.bat>

```
@echo off
```

```
rem OracleService の再起動スクリプトです。 }
```

```
sc start OracleServiceORCL
```

```
exit
```

sc コマンドを使用してサービスの起動を行います。

データベース作成時に指定したデータベース名に変更してください。
OracleServiceORCL → OracleService 【SID名】

13. Oracle 12c の監視事例

- ・ 本事例は Oracle 12c を使用しています。
Oracle にて提供されている以下のサービスを監視する場合を紹介します。

(1) システム一意

Oracle をインストールした際に登録されるサービスです。

Oracle 一意のサービスのため、複数登録されません。

【HOME_NAME】は、Oracle のインストール時に指定するホーム名です。

- ・ Oracle 【HOME_NAME】 TNSListener (OracleOraDB12Home1TNSListener)

*本書では、【HOME_NAME】 (ホーム名) を “OraDB12Home1” とします。

(2) インスタンス

DB を作成した際に登録されるサービスです。

DB 数分サービスが登録されます。

【SID】は、Oracle のデータベース作成時に指定するデータベース名です。

- ・ OracleService 【SID 名】 (OracleServiceORCL)
- ・ OracleVssWriter 【SID 名】 (OracleVssWriterORCL)

*本書では、【SID 名】 (データベース名) を “orcl” とします。

13. (続き)

1. システム一意

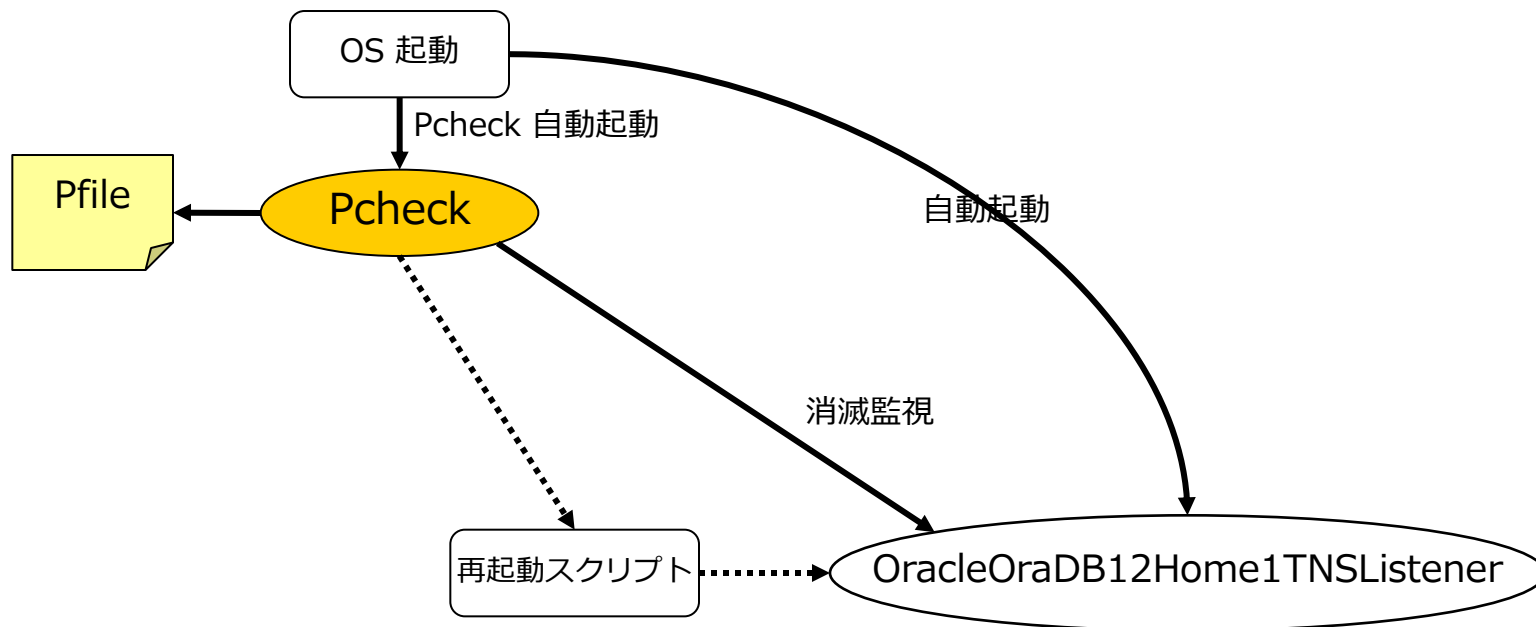
1-1. 概要

- OracleOraDB12Home1TNSListener

タイプ	Service
表示名	OracleOraDB12Home1TNSListener
サービス名	OracleOraDB12Home1TNSListener
常駐プロセス	TNSLSNR.EXE
スタートアップ	自動
回復機能	何もしない
依存関係	なし

13. (続き)

1-1. (続き)



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

13. (続き)

1-2. Pfile

Pfile <sample_oracle12c_system>

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥oracle12c_system.dmp"
UpMessageReduceMode   disable

##### PENT #####
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;Group
Tag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
OracleOraDB12Home1TNSListener,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_tnslistener_12c.bat",86400,3,Continue,Service
```

Oracle インストール時に指定したホーム名に変更してください。
OracleOraDB12Home1TNSListener → Oracle 【HOME_NAME】 TNSListener

13. (続き)

1-3. 再起動スクリプト

- OracleOraDB12Home1TNSListener<restart_tnslistener_12c.bat>

```
@echo off  
rem TNSListener の再起動スクリプトです。  
sc start OracleOraDB12Home1TNSListener }  
exit
```

sc コマンドを使用してサービスの起動を行います。

Oracle インストール時に指定したホーム名に変更してください。
OracleOraDB12Home1TNSListener → Oracle 【HOME_NAME】 TNSListener

13. (続き)

2. インスタンス

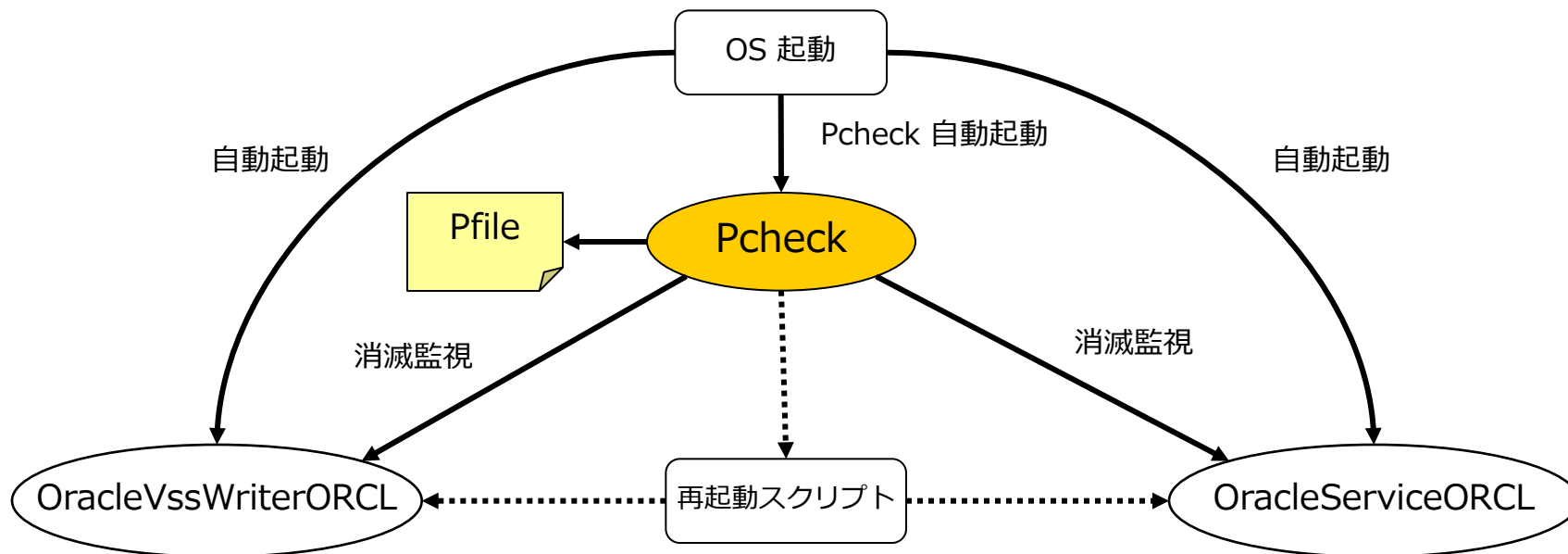
2-1. 概要

- OracleVssWriterORCL
- OracleServiceORCL

タイプ	Service	Service
表示名	OracleVssWriterORCL	OracleServiceORCL
サービス名	OracleVssWriterORCL	OracleServiceORCL
常駐プロセス	oravssw.exe	oracle.exe
スタートアップ	自動	自動
回復機能	何もしない	何もしない
依存関係	なし	なし

13. (続き)

2-1. (続き)



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開

13. (続き)

2-2. Pfile

Pfile <sample_oracle12c_instance>

```
##### PARAM #####  
MsgCheckInterval          5  
MonitorInterval           10  
DumpFileName              "C:\Program Files\HA\ProcessSaver\log\oracle12c_instance.dmp"  
UpMessageReduceMode       disable  
  
##### PENT #####  
##  
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
OracleVssWriterORCL,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_vsswriter12c.bat",86400,3,Continue,Service  
OracleServiceORCL,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_oracle12csvc.bat",86400,3,Continue,Service
```

データベース作成時に指定したデータベース名に変更してください。

OracleVssWriterORCL → OracleVssWriter 【SID名】

OracleServiceORCL → OracleService 【SID名】

13. (続き)

2-3. 再起動スクリプト

- OracleServiceORCL <restart_oracle12csvc.bat>

```
@echo off  
rem OracleService の再起動スクリプトです。 }  
sc start OracleServiceORCL  
exit
```

sc コマンドを使用してサービスの起動を行います。

- OracleVssWriterORCL <restart_vsswriter12c.bat>

```
@echo off  
rem VssWriterService の再起動スクリプトです。 }  
sc start OracleVssWriterORCL  
exit
```

sc コマンドを使用してサービスの起動を行います。

データベース作成時に指定したデータベース名に変更してください。
OracleServiceORCL → OracleService 【SID名】
OracleVssWriterORCL → OracleVssWriter 【SID名】

14. IIS6.0 の監視事例

1. 概要

本事例は IIS6.0 を使用しています。
IIS6.0 は Windows Server 2003 に標準で搭載されています。

IIS6.0 にて提供される機能は、下記のとおりです。

-Web サーバー

使用するサービス

- ・ IISADMIN
- ・ W3SVC
- ・ HTTPFilter

状態サーバーモード使用時に必要なサービス

- ・ aspnet_state

-FTP サーバー

使用するサービス

- ・ IISADMIN
- ・ MSFTPSVC

14. (続き)

-SMTP サーバー
使用するサービス

- IISADMIN
- SMTPSVC

-POP3 サーバー
使用するサービス

- IISADMIN
- SMTPSVC
- POP3SVC

-NNTP サーバー
使用するサービス

- IISADMIN
- NNTPSVC

* IISADMIN サービスはすべての機能で使用されます。

14. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードなし)

タイプ	Service	Service	Service
表示名	IIS Admin Service	World Wide Web Publishing Service	HTTP SSL
サービス名	IISADMIN	W3SVC	HTTPFilter
常駐プロセス	inetinfo.exe	svchost.exe	lsass.exe
スタートアップ	自動	自動	手動
回復機能	プログラムを実行する	サービスを再起動する	何もしない
依存するサービス	<ul style="list-style-type: none"> Remote Procedure Call(RPC) Security Accounts Manager 	<ul style="list-style-type: none"> IIS Admin Service Remote Procedure Call(RPC) HTTP SSL 	<ul style="list-style-type: none"> HTTP IIS Admin Service
依存されるサービス	<ul style="list-style-type: none"> World Wide Web Publishing Service FTP Publishing Service Simple Mail Transfer Protocol(SMTP) Network News Transfer Protocol(NNTP) Microsoft POP3 Service HTTP SSL 	なし	<ul style="list-style-type: none"> World Wide Web Publishing Service

14. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードあり)

タイプ	Service	Service	Service	Service
表示名	IIS Admin Service	World Wide Web Publishing Service	HTTP SSL	ASP.NET State Service
サービス名	IISADMIN	W3SVC	HTTPFilter	aspnet_state
常駐プロセス	inetinfo.exe	svchost.exe	lsass.exe	aspnet_state.exe
スタートアップ	自動	自動	手動	自動
回復機能	プログラムを実行する	サービスを再起動する	何もしない	何もしない
依存するサービス	<ul style="list-style-type: none"> Remote Procedure Call(RPC) Security Accounts Manager 	<ul style="list-style-type: none"> IIS Admin Service Remote Procedure Call(RPC) HTTP SSL 	<ul style="list-style-type: none"> HTTP IIS Admin Service 	なし
依存されるサービス	<ul style="list-style-type: none"> World Wide Web Publishing Service FTP Publishing Service Simple Mail Transfer Protocol(SMTP) Network News Transfer Protocol(NNTP) Microsoft POP3 Service HTTP SSL 	なし	<ul style="list-style-type: none"> World Wide Web Publishing Service 	なし

14. (続き)

FTP サーバー使用時に必要なサービス

タイプ	Service	Service
表示名	IIS Admin Service	FTP Publishing Service
サービス名	IISADMIN	MSFTPSVC
常駐プロセス	inetinfo.exe	inetinfo.exe
スタートアップ	自動	自動
回復機能	プログラムを実行する	何もしない
依存するサービス	<ul style="list-style-type: none">Remote Procedure Call(RPC)Security Accounts Manager	<ul style="list-style-type: none">IIS Admin Service
依存されるサービス	<ul style="list-style-type: none">World Wide Web Publishing ServiceFTP Publishing ServiceSimple Mail Transfer Protocol(SMTP)Network News Transfer Protocol(NNTP)Microsoft POP3 ServiceHTTP SSL	なし

14. (続き)

SMTP サーバー使用時に必要なサービス

タイプ	Service	Service
表示名	IIS Admin Service	Simple Mail Transfer Protocol(SMTP)
サービス名	IISADMIN	SMTPSVC
常駐プロセス	inetinfo.exe	inetinfo.exe
スタートアップ	自動	自動
回復機能	プログラムを実行する	何もしない
依存するサービス	<ul style="list-style-type: none">• Remote Procedure Call(RPC)• Security Accounts Manager	<ul style="list-style-type: none">• Event Log• IIS Admin Service
依存されるサービス	<ul style="list-style-type: none">• World Wide Web Publishing Service• FTP Publishing Service• Simple Mail Transfer Protocol(SMTP)• Network News Transfer Protocol(NNTP)• Microsoft POP3 Service• HTTP SSL	なし

14. (続き)

POP3 サーバー使用時に必要なサービス

タイプ	Service	Service	Service
表示名	IIS Admin Service	Microsoft POP3 Service	Simple Mail Transfer Protocol(SMTP)
サービス名	IISADMIN	POP3SVC	SMTPSVC
常駐プロセス	inetinfo.exe	POP3Svc.exe	inetinfo.exe
スタートアップ	自動	自動	自動
回復機能	プログラムを実行する	サービスを再起動する	何もしない
依存するサービス	<ul style="list-style-type: none"> Remote Procedure Call(RPC) Security Accounts Manager 	<ul style="list-style-type: none"> Event Log IIS Admin Service 	<ul style="list-style-type: none"> Event Log IIS Admin Service
依存されるサービス	<ul style="list-style-type: none"> World Wide Web Publishing Service FTP Publishing Service Simple Mail Transfer Protocol(SMTP) Network News Transfer Protocol(NNTP) Microsoft POP3 Service HTTP SSL 	なし	なし

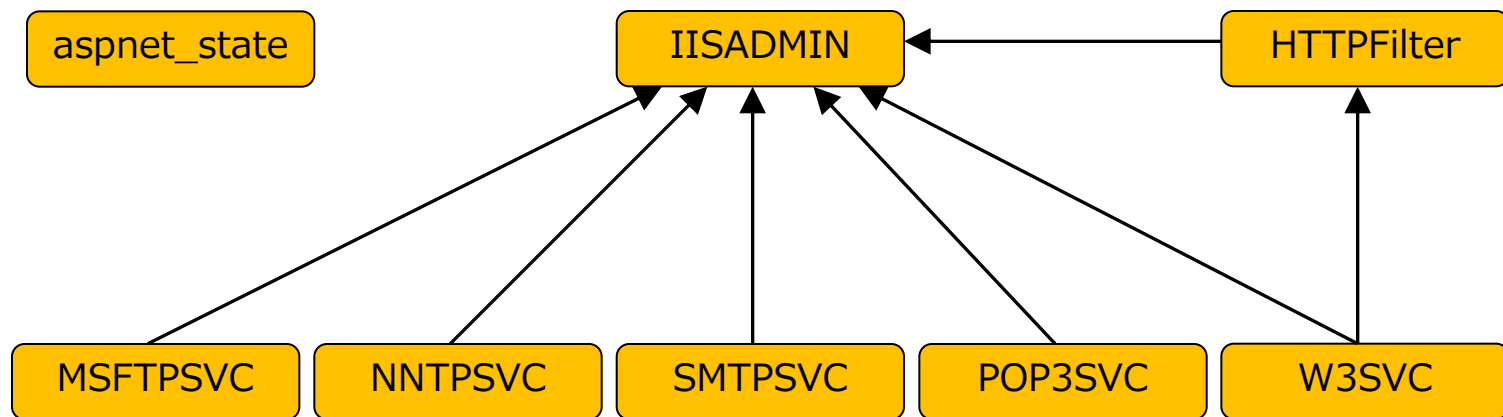
14. (続き)

NNTP サーバー使用時に必要なサービス

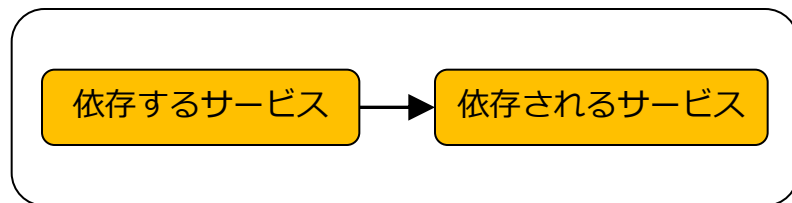
タイプ	Service	Service
表示名	IIS Admin Service	Network News Transfer Protocol(NNTP)
サービス名	IISADMIN	NNTPSVC
常駐プロセス	inetinfo.exe	inetinfo.exe
スタートアップ	自動	自動
回復機能	プログラムを実行する	何もしない
依存するサービス	<ul style="list-style-type: none">Remote Procedure Call(RPC)Security Accounts Manager	<ul style="list-style-type: none">Event LogIIS Admin Service
依存されるサービス	<ul style="list-style-type: none">World Wide Web Publishing ServiceFTP Publishing ServiceSimple Mail Transfer Protocol(SMTP)Network News Transfer Protocol(NNTP)Microsoft POP3 ServiceHTTP SSL	なし

14. (続き)

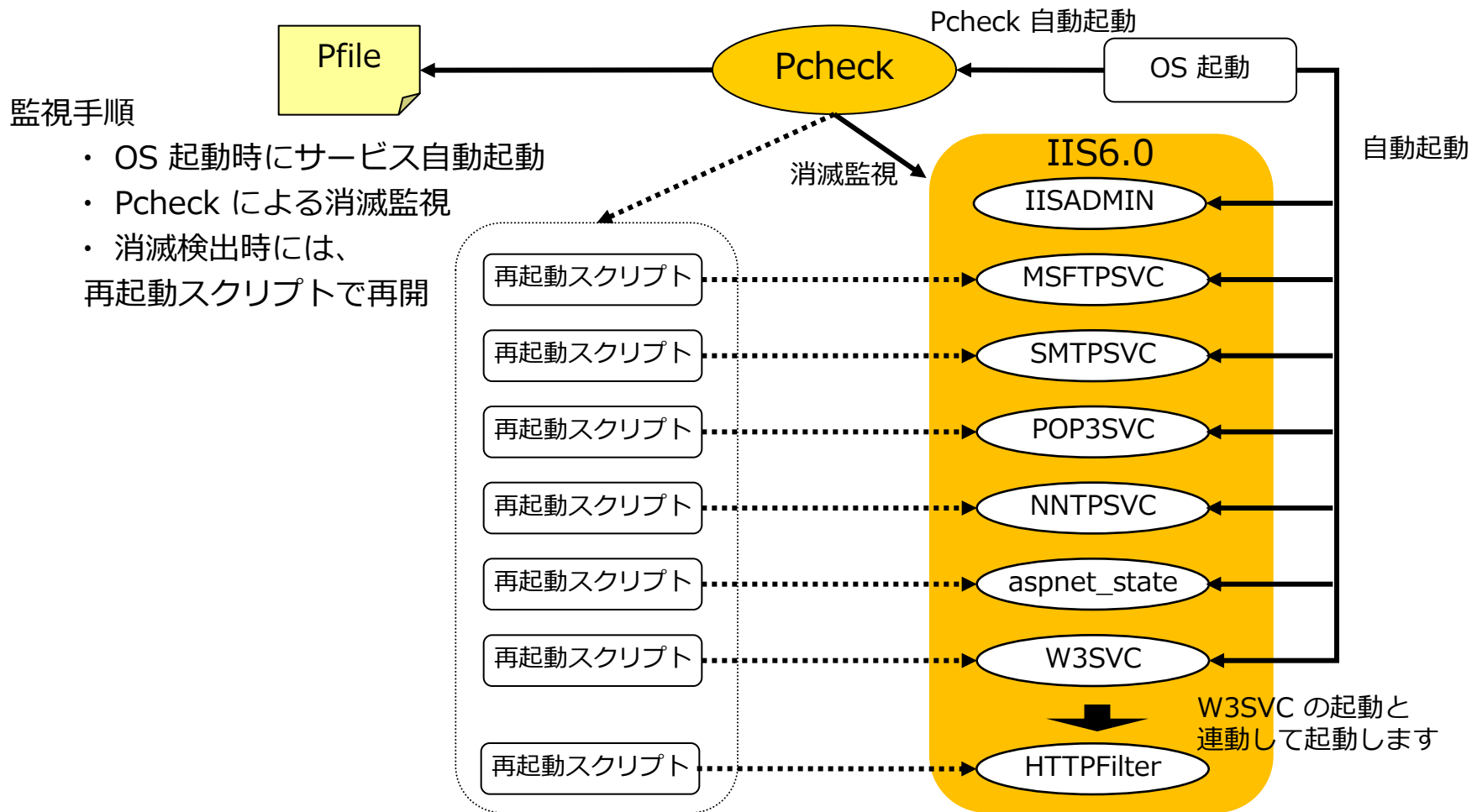
サービスの依存関係



例)



14. (続き)



14. (続き)

2. Pfile

Pfile <sample_iis6.0>

本 Pfile は IIS6.0 で 標準搭載されている機能をすべて監視する Pfile です。
本 Pfile は使用する機能や構築手順により、監視するサービスの名前や順番を
変更する必要があります。その場合、「4. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName           "C:¥Program Files¥HA¥ProcessSaver¥log¥IIS6.dmp"  
UpMessageReduceMode   disable  
  
##### PENT #####  
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;  
ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
  
## NNTP サーバー ##  
NNTPSVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS6_nttp.bat",86400,3,Continue,Service  
  
## POP3 サーバー ##  
POP3SVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS6_pop3.bat",86400,3,Continue,Service  
  
次ページへ続く
```

14. (続き)

続き

SMTP サーバー / POP3 サーバー

SMTPSVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS6_smtp.bat",86400,3,Continue,Service

FTP サーバー

MSFTPSVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS6_ftp.bat",86400,3,Continue,Service

Web サーバー

W3SVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS6_w3.bat",86400,3,Continue,Service

HTTPFilter,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS6_ssl.bat",86400,3,Continue,Service

全機能共通

IISADMIN,,86400,3,Continue,Service

Web サーバー (状態サーバーモード)

aspnet_state,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS6_aspnet_state.bat",86400,3,Continue,Service

14. (続き)

3. 再起動スクリプト

再起動スクリプト

- NNTPSVC <restart_IIS6_nntp.bat>

```
@echo off

rem NNTPSVC の再起動スクリプトです

:REPEAT

rem サービスの二重起動を防止するための処理を行います
sc query NNTPSVC | find "RUNNING"
if %ERRORLEVEL% == 0 exit

rem サービスの起動失敗を防止するための処理を行います
sc query IISADMIN | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理      }
sc start NNTPSVC }

exit
```

sc コマンドを使用してサービスの起動を行います。

14. (続き)

- POP3SVC <restart_IIS6_pop3.bat>

```
@echo off

rem POP3SVC の再起動スクリプトです

:REPEAT

rem サービスの二重起動を防止するための処理を行います
sc query POP3SVC | find "RUNNING"
if %ERRORLEVEL% == 0 exit

rem サービスの起動失敗を防止するための処理を行います
sc query IISADMIN | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理
sc start POP3SVC
exit
```

sc コマンドを使用してサービスの起動を行います。

14. (続き)

- SMTPSVC <restart_IIS6_smtp.bat>

```
@echo off

rem SMTPSVC の再起動スクリプトです

:REPEAT

rem サービスの二重起動を防止するための処理を行います
sc query SMTPSVC | find "RUNNING"
if %ERRORLEVEL% == 0 exit

rem サービスの起動失敗を防止するための処理を行います
sc query IISADMIN | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理
sc start SMTPSVC
exit
```

} sc コマンドを使用してサービスの起動を行います。

14. (続き)

- MSFTPSVC <restart_IIS6_ftp.bat>

```
@echo off
```

```
rem MSFTPSVC の再起動スクリプトです
```

```
:REPEAT
```

```
rem サービスの二重起動を防止するための処理を行います
```

```
sc query MSFTPSVC | find "RUNNING"
```

```
if %ERRORLEVEL% == 0 exit
```

```
rem サービスの起動失敗を防止するための処理を行います
```

```
sc query IISADMIN | find "STOP_PENDING"
```

```
if %ERRORLEVEL% == 0 goto REPEAT
```

```
rem 起動処理
```

```
sc start MSFTPSVC
```

```
}
```

sc コマンドを使用してサービスの起動を行います。

```
exit
```

14. (続き)

- W3SVC <restart_IIS6_w3.bat>

```
@echo off
```

```
rem W3SVC の再起動スクリプトです
```

```
:REPEAT
```

```
rem サービスの二重起動を防止するための処理を行います
```

```
sc query W3SVC | find "RUNNING"
```

```
if %ERRORLEVEL% == 0 exit
```

```
rem サービスの起動失敗を防止するための処理を行います
```

```
sc query HTTPFilter | find "STOP_PENDING"
```

```
if %ERRORLEVEL% == 0 goto REPEAT
```

```
sc query IISADMIN | find "STOP_PENDING"
```

```
if %ERRORLEVEL% == 0 goto REPEAT
```

```
rem 起動処理
```

```
sc start W3SVC
```

} sc コマンドを使用してサービスの起動を行います。

```
exit
```

14. (続き)

- HTTPFilter <restart_IIS6_ssl.bat>

```
@echo off

rem HTTPFilter の再起動スクリプトです

:REPEAT

rem サービスの二重起動を防止するための処理を行います
sc query HTTPFilter | find "RUNNING"
if %ERRORLEVEL% == 0 exit

rem サービスの起動失敗を防止するための処理を行います
sc query IISADMIN | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理
sc start HTTPFilter
exit
```

} sc コマンドを使用してサービスの起動を行います。

14. (続き)

- aspnet_state <restart_IIS6_aspnet_state.bat>

```
@echo off
```

```
rem aspnet_state の再起動スクリプトです
```

```
rem 起動処理
```

```
sc start aspnet_state
```

```
}
```

sc コマンドを使用してサービスの起動を行います。

```
exit
```

14. (続き)

4. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- ・ 監視しないサービスについては、Pfile から削除してください。
- ・ 機能の追加や削除を行った場合、サービス名が変わることがあります。
そのため、表示されているとおりに Pfile のサービス名を書き換えてください。
- ・ IISADMIN に依存するサービスは IISADMIN サービス停止時に、機能を追加した順番に停止します。そのため、機能を追加した逆順に Pfile を書き換えてください。

以降、Web サーバー（状態サーバーモードなし）、FTP サーバー、SMTP サーバー、POP3 サーバー、NNTP サーバーを監視する Pfile への書き換え手順を説明します。

14. (続き)

Pfile 書き換え手順

4-1. IIS6.0 の状態確認



4-2. Pfile の読み込み



4-3. 監視しないサービスの削除



4-4. 監視するサービスの並び替え



4-5. Pfile の保存

14. (続き)

4-1. IIS6.0 の状態確認

IIS6.0 に関連するインストール済みのサービス、サービス名、サービスのインストール順の確認を行います。

- (1) IIS6.0 の構成が終わった後に、コマンドプロンプトにて以下のコマンドを実行し、IIS6.0 に関連するすでにインストールされているサービスの一覧の情報を表示します。

(すべてのフォルダー配下で実行可能) > iisreset /status

出力例 *インストールしている構成によってサービスの表示件数が異なります。

```
C:¥Documents and Settings¥Administrator>iisreset /status
```

```
Microsoft POP3 Service (POP3SVC) の状態 : 実行中
```

```
World Wide Web Publishing Service ( W3SVC ) の状態 : 実行中
```

```
Simple Mail Transfer Protocol (SMTP) (SMTPSVC) の状態 : 実行中
```

```
Network News Transfer Protocol (NNTP) (NntpSvc) の状態 : 実行中
```

```
FTP Publishing Service (MSFtpsvc) の状態 : 実行中
```

```
HTTP SSL (HTTPFilter) の状態 : 実行中
```

14. (続き)

- (2) IIS6.0 でインストールされているサービスを確認し、そのサービス名 (下線部) と表示順を確認してください。

```
C:¥Documents and Settings¥Administrator>iisreset /status
```

Microsoft POP3 Service (POP3SVC) の状態 : 実行中

World Wide Web Publishing Service (_W3SVC) の状態 : 実行中

Simple Mail Transfer Protocol (SMTP) (SMTPSVC) の状態 : 実行中

Network News Transfer Protocol (NNTP) (NntpSvc) の状態 : 実行中

FTP Publishing Service (MSFtpsvc) の状態 : 実行中

HTTP SSL (HTTPFilter) の状態 : 実行中

14. (続き)

4-2. Pfile の読み込み

(1) 『Create Pfile』 で *Read* を押して

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_iis6.0> の読み込みを行ってください。

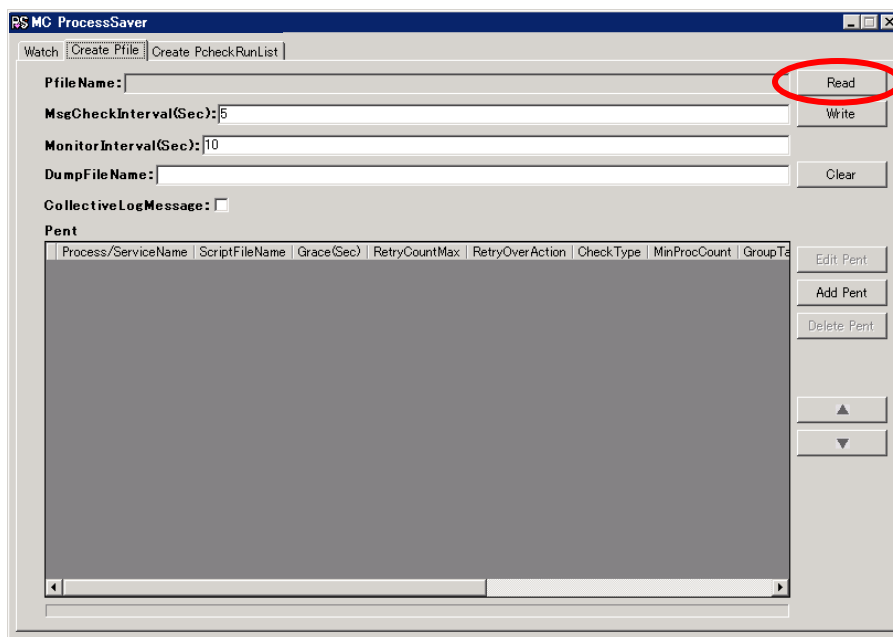


図1 『Create Pfile』

4-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から監視しないサービスの aspnet_state にカーソルをあわせて *Delete Pent* を押して削除してください。

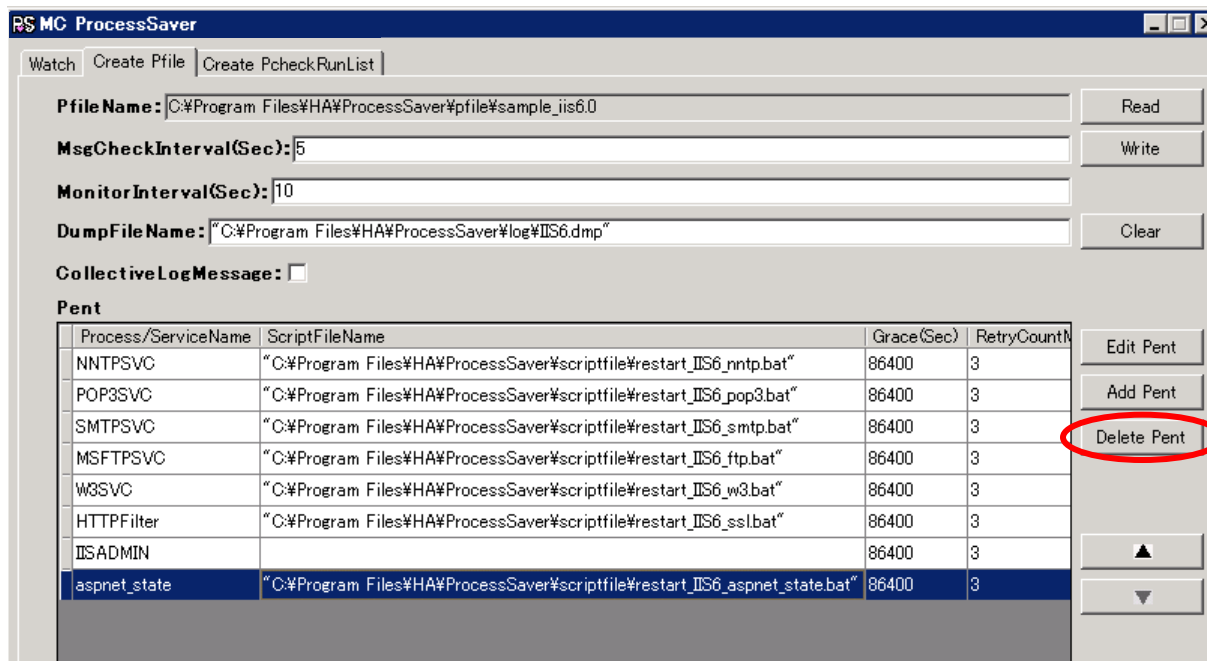


図2 『Create Pfile』

14. (続き)

4-4. 監視するサービスの並び替え

- (1) 「IIS の状態確認」で確認した表示順と通りに [Pent] を並び替えてください。並び替えの手順について「CLUSTERPRO MC ProcessSaver 2.10 for Windows ユーザーズガイド (コンソール編)」を参照してください。また、IISADMIN が一番下であることを確認してください。

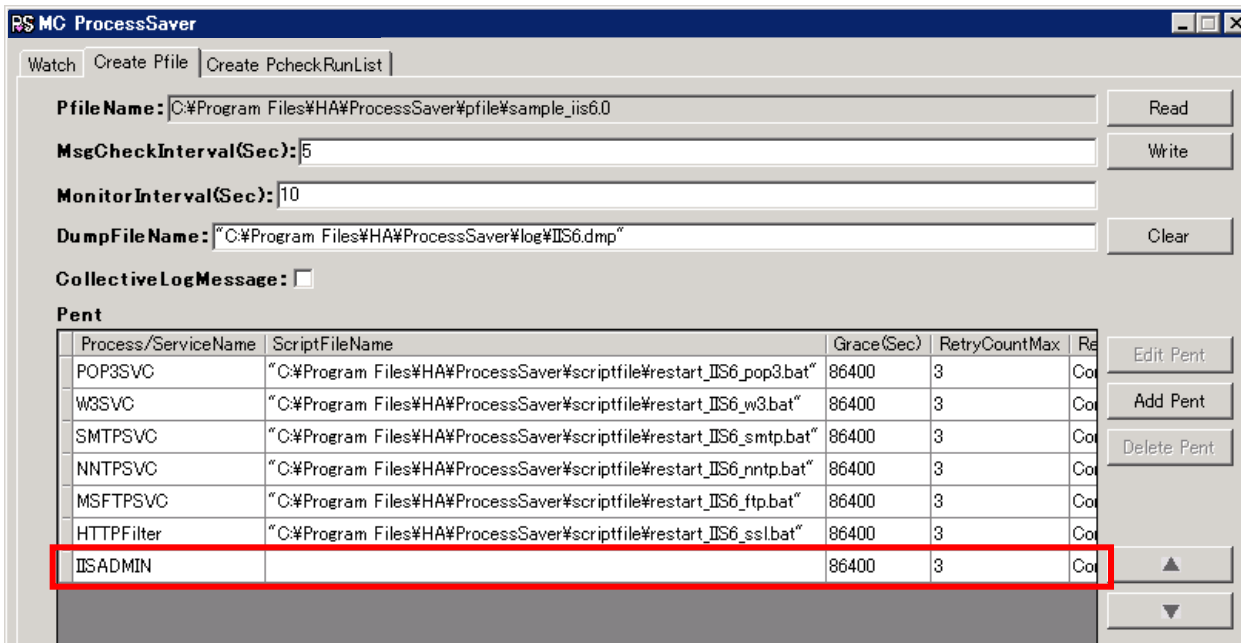


図3 『Create Pfile』

14. (続き)

- (2) 監視するすべての [Process/ServiceName] が「4-1. IIS6.0 の状態確認」で確認したサービス名と同じであるか確認してください。

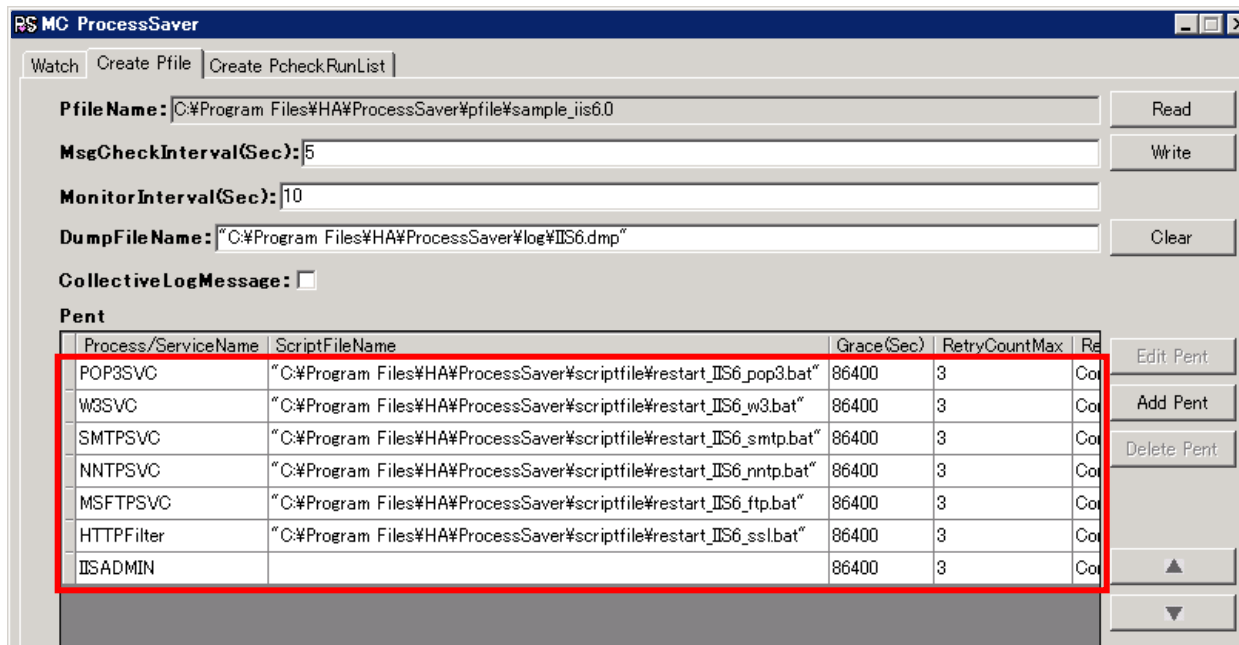


図4 『Create Pfile』

14. (続き)

4-5. Pfile の保存

(1) 『Create Pfile』 で *Write* を押して

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_iis6.0> を保存してください。

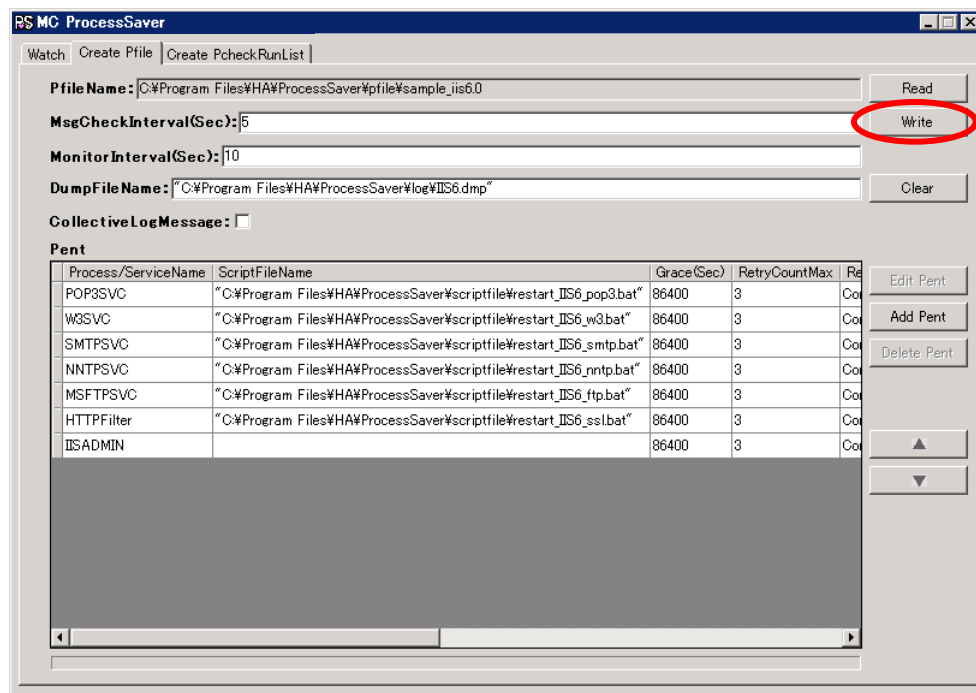


図5 『Create Pfile』

1. 概要

本事例は IIS7.0 を使用しています。
IIS7.0 は Windows Server 2008 に標準で搭載されています。

IIS7.0 にて提供される機能は、下記のとおりです。

-Web サーバー

使用するサービス

- ・ WAS
- ・ W3SVC
- ・ AppHostSvc

状態サーバーモード使用時に必要なサービス

- ・ aspnet_state

-リモート管理

使用するサービス

- ・ WMSvc

15. (続き)

-FTP サーバー

IIS7.0 では FTP サーバーが 2 バージョンから選択可能で、どちらか 1 つしかインストールできません。

それぞれの FTP サーバーが使用するサービスは以下のとおりです。

(FTP)

使用するサービス

- IISADMIN
- MSFTPSVC

(FTP 7)

使用するサービス

- ftpsvc

15. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードなし)

タイプ	Service	Service	Service
表示名	Windows Process Activation Service	World Wide Web Publishing Service	Application Host Helper Service
サービス名	WAS	W3SVC	AppHostSvc
常駐プロセス	svchost.exe	svchost.exe	svchost.exe
スタートアップ	手動	自動	自動
回復機能	プログラムを実行する	何もしない	サービスを再起動する
依存するサービス	・ Remote Procedure Call(RPC)	・ Windows Process Activation Service	なし
依存されるサービス	・ World Wide Web Publishing Service	なし	なし

15. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードあり)

タイプ	Service	Service	Service	Service
表示名	Windows Process Activation Service	World Wide Web Publishing Service	Application Host Helper Service	ASP.NET State Service
サービス名	WAS	W3SVC	AppHostSvc	aspnet_state
常駐プロセス	svchost.exe	svchost.exe	svchost.exe	aspnet_state.exe
スタートアップ	手動	自動	自動	自動
回復機能	プログラムを実行する	何もしない	サービスを再起動する	サービスを再起動する
依存するサービス	<ul style="list-style-type: none"> Remote Procedure Call(RPC) 	<ul style="list-style-type: none"> Windows Process Activation Service 	なし	なし
依存されるサービス	<ul style="list-style-type: none"> World Wide Web Publishing Service 	なし	なし	なし

15. (続き)

リモート管理使用時に必要なサービス

タイプ	Service
表示名	Web Management Service
サービス名	WMSvc
常駐プロセス	WMSvc.exe
スタートアップ	自動
回復機能	サービスを再起動する
依存するサービス	・ HTTP
依存されるサービス	なし

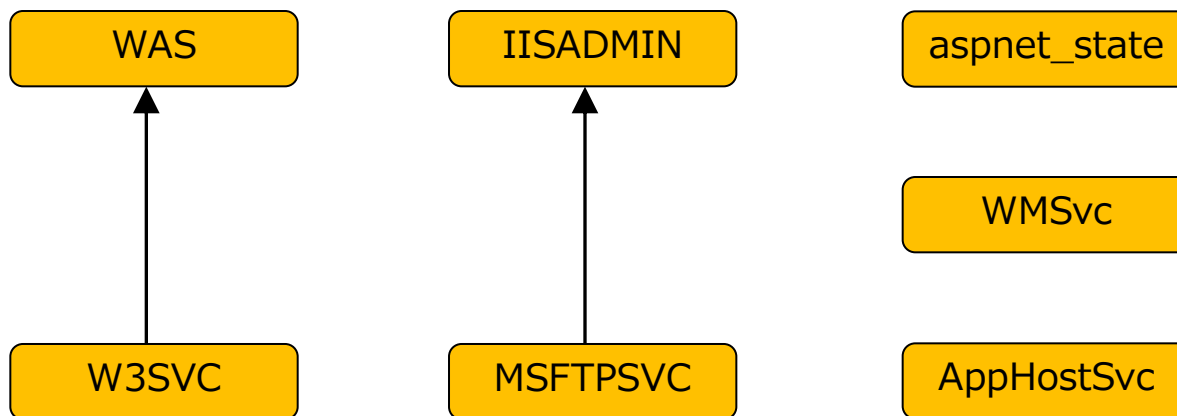
15. (続き)

FTP サーバー (FTP) 使用時に必要なサービス

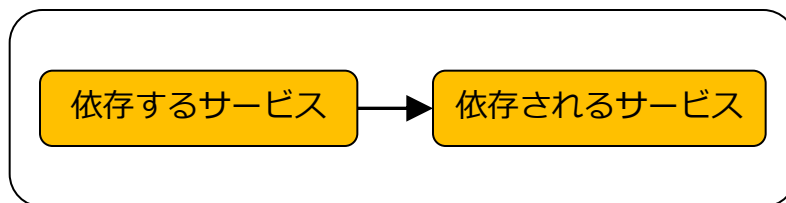
タイプ	Service	Service
表示名	IIS Admin Service	FTP Publishing Service
サービス名	IISADMIN	MSFTPSVC
常駐プロセス	inetinfo.exe	inetinfo.exe
スタートアップ	自動	自動
回復機能	プログラムを実行する	何もしない
依存するサービス	<ul style="list-style-type: none">Remote Procedure Call(RPC)Security Accounts Manager	<ul style="list-style-type: none">IIS Admin Service
依存されるサービス	<ul style="list-style-type: none">HTTPFTP Publishing Service	なし

15. (続き)

FTP サーバー (FTP) 使用
時 サービスの依存関係

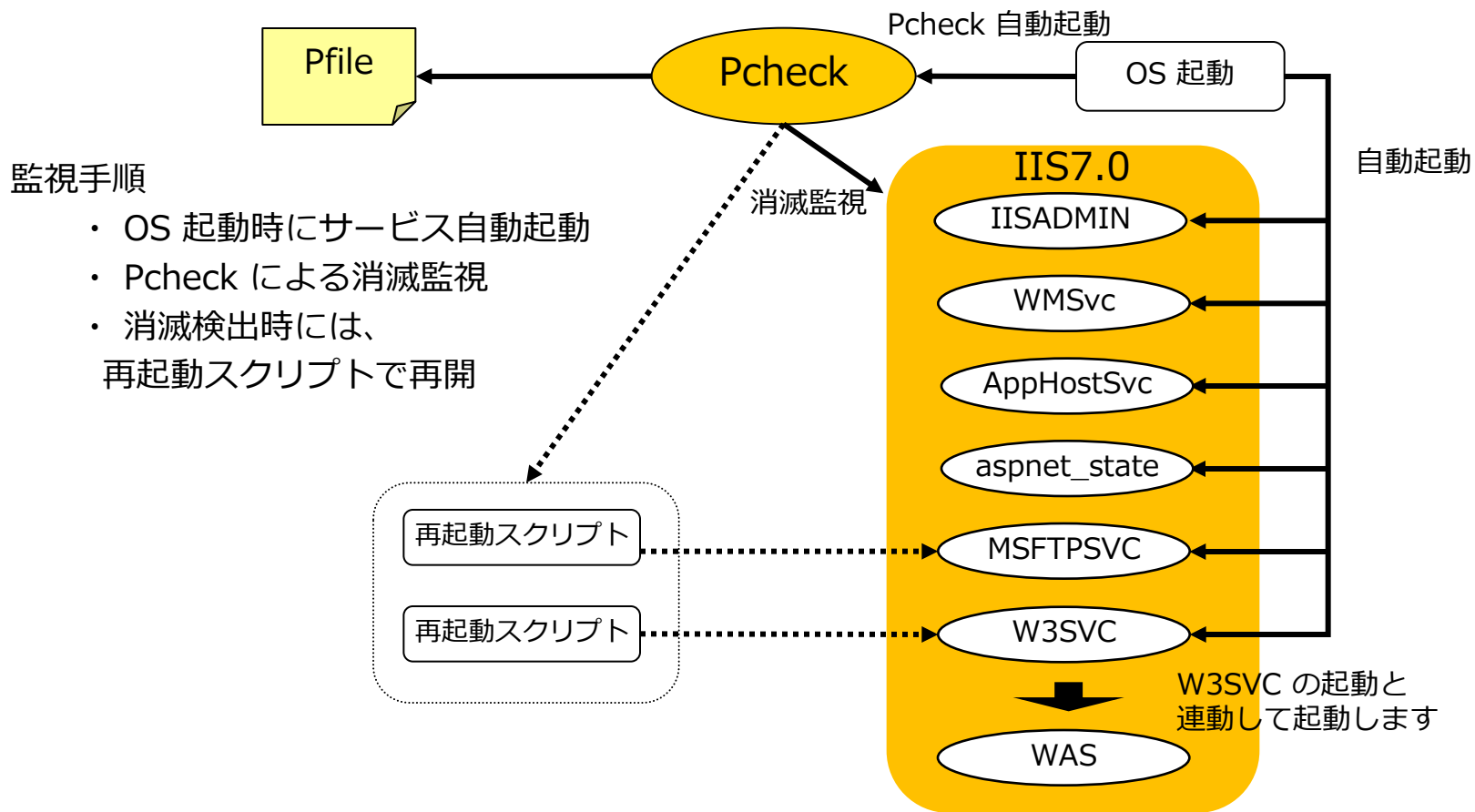


例)



15. (続き)

FTP サーバー (FTP) 使用時



15. (続き)

FTP サーバー (FTP 7) 使用時に必要なサービス

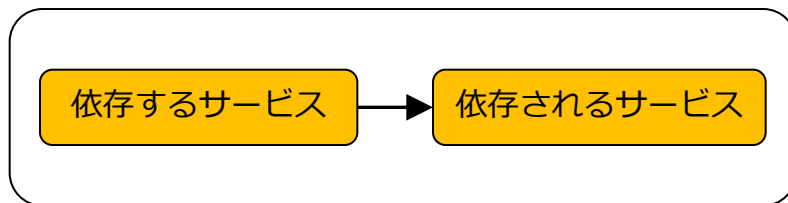
タイプ	Service
表示名	Microsoft FTP Service
サービス名	ftpsvc
常駐プロセス	svchost.exe
スタートアップ	自動
回復機能	サービスを再起動する
依存するサービス	・ Remote Procedure Call(RPC)
依存されるサービス	なし

15. (続き)

FTP サーバー (FTP 7) 使用時
サービスの依存関係

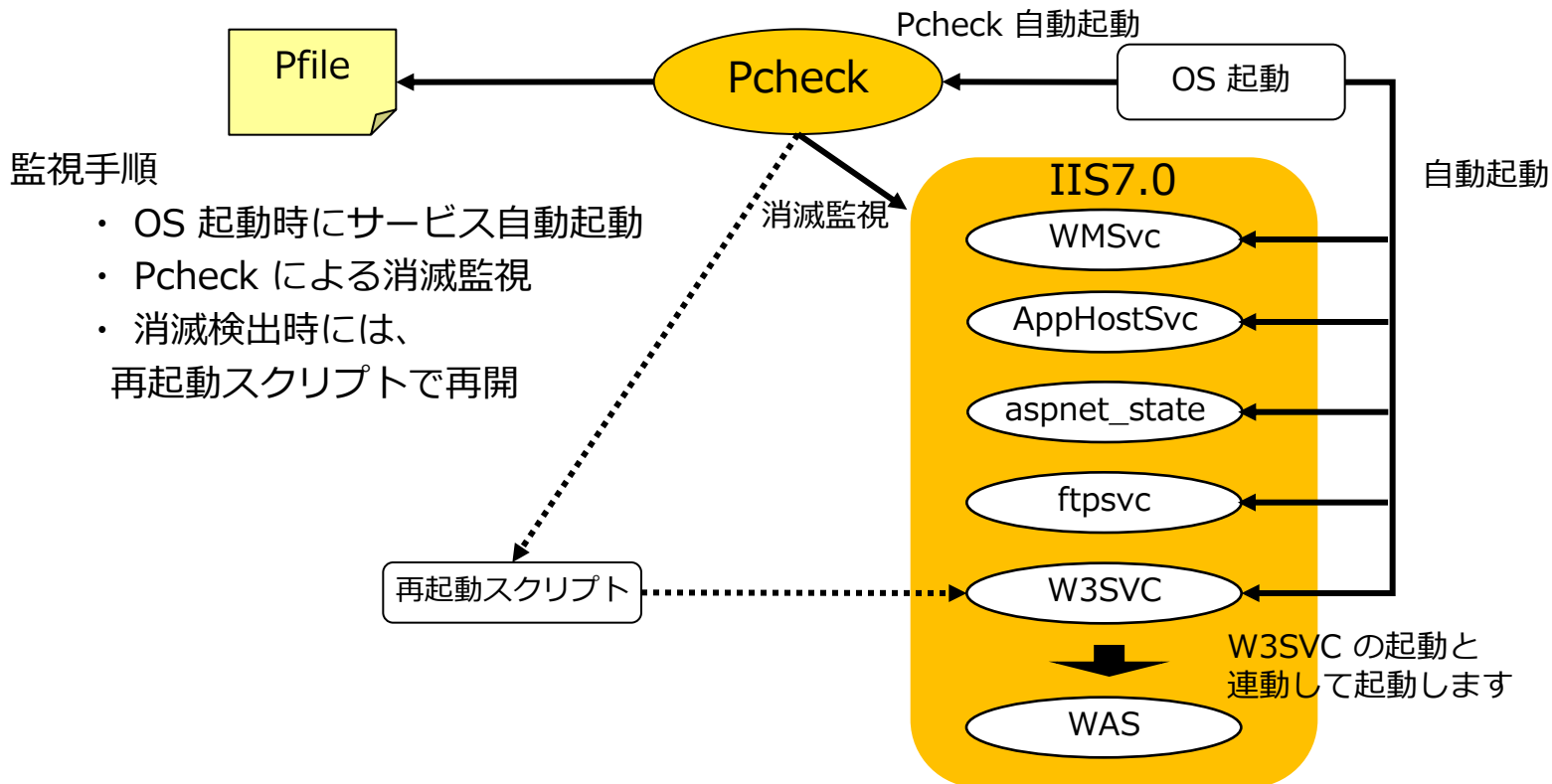


例)



15. (続き)

FTP サーバー (FTP 7) 使用時



15. (続き)

2. Pfile

Pfile <sample_iis7.0>

本 Pfile は IIS7.0 で 以下のサーバーまたは、機能を監視する Pfile です。

- Web サーバー (状態サーバーモードあり)
- リモート管理
- FTP サーバー (FTP)

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

本 Pfile は FTP サーバー (FTP 7) の監視はコメントアウトしていますので FTP サーバー (FTP 7) を監視するためには Pfile を変更する必要があります。その場合、「4. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥IIS7.dmp"  
UpMessageReduceMode   disable
```

[次ページへ続く](#)

15. (続き)

続き

```
##### PENT #####
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;
ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch

## Web サーバー ##
AppHostSvc,,86400,3,Continue,Service
W3SVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS7_w3.bat",86400,3,Continue,Service
WAS,,86400,3,Continue,Service

## Web サーバー ( 状態サーバーモード ) ##
aspnet_state,,86400,3,Continue,Service

## リモート管理 ##
WMSvc,,86400,3,Continue,Service

## FTP サーバー ##
MSFTPSVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS7_ftp.bat",86400,3,Continue,Service
IISADMIN,,86400,3,Continue,Service

### FTP 7 ###
# FTP を使用する設定にしているため FTP 7 はコメントアウトしています
#ftpsvc,,86400,3,Continue,Service
```

3. 再起動スクリプト

再起動スクリプト

- W3SVC <restart_IIS7_w3.bat>

```
@echo off

rem W3SVC の再起動スクリプトです

:REPEAT

rem サービスの二重起動を防止するための処理を行います
sc query W3SVC | find "RUNNING"
if %ERRORLEVEL% == 0 exit

rem サービスの起動失敗を防止するための処理を行います
sc query WAS | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理      }
sc start W3SVC    }

exit
```

sc コマンドを使用してサービスの起動を行います。

15. (続き)

- MSFTPSVC <restart_IIS7_ftp.bat>

```
@echo off

rem MSFTPSVC の再起動スクリプトです

:REPEAT

rem サービスの二重起動を防止するための処理を行います
sc query MSFTPSVC | find "RUNNING"
if %ERRORLEVEL% == 0 exit

rem サービスの起動失敗を防止するための処理を行います
sc query IISADMIN | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理
sc start MSFTPSVC
exit
```

sc コマンドを使用してサービスの起動を行います。

4. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- ・ 監視しないサービスについては、Pfile から削除してください。
- ・ FTP サーバー（FTP 7）の監視については、コメントアウトしています。
FTP サーバー（FTP 7）を使用する場合は、Pfile を書き換えて、コメントインしてください。

以降、Web サーバー（状態サーバーモードあり）、リモート管理、FTP サーバー（FTP 7）を使用する場合の Pfile への書き換え手順を説明します。

15. (続き)

Pfile 書き換え手順

4-1. Pfile の直接編集



4-2. Pfile の読み込み



4-3. 監視しないサービスの削除



4-4. Pfile の保存

15. (続き)

4-1. Pfile の直接編集

(1) テキストエディターなどで

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_iis7.0> を開き、
FTP 7 ### の下の # を削除し、コメントインしてください。

```
      .  
      .  
      .  
## FTP サーバー ##  
### FTP ###  
MSFTPSVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS7_ftp.bat",86400,3,Continue,Service  
IISADMIN,,86400,3,Continue,Service  
  
### FTP 7 ###  
# FTP を使用する設定にしているため FTP 7 はコメントアウトしています  
# ftpsvc,,86400,3,Continue,Service
```

を削除してコメントインしてください。

15. (続き)

4-2. Pfile の読み込み

(1) 『Create Pfile』 で *Read* を押して

Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_iis7.0> の読み込みを行ってください。

下記画面にならなかった場合、または Pfile の読み込み時にエラーか警告が出力された場合は、Page 18 の手順により Pfile を再取得し、もう一度「4-1. Pfile の直接編集」からやり直してください。

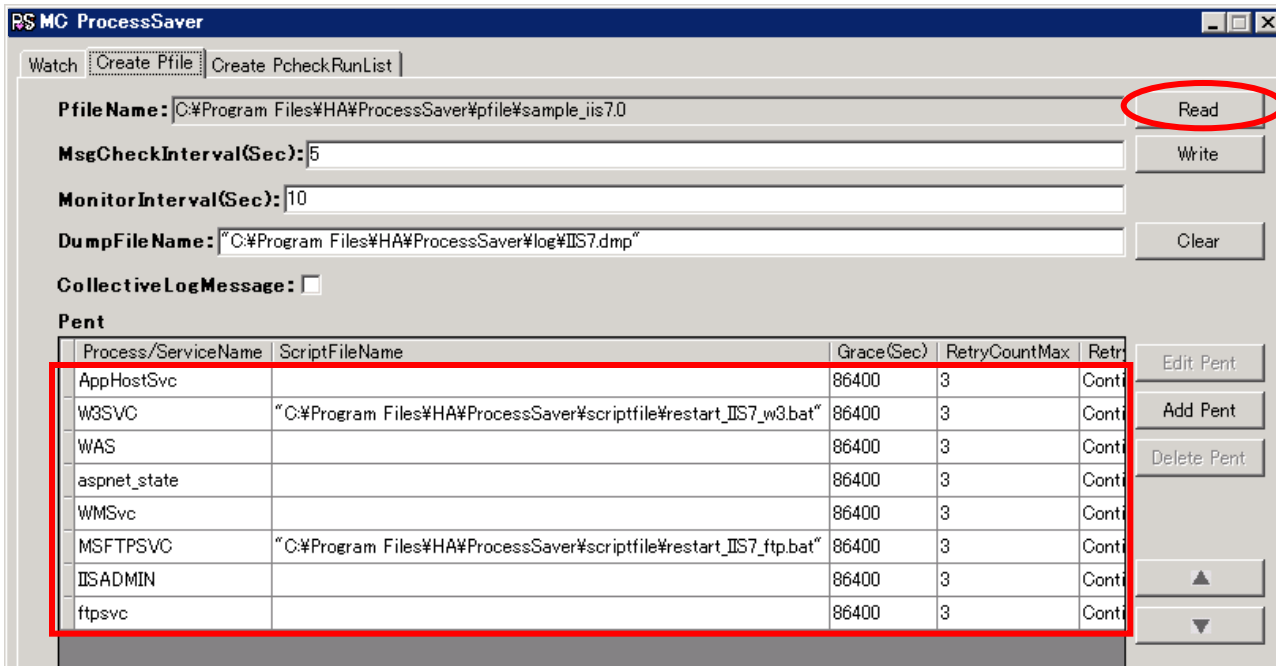


図6 『Create Pfile』

15. (続き)

4-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から FTP サーバー (FTP) のサービスの MSFTPSVC と IISADMIN を *Delete Pent* を押して削除してください。

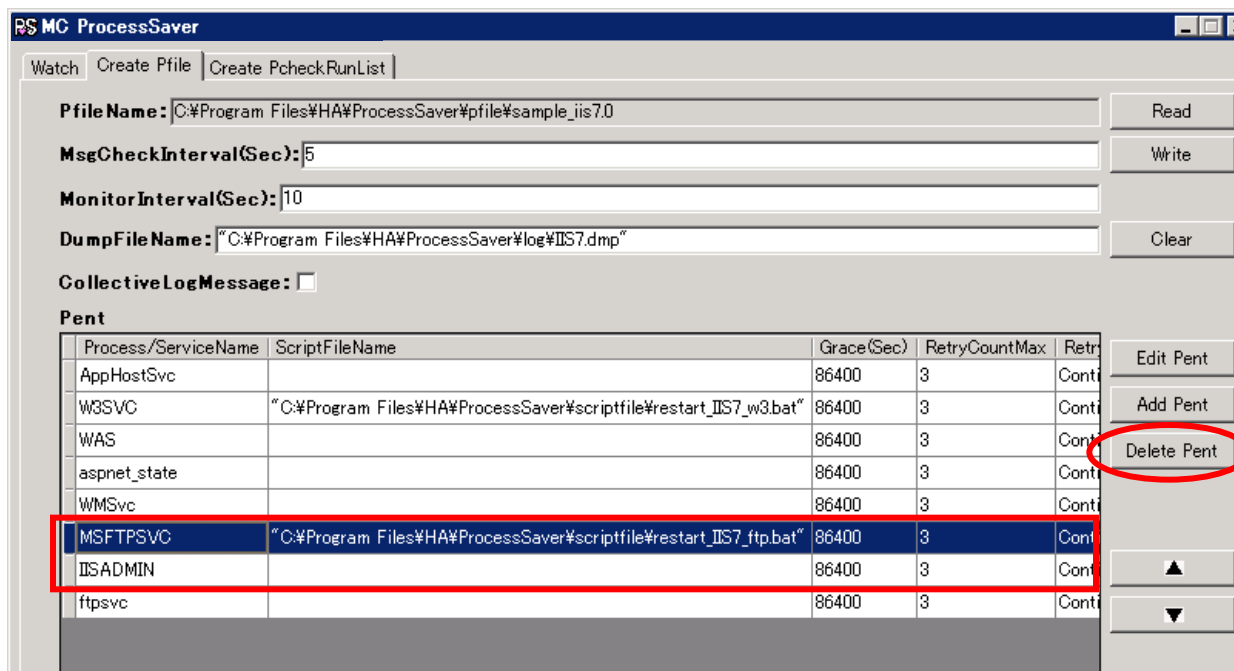


図7 『Create Pfile』

15. (続き)

4-4. Pfile の保存

- (1) 『Create Pfile』 で *Write* を押して
Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_iis7.0> を保存してください。

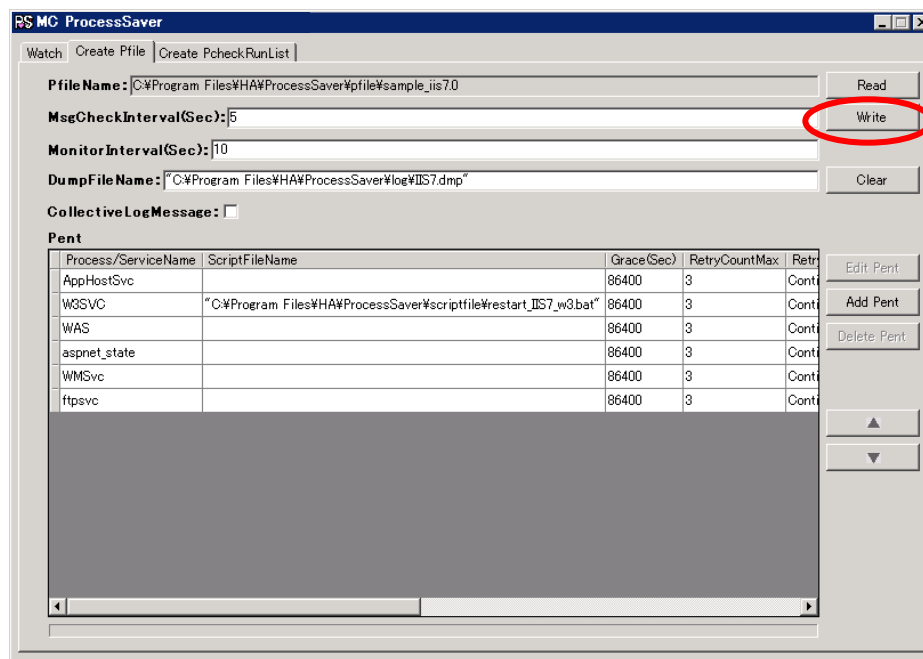


図8 『Create Pfile』

16. IIS7.5 の監視事例

1. 概要

本事例は IIS7.5 を使用しています。

IIS7.5 は Windows Server 2008 R2 に標準で搭載されています。

IIS7.5 にて提供される機能は、下記のとおりです。

-Web サーバー

使用するサービス

- ・ WAS
- ・ W3SVC
- ・ AppHostSvc

状態サーバーモード使用時に必要なサービス

- ・ aspnet_state

-リモート管理

使用するサービス

- ・ WMSVC

-FTP サーバー

使用するサービス

- ・ IISADMIN
- ・ ftpsvc

16. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードなし)

タイプ	Service	Service	Service
表示名	Windows Process Activation Service	World Wide Web Publishing Service	Application Host Helper Service
サービス名	WAS	W3SVC	AppHostSvc
常駐プロセス	svchost.exe	svchost.exe	svchost.exe
スタートアップ	手動	自動	自動
回復機能	プログラムを実行する	何もしない	サービスを再起動する
依存するサービス	・ Remote Procedure Call(RPC)	・ HTTP ・ Windows Process Activation Service	なし
依存されるサービス	・ World Wide Web Publishing Service	なし	なし

16. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードあり)

タイプ	Service	Service	Service	Service
表示名	Windows Process Activation Service	World Wide Web Publishing Service	Application Host Helper Service	ASP.NET State Service
サービス名	WAS	W3SVC	AppHostSvc	aspnet_state
常駐プロセス	svchost.exe	svchost.exe	svchost.exe	aspnet_state.exe
スタートアップ	手動	自動	自動	自動
回復機能	プログラムを実行する	何もしない	サービスを再起動する	サービスを再起動する
依存するサービス	・ Remote Procedure Call(RPC)	・ HTTP ・ Windows Process Activation Service	なし	なし
依存されるサービス	・ World Wide Web Publishing Service	なし	なし	なし

16. (続き)

リモート管理使用時に必要なサービス

タイプ	Service
表示名	Web Management Service
サービス名	WMSVC
常駐プロセス	WMSvc.exe
スタートアップ	手動
回復機能	サービスを再起動する
依存するサービス	・ HTTP
依存されるサービス	なし

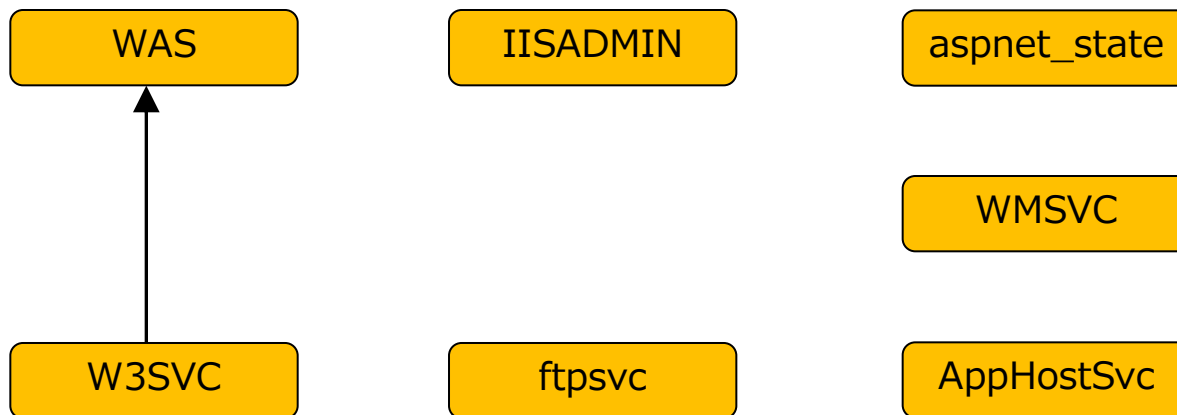
16. (続き)

FTP サーバー (FTP) 使用時に必要なサービス

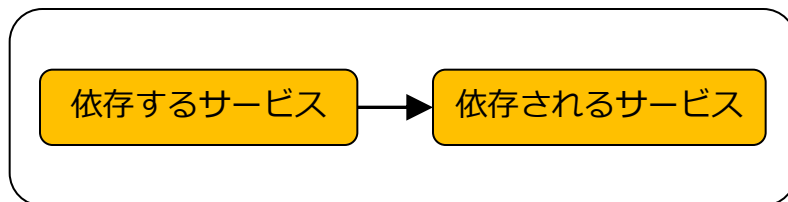
タイプ	Service	Service
表示名	IIS Admin Service	Microsoft FTP Service
サービス名	IISADMIN	ftpsvc
常駐プロセス	inetinfo.exe	svchost.exe
スタートアップ	自動	自動
回復機能	プログラムを実行する	サービスを再起動する
依存するサービス	<ul style="list-style-type: none">Remote Procedure Call(RPC)Security Accounts Manager	<ul style="list-style-type: none">Remote Procedure Call(RPC)
依存されるサービス	なし	なし

16. (続き)

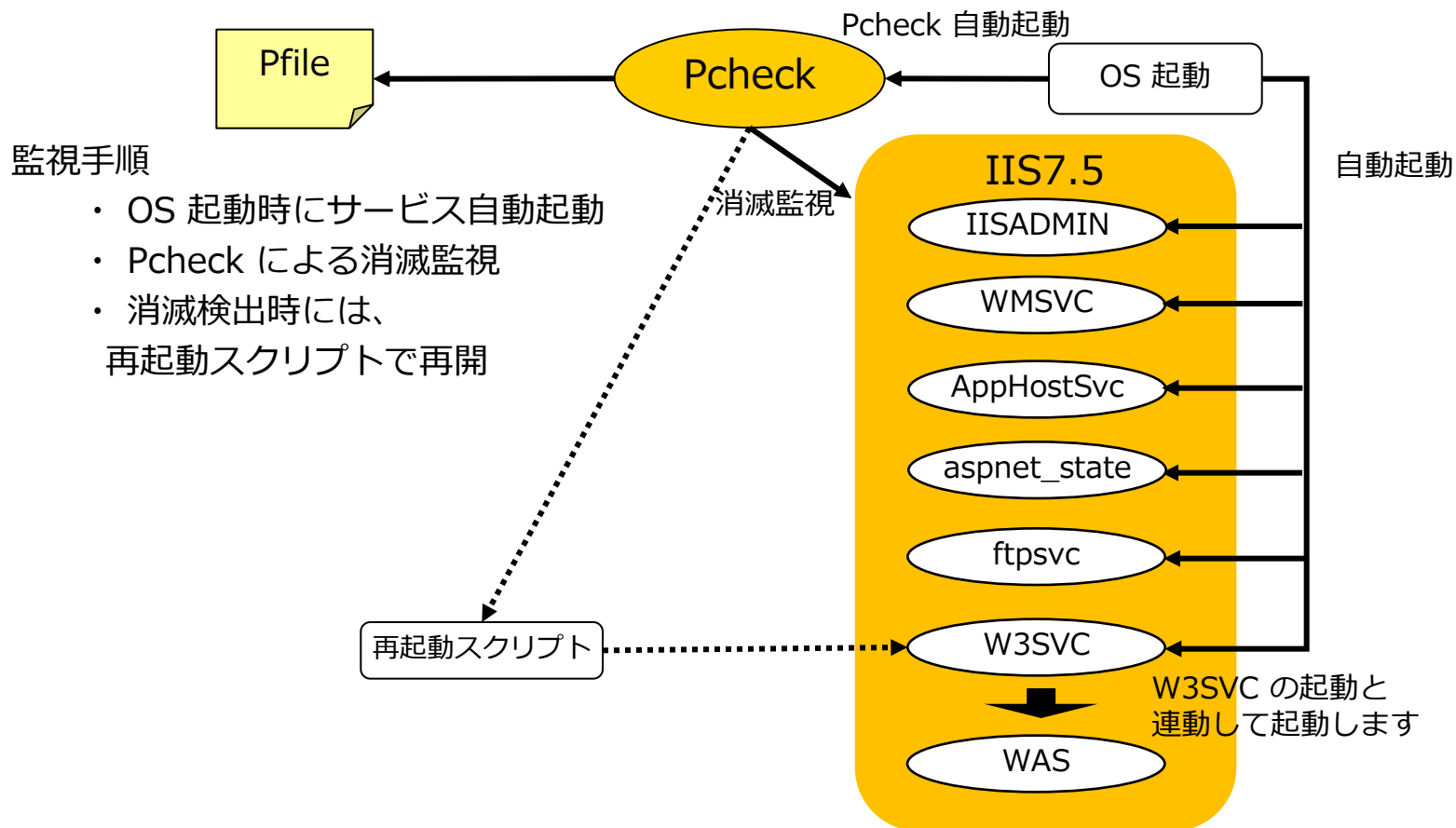
サービスの依存関係



例)



16. (続き)



16. (続き)

2. Pfile

Pfile <sample_iis7.5>

本 Pfile は IIS7.5 で 以下のサーバーまたは、機能を監視する Pfile です。

- Web サーバー (状態サーバーモードあり)
- リモート管理
- FTP サーバー

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。
その場合、「4. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥IIS75.dmp"  
UpMessageReduceMode   disable
```

次ページへ続く

16. (続き)

続き

```
##### PENT #####
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;
ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch

## Web サーバー ##
AppHostSvc,,86400,3,Continue,Service
W3SVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS75_w3.bat",86400,3,Continue,Service
WAS,,86400,3,Continue,Service

## Web サーバー ( 状態サーバーモード ) ##
aspnet_state,,86400,3,Continue,Service

## リモート管理 ##
WMSVC,,86400,3,Continue,Service

## FTP サーバー ##
ftpsvc,,86400,3,Continue,Service
IISADMIN,,86400,3,Continue,Service
```

3. 再起動スクリプト

再起動スクリプト

- W3SVC <restart_IIS75_w3.bat>

```
@echo off

rem W3SVC の再起動スクリプトです

:REPEAT

rem サービスの起動失敗を防止するための処理を行います
sc query WAS | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理
sc start W3SVC

exit
```

sc コマンドを使用してサービスの起動を行います。

16. (続き)

4. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- ・ 監視しないサービスについては、Pfile から削除してください。

以降、状態サーバーモードなしにする、または、リモート管理や FTP サーバー を監視しない場合の Pfile への書き換え手順を説明します。

16. (続き)

Pfile 書き換え手順

4-1. Pfile の直接編集



4-2. Pfile の読み込み



4-3. 監視しないサービスの削除



4-4. Pfile の保存

16. (続き)

4-1. Pfile の直接編集

(1) テキストエディターなどで

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_iis7.5> を開き、監視しないサービスを削除するか、コメントアウトしてください。

```

.
.
.
## Web サーバー ##
AppHostSvc,,86400,3,Continue,Service
W3SVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS75_w3.bat",86400,3,Continue,Service
WAS,,86400,3,Continue,Service

## Web サーバー ( 状態サーバーモード ) ##
aspnet_state,,86400,3,Continue,Service

## リモート管理 ##
WMSVC,,86400,3,Continue,Service

## FTP サーバー ##
ftpsvc,,86400,3,Continue,Service
IISADMIN,,86400,3,Continue,Service

```

監視しないサービスの記載を削除してください。

を記載してコメントアウトしてください。

16. (続き)

4-2. Pfile の読み込み

- (1) 『Create Pfile』 で *Read* を押して
Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_iis7.5> の
読み込みを行ってください。

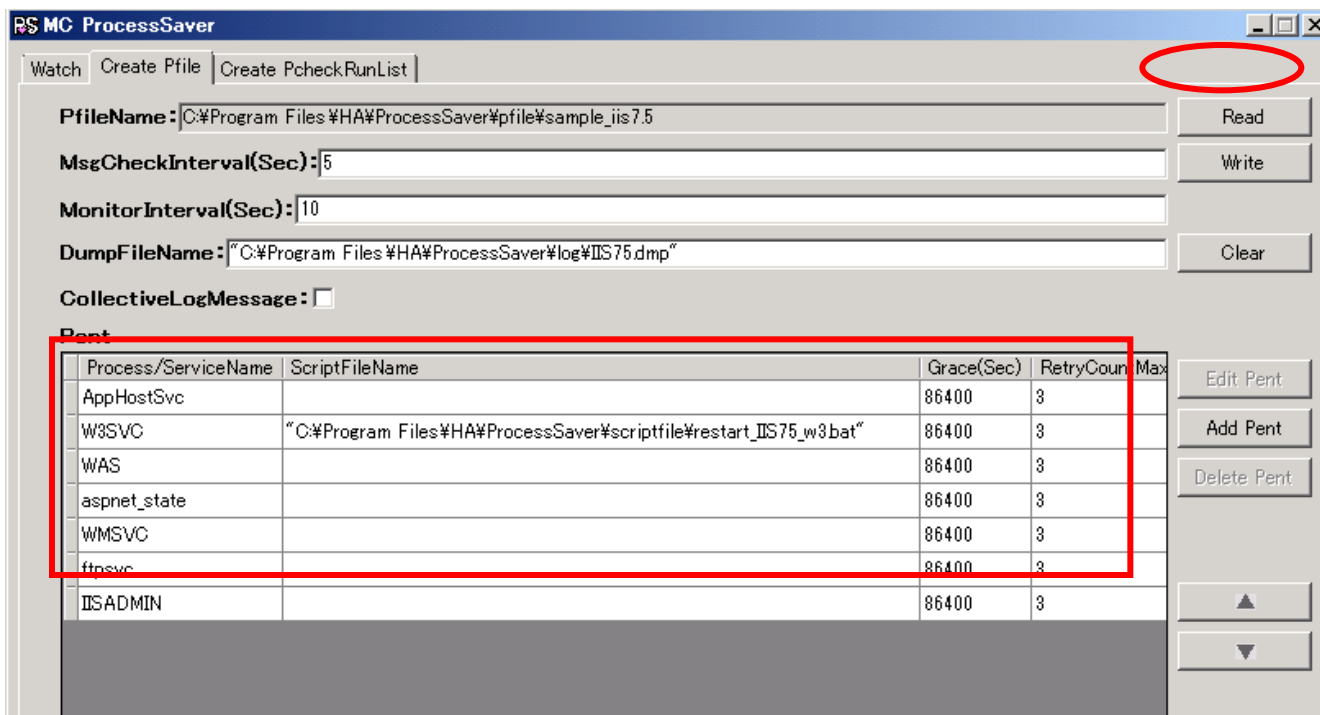


図9 『Create Pfile』

16. (続き)

4-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から、監視しないサービスを選択し、*Delete Pent* を押して削除してください。

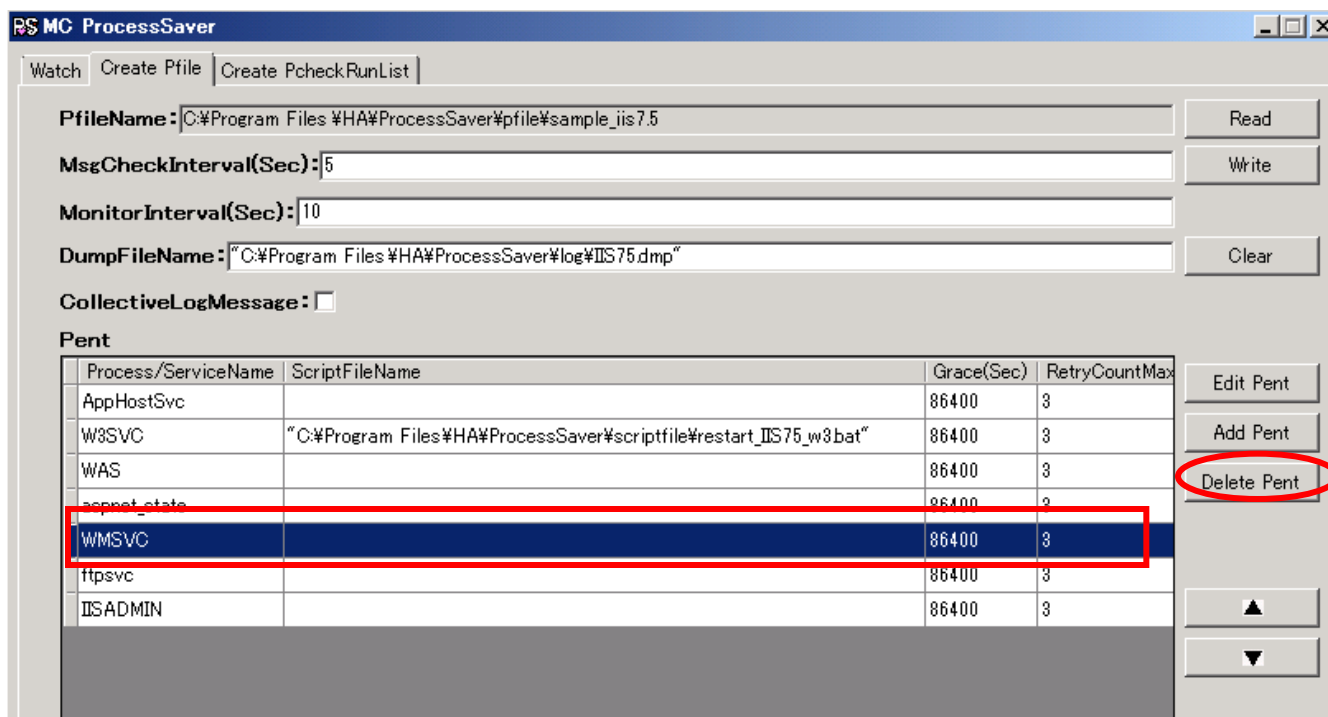


図10 『Create Pfile』

16. (続き)

4-4. Pfile の保存

- (1) 『Create Pfile』 で *Write* を押して
Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_iis7.5> を保存してください。

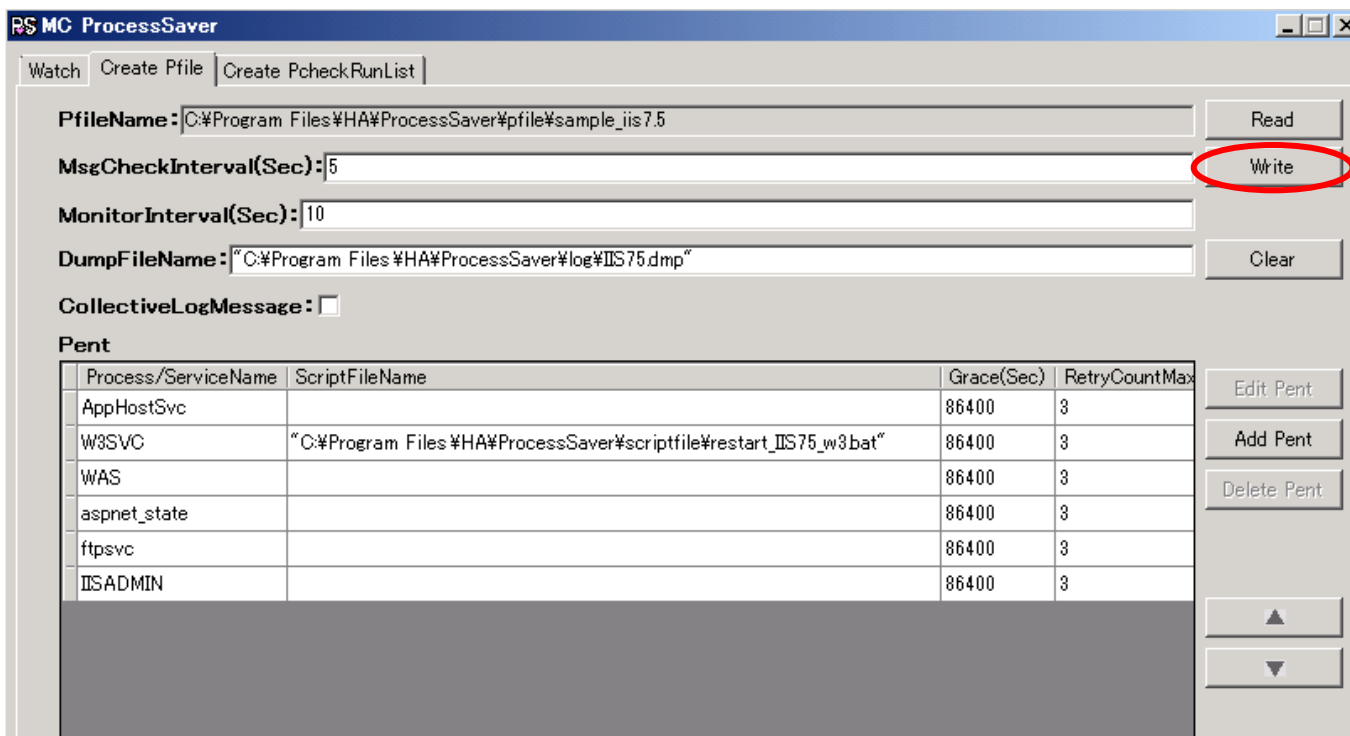


図11 『Create Pfile』

17. IIS8.0 の監視事例

1. 概要

本事例は IIS8.0 を使用しています。

IIS8.0 は Windows Server 2012 に標準で搭載されています。

IIS8.0 にて提供される機能は、下記のとおりです。

※ IIS8.0 にて提供される機能は、IIS7.5 にて提供される機能と同じものです。

ただし、IIS7.5 と IIS8.0 とではサービス名が異なり、Pfile や再起動スクリプトに指定するサービス名を変更する必要がありますので、バージョンにあったテンプレートを使用してください。

-Web サーバー

使用するサービス

- WAS
- W3SVC
- AppHostSvc

状態サーバーモード使用時に必要なサービス

- aspnet_state

-リモート管理

使用するサービス

- WMSvc

-FTP サーバー

使用するサービス

- IISADMIN
- FTPSVC

17. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードなし)

タイプ	Service	Service	Service
表示名	Windows Process Activation Service	World Wide Web Publishing Service	Application Host Helper Service
サービス名	WAS	W3SVC	AppHostSvc
常駐プロセス	svchost.exe	svchost.exe	svchost.exe
スタートアップ	手動	自動	自動
回復機能	プログラムを実行する	何もしない	サービスを再起動する
依存するサービス	・ Remote Procedure Call(RPC)	・ HTTP ・ Windows Process Activation Service	なし
依存されるサービス	・ World Wide Web Publishing Service	なし	なし

17. (続き)

Web サーバー使用時に必要なサービス (状態サーバーモードあり)

タイプ	Service	Service	Service	Service
表示名	Windows Process Activation Service	World Wide Web Publishing Service	Application Host Helper Service	ASP.NET State Service
サービス名	WAS	W3SVC	AppHostSvc	aspnet_state
常駐プロセス	svchost.exe	svchost.exe	svchost.exe	aspnet_state.exe
スタートアップ	手動	自動	自動	手動
回復機能	プログラムを実行する	何もしない	サービスを再起動する	サービスを再起動する
依存するサービス	<ul style="list-style-type: none"> Remote Procedure Call(RPC) 	<ul style="list-style-type: none"> HTTP Windows Process Activation Service 	なし	なし
依存されるサービス	<ul style="list-style-type: none"> World Wide Web Publishing Service 	なし	なし	なし

17. (続き)

リモート管理使用時に必要なサービス

タイプ	Service
表示名	Web Management Service
サービス名	WMSvc
常駐プロセス	WMSvc.exe
スタートアップ	手動
回復機能	サービスを再起動する
依存するサービス	・ HTTP
依存されるサービス	なし

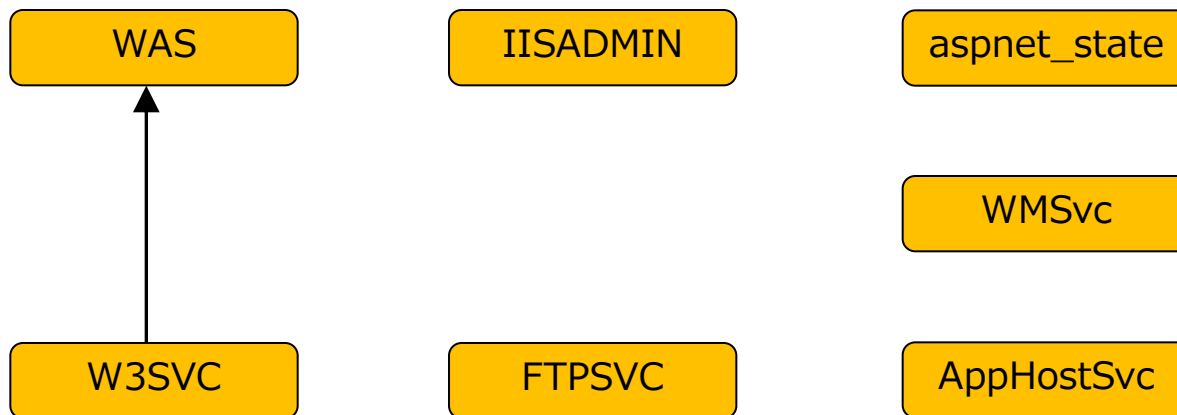
17. (続き)

FTP サーバー (FTP) 使用時に必要なサービス

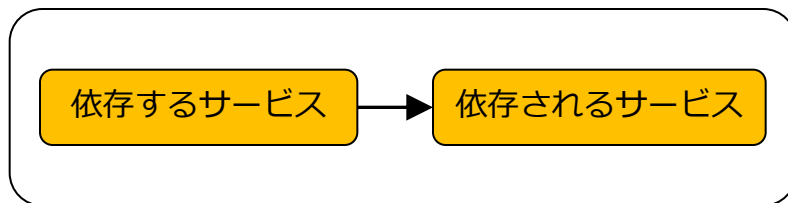
タイプ	Service	Service
表示名	IIS Admin Service	Microsoft FTP Service
サービス名	IISADMIN	FTPSVC
常駐プロセス	inetinfo.exe	svchost.exe
スタートアップ	自動	自動
回復機能	プログラムを実行する	サービスを再起動する
依存するサービス	<ul style="list-style-type: none">Remote Procedure Call(RPC)Security Accounts Manager	<ul style="list-style-type: none">Remote Procedure Call(RPC)
依存されるサービス	なし	なし

17. (続き)

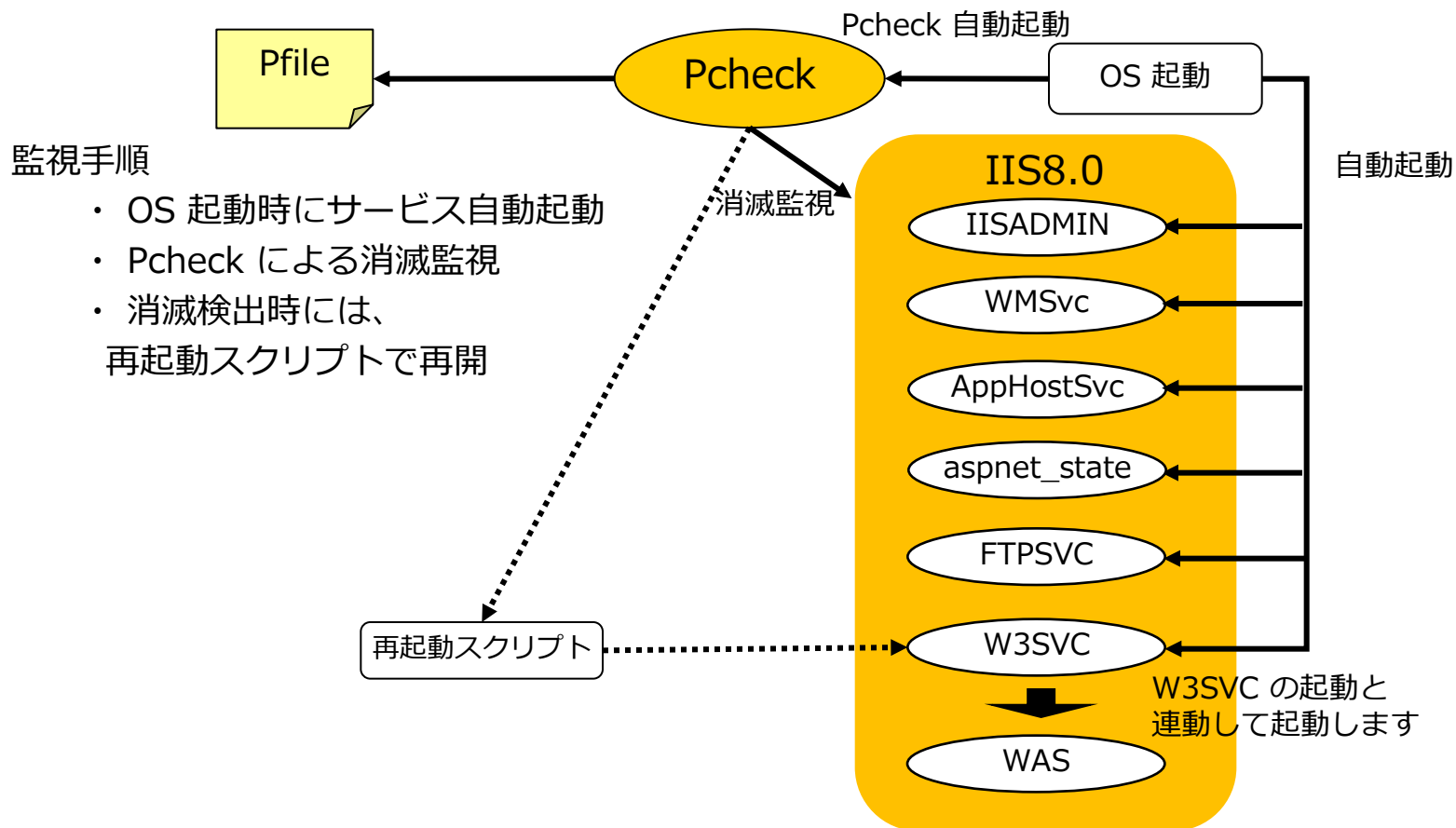
サービスの依存関係



例)



17. (続き)



17. (続き)

2. Pfile

Pfile <sample_iis8.0>

本 Pfile は IIS8.0 で 以下のサーバーまたは、機能を監視する Pfile です。

- Web サーバー (状態サーバーモードあり)
- リモート管理
- FTP サーバー

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。
その場合、「4. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥IIS8.dmp"  
UpMessageReduceMode   disable
```

次ページへ続く

17. (続き)

続き

```
##### PENT #####  
## Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;  
ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
  
## Web サーバー ##  
AppHostSvc,,86400,3,Continue,Service  
W3SVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS8_w3.bat",86400,3,Continue,Service  
WAS,,86400,3,Continue,Service  
  
## Web サーバー ( 状態サーバーモード ) ##  
aspnet_state,,86400,3,Continue,Service  
  
## リモート管理 ##  
WMSvc,,86400,3,Continue,Service  
  
## FTP サーバー ##  
FTPSVC,,86400,3,Continue,Service  
IISADMIN,,86400,3,Continue,Service
```

3. 再起動スクリプト

再起動スクリプト

- W3SVC <restart_IIS8_w3.bat>

```
@echo off

rem W3SVC の再起動スクリプトです

:REPEAT

rem サービスの起動失敗を防止するための処理を行います
sc query WAS | find "STOP_PENDING"
if %ERRORLEVEL% == 0 goto REPEAT

rem 起動処理
sc start W3SVC

exit
```

sc コマンドを使用してサービスの起動を行います。

17. (続き)

4. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- ・ 監視しないサービスについては、Pfile から削除してください。

以降、状態サーバーモードなしにする、または、リモート管理や FTP サーバー を監視しない場合の Pfile への書き換え手順を説明します。

17. (続き)

Pfile 書き換え手順

4-1. Pfile の直接編集



4-2. Pfile の読み込み



4-3. 監視しないサービスの削除



4-4. Pfile の保存

17. (続き)

4-1. Pfile の直接編集

(1) テキストエディターなどで

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_iis8.0> を開き、
監視しないサービスを削除するか、コメントアウトしてください。

```

.
.
.
## Web サーバー ##
AppHostSvc,,86400,3,Continue,Service
W3SVC,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS8_w3.bat",86400,3,Continue,Service
WAS,,86400,3,Continue,Service
## Web サーバー ( 状態サーバーモード ) ##
aspnet_state,,86400,3,Continue,Service
## リモート管理 ##
WMSvc,,86400,3,Continue,Service
## FTP サーバー ##
FTPSVC,,86400,3,Continue,Service
IISADMIN,,86400,3,Continue,Service
```

監視しないサービスの記載を削除してください。

を記載してコメントアウトしてください。

17. (続き)

4-2. Pfile の読み込み

- (1) 『Create Pfile』 で *Read* を押して
Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_iis8.0> の
読み込みを行ってください。

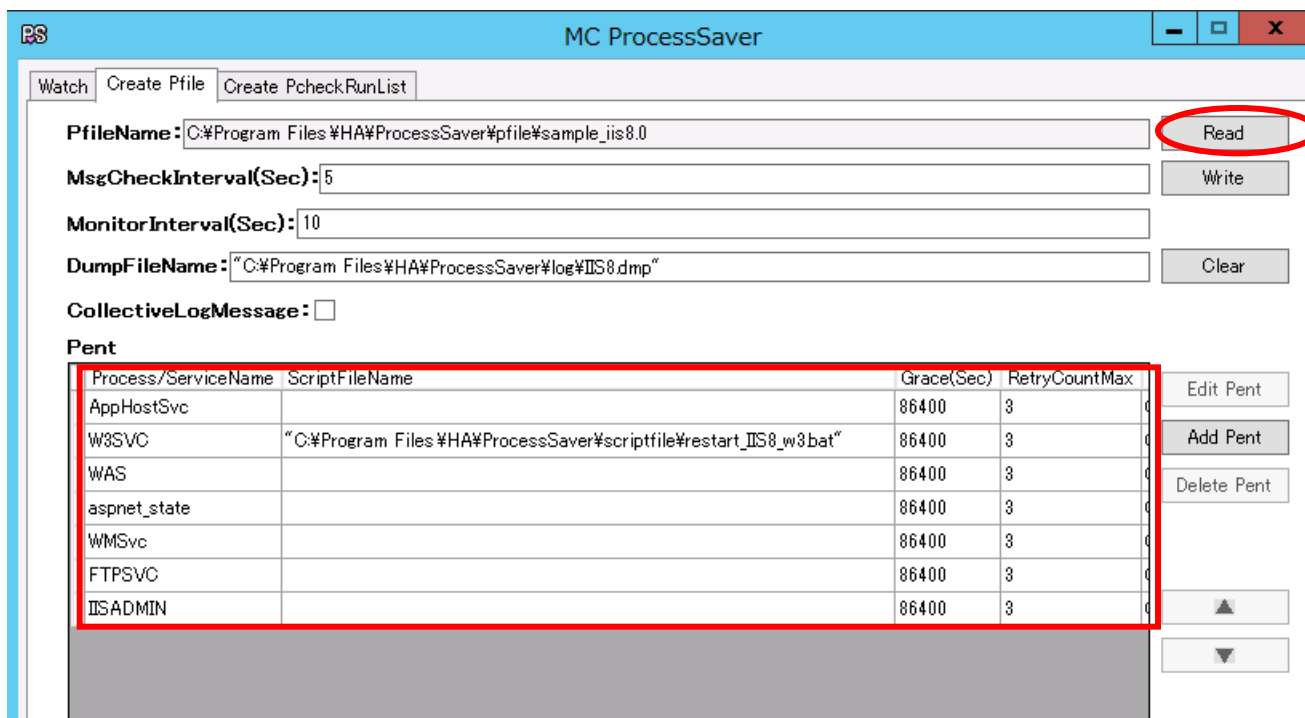


図12 『Create Pfile』

17. (続き)

4-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から、監視しないサービスを選択し、*Delete Pent* を押して削除してください。

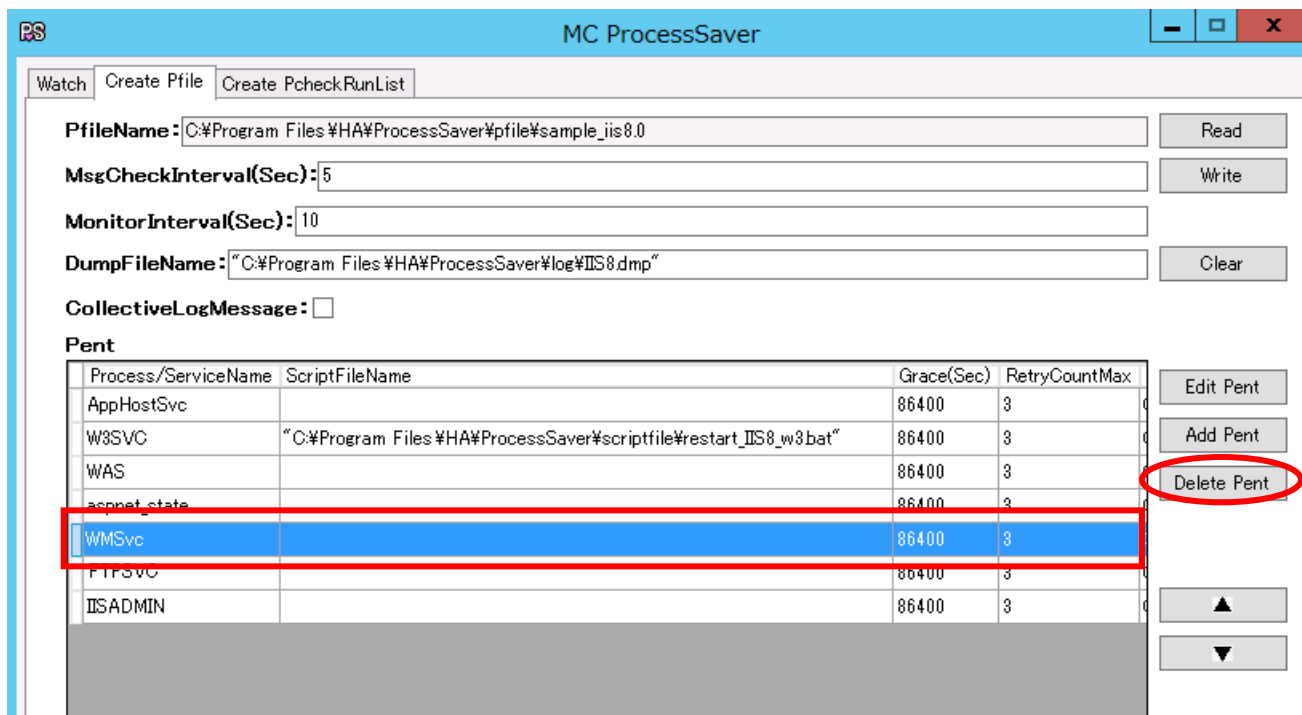


図13 『Create Pfile』

4-4. Pfile の保存

- (1) 『Create Pfile』 で Write を押して
Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_iis8.0> を保存してください。

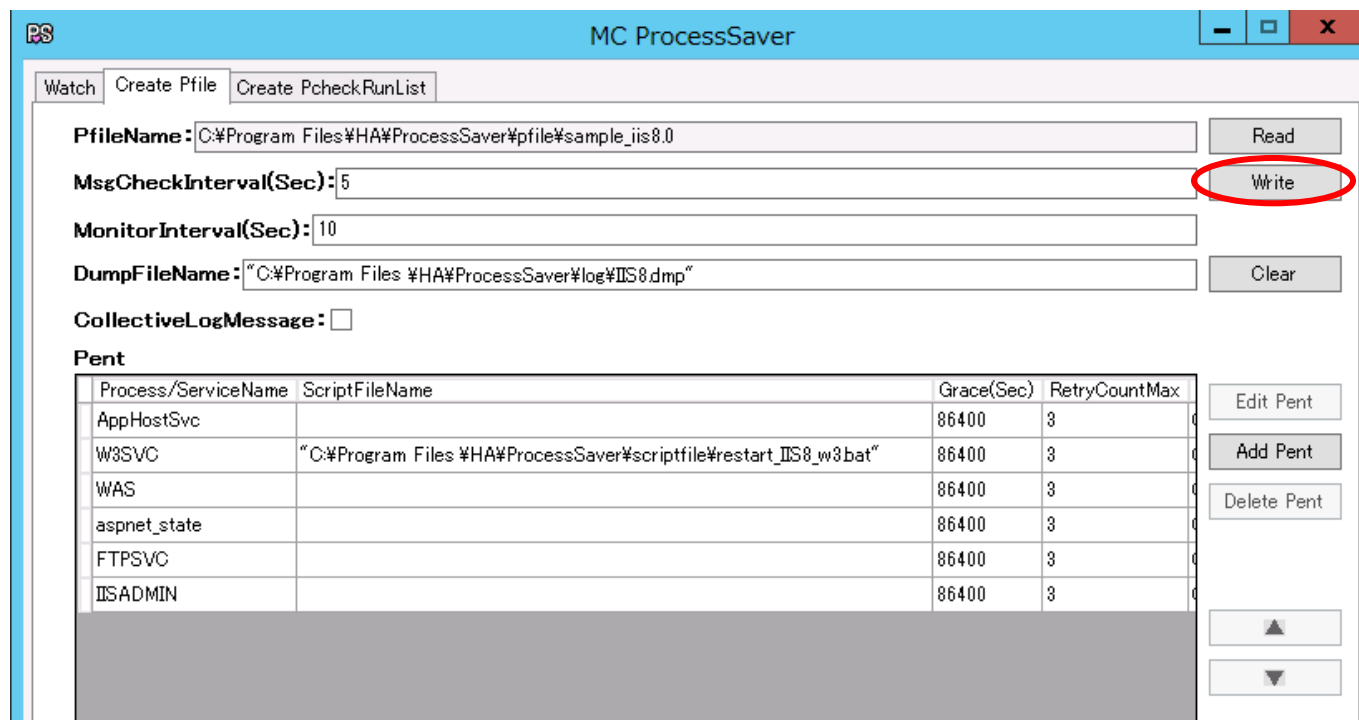


図14 『Create Pfile』

18. Microsoft SQL Server 2008 の監視事例

- ・ 本事例は Microsoft SQL Server 2008 を使用しています。
Microsoft SQL Server にて提供されているサービスを監視する場合を紹介します。

Microsoft SQL Server で提供されるサービスは以下のとおりですが、インストール時に選択する機能によって、登録されないサービスがあります。

(1) システム一意

Microsoft SQL Server 一意のサービスのため、複数登録されません。

- ・ SQLBrowser
- ・ SQLWriter
- ・ MsDtsServer100
- ・ MSSQLServerADHelper100

(2) インスタンス

インスタンス数分サービスが登録されます。

- ・ MSSQLSERVER
- ・ SQLSERVERAGENT
- ・ MSSQLFDLauncher
- ・ MSSQLServerOLAPService
- ・ ReportServer

18. (続き)

(2) (続き)

インスタンスのサービス名はインストール時のインスタンス名の設定方法によって、以下のようになります。

既定のインスタンスの場合	名前つきインスタンスの場合
MSSQLSERVER	MSSQL\$【インスタンス名】
SQLSERVERAGENT	SQLAgent\$【インスタンス名】
MSSQLFDLauncher	MSSQLFDLauncher\$【インスタンス名】
MSSQLServerOLAPService	MSOLAP\$【インスタンス名】
ReportServer	ReportServer\$【インスタンス名】

* 本書ではインスタンス名は既定のインスタンスで設定したものとします。

18. (続き)

1. システム一意

1-1. 概要

- SQLBrowser
- SQLWriter

タイプ	Service	Service
表示名	SQL Server Browser	SQL Server VSS Writer
サービス名	SQLBrowser	SQLWriter
常駐プロセス	sqlbrowser.exe	sqlwriter.exe
スタートアップ	自動	自動
回復機能	サービスを再起動する	何もしない
依存関係	なし	なし

- * スタートアップの値はインストールする機能の構成により変わる場合があります。
また、SQL Server インストール時に設定変更が可能となっています。

18. (続き)

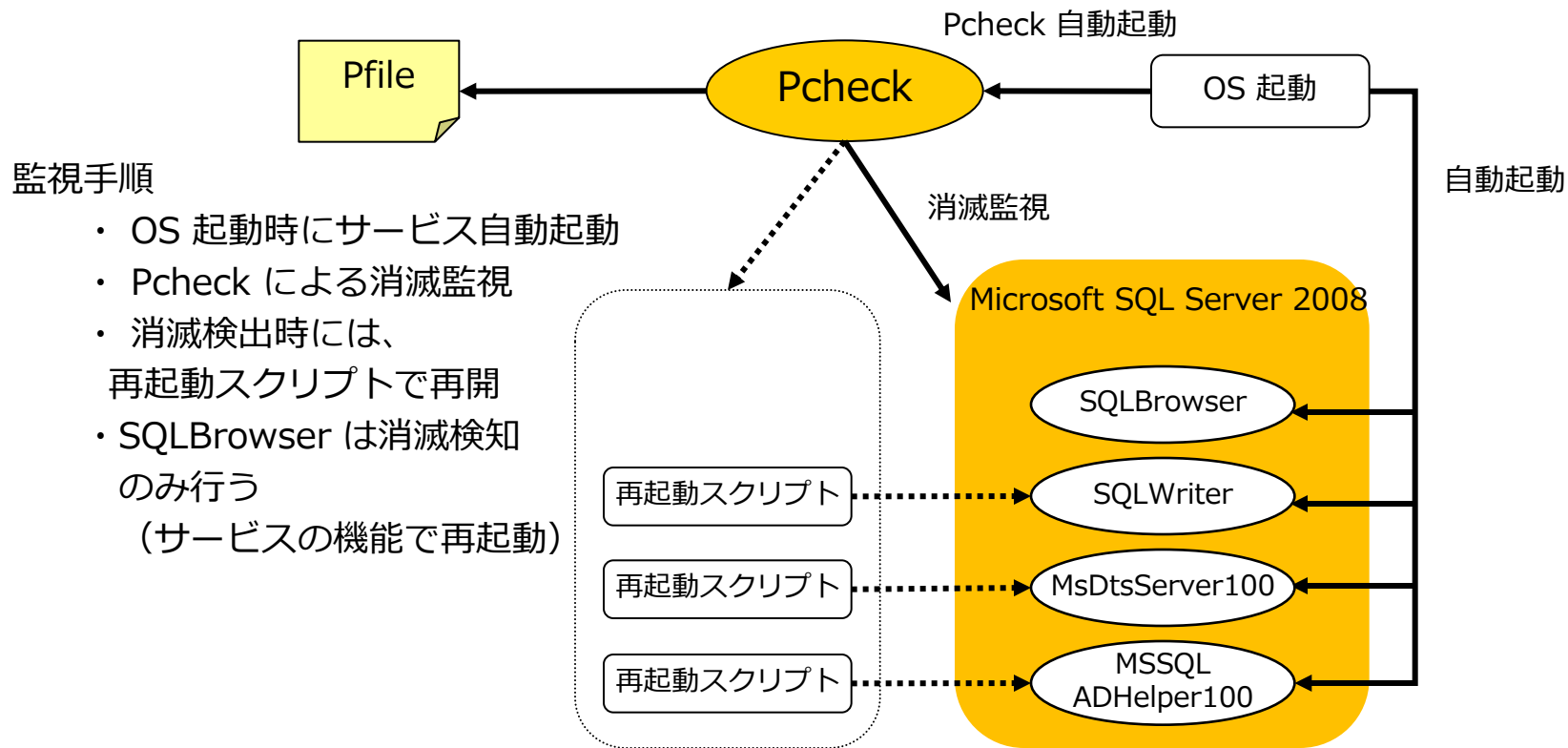
1-1. (続き)

- MsDtsServer100
- MSSQLServerADHelper100

タイプ	Service	Service
表示名	SQL Server Integration Services 10.0	SQL Active Directory Helper Service
サービス名	MsDtsServer100	MSSQLServerADHelper100
常駐プロセス	MsDtsSrvr.exe	SQLADHLP.EXE
スタートアップ	自動	無効
回復機能	何もしない	何もしない
依存関係	なし	なし

- * スタートアップの値はインストールする機能の構成により変わる場合があります。
また、SQL Server インストール時に設定変更が可能となっています。

18. (続き)



18. (続き)

1-2. Pfile

Pfile <sample_sqlserver_system>

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。
本 Pfile は MSSQLServerADHelper100 の監視はコメントアウトしていますので
MSSQLServerADHelper100 を監視するためには Pfile を変更する必要があります。
その場合、「3. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver_system.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCm
d;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
SQLBrowser,,86400,0,Continue,Service
SQLWriter,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlwriter.bat",86400,3,Continue,Service
MsDtsServer100,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_msdtsserver.bat",86400,3,Continue,Service
#MSSQLServerADHelper100,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_adhelper.bat",86400,3,Continue,Service
```

18. (続き)

1-3. 再起動スクリプト

- SQLWriter <restart_sqlwriter.bat>

```
@echo off
```

```
rem SQLWriter の再起動スクリプトです。  
sc start SQLWriter
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

- MsDtsServer100 <restart_msdtsserver.bat>

```
@echo off
```

```
rem MsDtsServer100 の再起動スクリプトです。  
sc start MsDtsServer100
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

- MSSQLServerADHelper100 <restart_adhelper.bat>

```
@echo off
```

```
rem MSSQLServerADHelper100 の再起動スクリプトです。  
sc start MSSQLServerADHelper100
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

2. インスタンス

2-1. 概要

- MSSQLSERVER
- SQLSERVERAGENT

タイプ		Service	Service
表示名	既定インスタンスの場合	SQL Server (MSSQLSERVER)	SQL Server Agent (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Server (【インスタンス名】)	SQL Server Agent (【インスタンス名】)
サービス名	既定インスタンスの場合	MSSQLSERVER	SQLSERVERAGENT
	名前つきインスタンスの場合	MSSQL\$【インスタンス名】	SQLAgent\$【インスタンス名】
常駐プロセス		sqlserver.exe	SQLAGENT.EXE
スタートアップ		自動	手動
回復機能		何もしない	何もしない
依存するサービス		なし	SQL Server (【インスタンス名】)
依存されるサービス		SQL Server Agent (【インスタンス名】)	なし

18. (続き)

2-1. (続き)

- MSSQLFDLauncher
- MSSQLServerOLAPService

タイプ		Service	Service
表示名	既定インスタンスの場合	SQL Full-text Filter Daemon Launcher (MSSQLSERVER)	SQL Server Analysis Services (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Full-text Filter Daemon Launcher (【インスタンス名】)	SQL Server Analysis Services (【インスタンス名】)
サービス名	既定インスタンスの場合	MSSQLFDLauncher	MSSQLServerOLAPService
	名前つきインスタンスの場合	MSSQLFDLauncher\$【インスタンス名】	MSOLAP\$【インスタンス名】
常駐プロセス		fdlauncher.exe fdhost.exe	msmdsrv.exe
スタートアップ		手動	自動
回復機能		何もしない	何もしない
依存関係		なし	なし

* MSSQLFDLauncher は MSSQLSERVER 起動時に起動されます。

18. (続き)

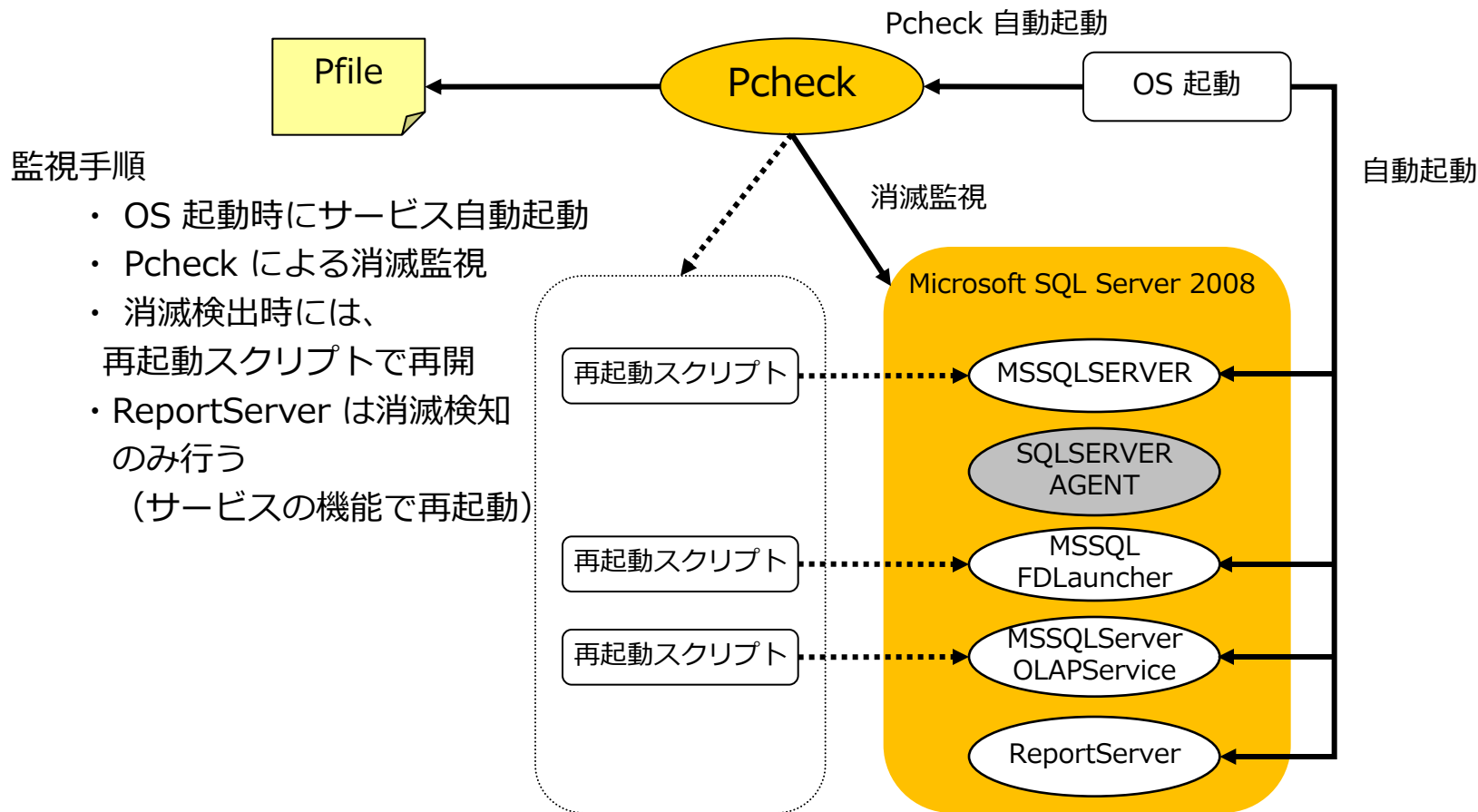
2-1. (続き)

- ReportServer

タイプ	Service	
表示名	既定インスタンスの場合	SQL Server Reporting Services (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Server Reporting Services (【インスタンス名】)
サービス名	既定インスタンスの場合	ReportServer
	名前つきインスタンスの場合	ReportServer\$【インスタンス名】
常駐プロセス	ReportingServicesService.exe	
スタートアップ	自動	
回復機能	サービスを再起動する	
依存関係	なし	

18. (続き)

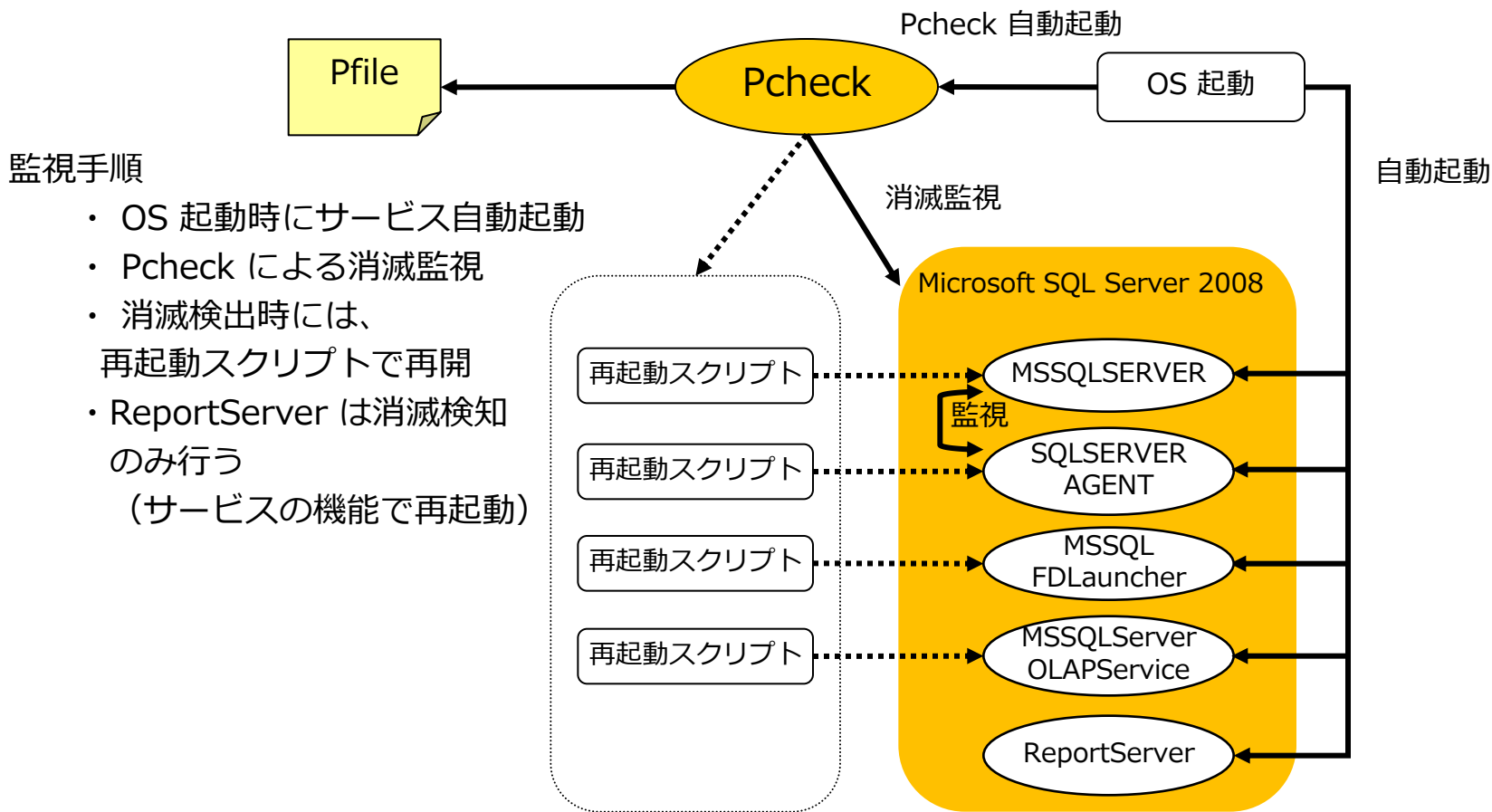
SQLSERVERAGENT 未使用時



18. (続き)

SQLSERVERAGENT 使用時

SQLSERVERAGENT を使用している場合、MSSQLSERVER と SQLSERVERAGENT は相互に監視を行い、異常時には、起動しているサービスが異常終了したサービスを起動します。ただし、両サービスが停止した場合は、再起動が行われないため、ProcessSaver にて各サービスを監視し、両サービス停止時に再起動を行います。



監視手順

- OS 起動時にサービス自動起動
- Pcheck による消滅監視
- 消滅検出時には、再起動スクリプトで再開
- ReportServer は消滅検知のみ行う
(サービスの機能で再起動)

18. (続き)

2-2. Pfile

Pfile <sample_sqlserver_instance_notuse_agent>

本 Pfile は SQLSERVERAGENT を使用しない場合の Pfile です。

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

本 Pfile は既定のインスタンスで作成されたサービスを監視しています。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver_instance_notuse_agent.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;In
cludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver.bat",86400,3,Continue,Service
MSSQLFDLauncher,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_olapservice.bat",86400,3,Continue,Service
ReportServer,,86400,0,Continue,Service
```

18. (続き)

2-2. (続き)

Pfile <sample_sqlserver_instance_use_agent>

本 Pfile は SQLSERVERAGENT を使用する場合の Pfile です。

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

本 Pfile は既定のインスタンスで作成されたサービスを監視しています。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver_instance_use_agent.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;In
cludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service
SQLSERVERAGENT,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlagent.bat",86400,3,Continue,Service
MSSQLFDLauncher,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_olapservice.bat",86400,3,Continue,Service
ReportServer,,86400,0,Continue,Service
```

18. (続き)

2-2. (続き)

名前付きインスタンスで作成されたサービスを監視する場合、使用する Pfile を以下のように変更してください。

```
##### PARAM #####
MsgCheckInterval 5
MonitorInterval 10
DumpFileName "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver_instance_use_agent.dmp"
UpMessageReduceMode disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service
SQLSERVERAGENT,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlagent.bat",86400,3,Continue,Service
MSSQLFDLauncher,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_olapservice.bat",86400,3,Continue,Service
ReportServer,,86400,0,Continue,Service
```

名前つきインスタンスのサービスを監視する場合は

各行の Process/ServiceName をそれぞれ以下のように変更してください。

MSSQLSERVER	→	MSSQL\$【インスタンス名】
SQLSERVERAGENT	→	SQLAgent\$【インスタンス名】
MSSQLFDLauncher	→	MSSQLFDLauncher\$【インスタンス名】
MSSQLServerOLAPService	→	MSOLAP\$【インスタンス名】
ReportServer	→	ReportServer\$【インスタンス名】

18. (続き)

2-3. 再起動スクリプト

本 スクリプトは既定のインスタンスで作成されたサービスを起動します。

名前つきインスタンスで作成されたサービスを起動するにはスクリプトを変更する必要があります。

- MSSQLSERVER <restart_sqlserver.bat>
本スクリプトは Pfile <sample_sqlserver_instance_notuse_agent> で使用します。

```
@echo off  
  
rem MSSQLSERVER の再起動スクリプトです。 }  
sc start MSSQLSERVER  
  
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は
MSSQL\$【インスタンス名】 に変更してください。

18. (続き)

2-3. (続き)

- MSSQLSERVER <restart_sqlserver_useagent.bat>
本スクリプトは Pfile <sample_sqlserver_instance_use_agent> で使用します。

```
@echo off
```

```
rem SQLSERVERAGENT 使用時の MSSQLSERVER の再起動スクリプトです。
```

```
rem SQLSERVERAGENT が起動している場合、何もしません。
```

```
sc query SQLSERVERAGENT | find "STOPPED"
```

```
if %ERRORLEVEL% == 1 exit
```

```
rem MSSQLSERVER, SQLSERVERAGENT を起動する。}
```

```
sc start MSSQLSERVER
```

```
sc start SQLSERVERAGENT
```

```
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は、以下のように変更してください。

MSSQLSERVER → MSSQL\$【インスタンス名】

SQLSERVERAGENT → SQLAgent\$【インスタンス名】

18. (続き)

2-3. (続き)

- SQLSERVERAGENT <restart_sqlagent.bat>
本スクリプトは Pfile <sample_sqlserver_instance_use_agent> で使用します。

```
@echo off

rem SQLSERVERAGENT の再起動スクリプトです。

rem MSSQLSERVER が起動している場合、何もしません。
sc query MSSQLSERVER | find "STOPPED"
if %ERRORLEVEL% == 1 exit

rem SQLSERVERAGENT を起動する。 }
sc start SQLSERVERAGENT          } sc コマンドを使用してサービスの起動を行います。

exit
```

名前つきインスタンスの場合は、以下のように変更してください。

MSSQLSERVER	→	MSSQL\$【インスタンス名】
SQLSERVERAGENT	→	SQLAgent\$【インスタンス名】

18. (続き)

2-3. (続き)

- MSSQLFDLauncher <restart_fdlauncher.bat>

```
@echo off  
rem MSSQLFDLauncher の再起動スクリプトです。 }  
sc start MSSQLFDLauncher  
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は
MSSQLFDLauncher\$【インスタンス名】 に変更してください。

- MSSQLServerOLAPService<restart_olapervice.bat>

```
@echo off  
rem MSSQLServerOLAPService の再起動スクリプトです。 }  
sc start MSSQLServerOLAPService  
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は
MSOLAP\$【インスタンス名】 に変更してください。

3. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- 監視しないサービスについては、Pfile から削除してください。
- MSSQLServerADHelper100 サービスの監視については、コメントアウトしています。MSSQLServerADHelper100 サービスを使用、監視する場合は、Pfile を書き換えて、コメントインしてください。
- SQLSERVERAGENT サービスを使用する場合、SQLSERVERAGENT サービスが MSSQLSERVER サービスを監視、再起動します。ProcessSaver による監視が不要な場合は Pfile から削除してください。

以降、システム一意のサービスの監視を削除する場合の Pfile の書き換え手順を説明します。

インスタンスのサービスの監視を削除する場合は、使用する Pfile

```
<C:¥Program
```

```
Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver_instance_notuse_agent>
```

または

```
<C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample¥sqlserver_instance_use_agent>
```

を「3-2. Pfile の読み込み」以降の手順にしたがい編集してください。

18. (続き)

Pfile 書き換え手順

3-1. Pfile の直接編集



3-2. Pfile の読み込み



3-3. 監視しないサービスの削除



3-4. Pfile の保存

18. (続き)

3-1. Pfile の直接編集

- (1) MSSQLServerADHelper100 を監視する場合、
テキストエディターなどで
Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver_system> を
開き、MSSQLServerADHelper100 の行の # を削除し、コメントインしてください。

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver_system.dmp"  
UpMessageReduceMode  disable  
  
##### PENT #####  
##  
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
SQLBrowser,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlbrowser.bat",86400,3,Continue,Service  
SQLWriter,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlwriter.bat",86400,3,Continue,Service  
MsDtsServer100,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_msdtsserver.bat",86400,3,Continue,Service  
#MSSQLServerADHelper100,"C:¥Program  
Files¥HA¥ProcessSaver¥scriptfile¥restart_adhelper.bat",86400,3,Continue,Service
```

を削除してコメントインしてください。

3-2. Pfile の読み込み

(1) 『Create Pfile』で Read を押して

Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_sqlserver_system>
の読み込みを行ってください。

下記画面にならなかった場合、または Pfile の読み込み時にエラーか警告が出力された場合は、
Page 18 の手順により Pfile を再取得し、もう一度やり直してください。

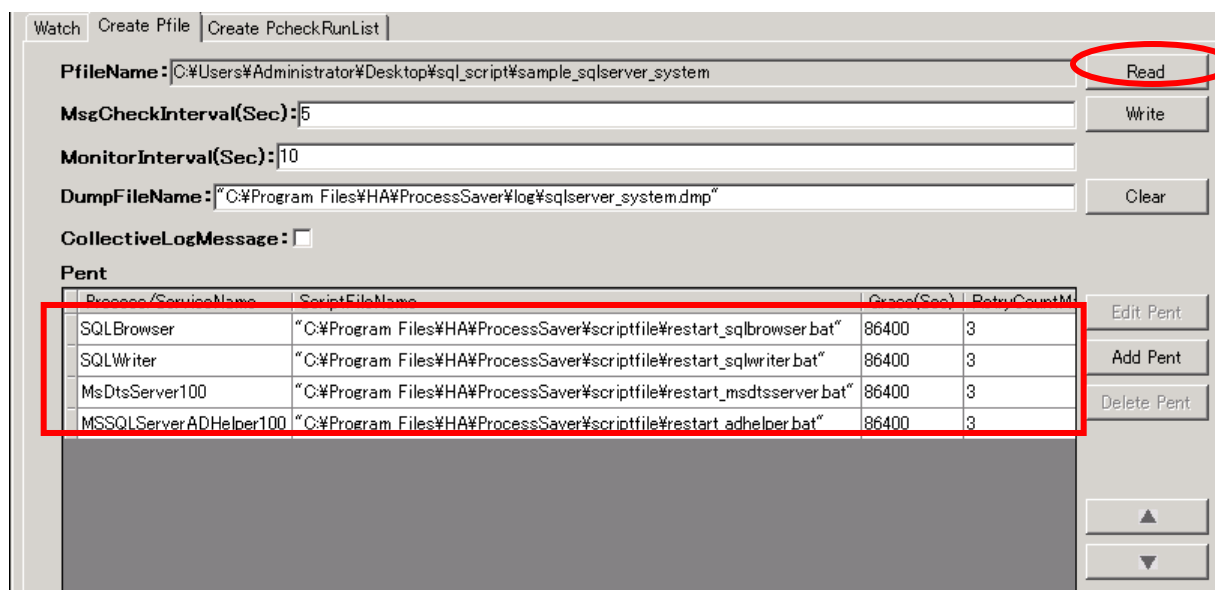


図15 『Create Pfile』

3-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から 監視しないサービスを選択し、*Delete Pent* を押して削除してください。

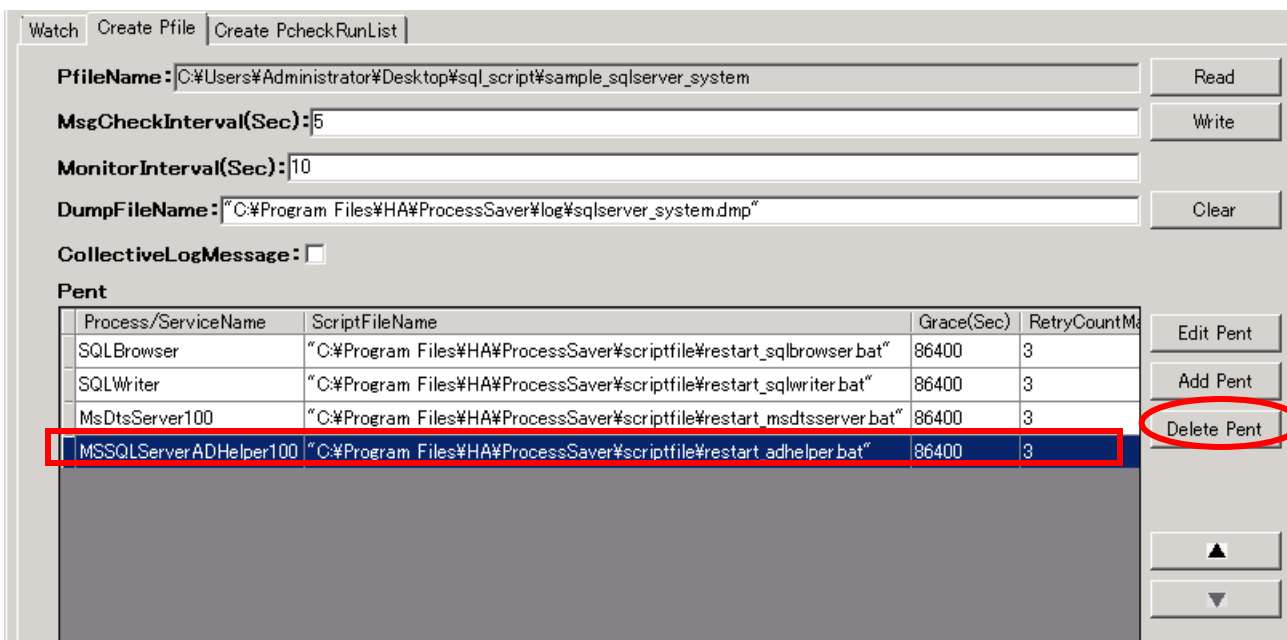


図16 『Create Pfile』

3-4. Pfile の保存

(1) 『Create Pfile』 で *Write* を押して

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver_system>
を保存してください。

Watch Create Pfile Create PcheckRunList

PfileName: C:\Users\Administrator\Desktop\sql_script\sample_sqlserver_system

MsgCheckInterval(Sec): 5

MonitorInterval(Sec): 10

DumpFileName: "C:\Program Files\HA\ProcessSaver\log\sqlserver_system.dmp"

CollectiveLogMessage:

Pent

Process/ServiceName	ScriptFileName	Grace(Sec)	RetryCountMax
SQLBrowser	"C:\Program Files\HA\ProcessSaver\scriptfile\restart_sqlbrowser.bat"	86400	3
SQLWriter	"C:\Program Files\HA\ProcessSaver\scriptfile\restart_sqlwriter.bat"	86400	3
MsDtsServer100	"C:\Program Files\HA\ProcessSaver\scriptfile\restart_mdtsserver.bat"	86400	3

Buttons: Read, Write, Clear, Edit Pent, Add Pent, Delete Pent, Up Arrow, Down Arrow

図17 『Create Pfile』

19. Microsoft SQL Server 2008 R2 の監視事例

- ・ 本事例は Microsoft SQL Server 2008 R2 を使用しています。
Microsoft SQL Server にて提供されているサービスを監視する場合を紹介します。

Microsoft SQL Server で提供されるサービスは以下のとおりですが、インストール時に選択する機能によって、登録されないサービスがあります。

(1) システム一意

Microsoft SQL Server 一意のサービスのため、複数登録されません。

- ・ SQLBrowser
- ・ SQLWriter
- ・ MsDtsServer100
- ・ MSSQLServerADHelper100

(2) インスタンス

インスタンス数分サービスが登録されます。

- ・ MSSQLSERVER
- ・ SQLSERVERAGENT
- ・ MSSQLFDLauncher
- ・ MSSQLServerOLAPService
- ・ ReportServer

19. (続き)

(2) (続き)

インスタンスのサービス名はインストール時のインスタンス名の設定方法によって、以下ようになります。

既定のインスタンスの場合	名前つきインスタンスの場合
MSSQLSERVER	MSSQL\$【インスタンス名】
SQLSERVERAGENT	SQLAgent\$【インスタンス名】
MSSQLFDLauncher	MSSQLFDLauncher\$【インスタンス名】
MSSQLServerOLAPService	MSOLAP\$【インスタンス名】
ReportServer	ReportServer\$【インスタンス名】

* 本書ではインスタンス名は既定のインスタンスで設定したものとします。

19. (続き)

1. システム一意

1-1. 概要

- SQLBrowser
- SQLWriter

タイプ	Service	Service
表示名	SQL Server Browser	SQL Server VSS Writer
サービス名	SQLBrowser	SQLWriter
常駐プロセス	sqlbrowser.exe	sqlwriter.exe
スタートアップ	無効	自動
回復機能	サービスを再起動する	何もしない
依存関係	なし	なし

- * スタートアップの値はインストールする機能の構成により変わる場合があります。
また、SQL Server インストール時に設定変更が可能となっています。

19. (続き)

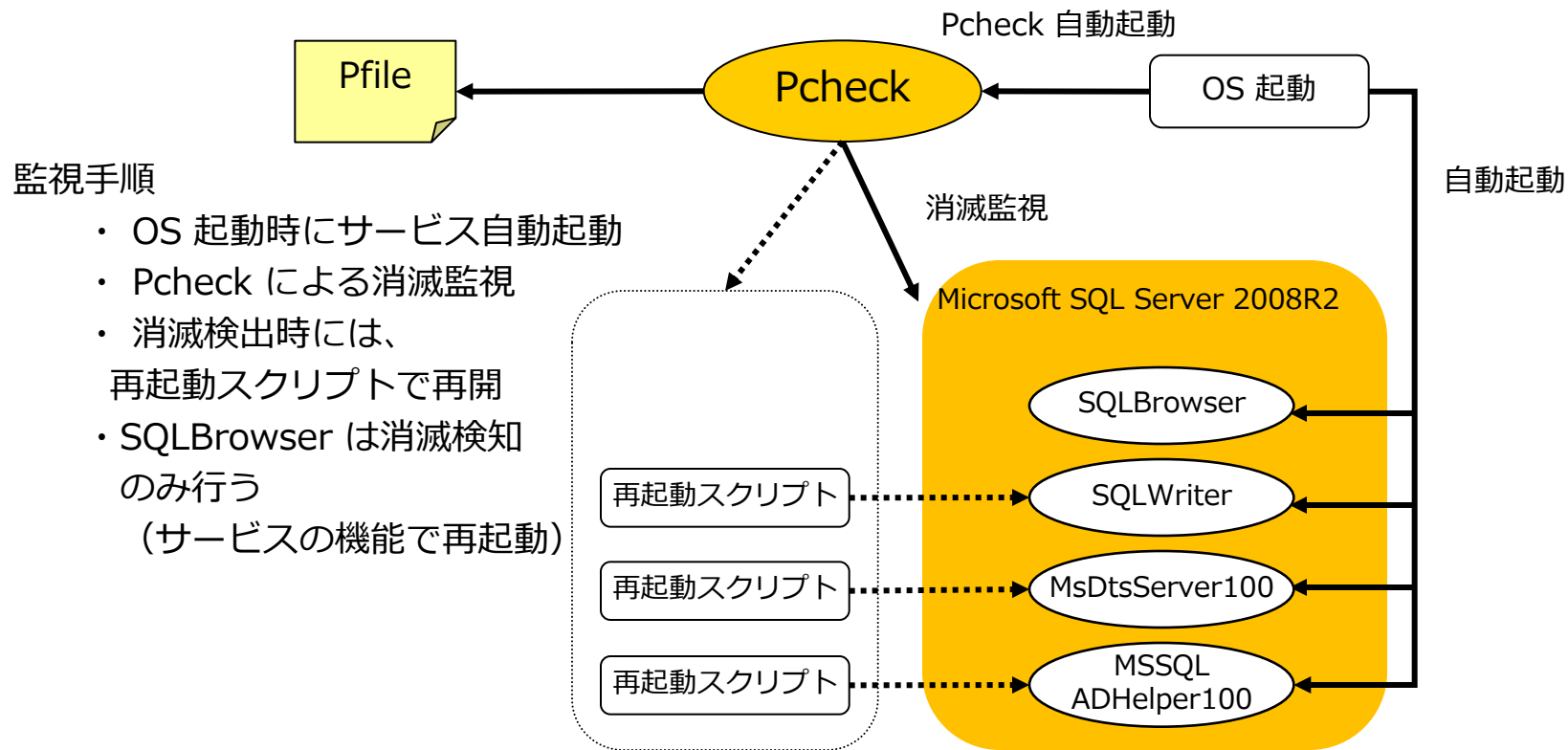
1-1. (続き)

- MsDtsServer100
- MSSQLServerADHelper100

タイプ	Service	Service
表示名	SQL Server Integration Services 10.0	SQL Active Directory Helper Service
サービス名	MsDtsServer100	MSSQLServerADHelper100
常駐プロセス	MsDtsSrvr.exe	SQLADHLP.EXE
スタートアップ	自動	無効
回復機能	何もしない	何もしない
依存関係	なし	なし

- * スタートアップの値はインストールする機能の構成により変わる場合があります。
また、SQL Server インストール時に設定変更が可能となっています。

19. (続き)



1-2. Pfile

Pfile <sample_sqlserver2008r2_system>

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

本 Pfile は SQLBrower、MSSQLServerADHelper100 の監視はコメントアウトしていますので SQLBrower、MSSQLServerADHelper100 を監視するためには Pfile を変更する必要があります。その場合、「3. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver2008r2_system.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
#SQLBrower,,86400,0,Continue,Service
SQLWriter,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlwriter.bat",86400,3,Continue,Service
MsDtsServer100,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_msdtsserver.bat",86400,3,Continue,Service
#MSSQLServerADHelper100,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_adhelper.bat",86400,3,Continue,Service
```

19. (続き)

1-3. 再起動スクリプト

- SQLWriter <restart_sqlwriter.bat>

```
@echo off
```

```
rem SQLWriter の再起動スクリプトです。  
sc start SQLWriter
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

- MsDtsServer100 <restart_msdtserver.bat>

```
@echo off
```

```
rem MsDtsServer100 の再起動スクリプトです。  
sc start MsDtsServer100
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

- MSSQLServerADHelper100 <restart_adhelper.bat>

```
@echo off
```

```
rem MSSQLServerADHelper100 の再起動スクリプトです。  
sc start MSSQLServerADHelper100
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

2. インスタンス

2-1. 概要

- MSSQLSERVER
- SQLSERVERAGENT

タイプ		Service	Service
表示名	既定インスタンスの場合	SQL Server (MSSQLSERVER)	SQL Server Agent (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Server (【インスタンス名】)	SQL Server Agent (【インスタンス名】)
サービス名	既定インスタンスの場合	MSSQLSERVER	SQLSERVERAGENT
	名前つきインスタンスの場合	MSSQL\$【インスタンス名】	SQLAgent\$【インスタンス名】
常駐プロセス		sqlserver.exe	SQLAGENT.EXE
スタートアップ		自動	手動
回復機能		何もしない	何もしない
依存するサービス		なし	SQL Server (【インスタンス名】)
依存されるサービス		SQL Server Agent (【インスタンス名】)	なし

19. (続き)

2-1. (続き)

- MSSQLFDLauncher
- MSSQLServerOLAPService

タイプ		Service	Service
表示名	既定インスタンスの場合	SQL Full-text Filter Daemon Launcher (MSSQLSERVER)	SQL Server Analysis Services (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Full-text Filter Daemon Launcher (【インスタンス名】)	SQL Server Analysis Services (【インスタンス名】)
サービス名	既定インスタンスの場合	MSSQLFDLauncher	MSSQLServerOLAPService
	名前つきインスタンスの場合	MSSQLFDLauncher\$【インスタンス名】	MSOLAP\$【インスタンス名】
常駐プロセス		fdlauncher.exe fdhost.exe	msmdsrv.exe
スタートアップ		手動	自動
回復機能		何もしない	何もしない
依存関係		なし	なし

* MSSQLFDLauncher は MSSQLSERVER 起動時に起動されます。

19. (続き)

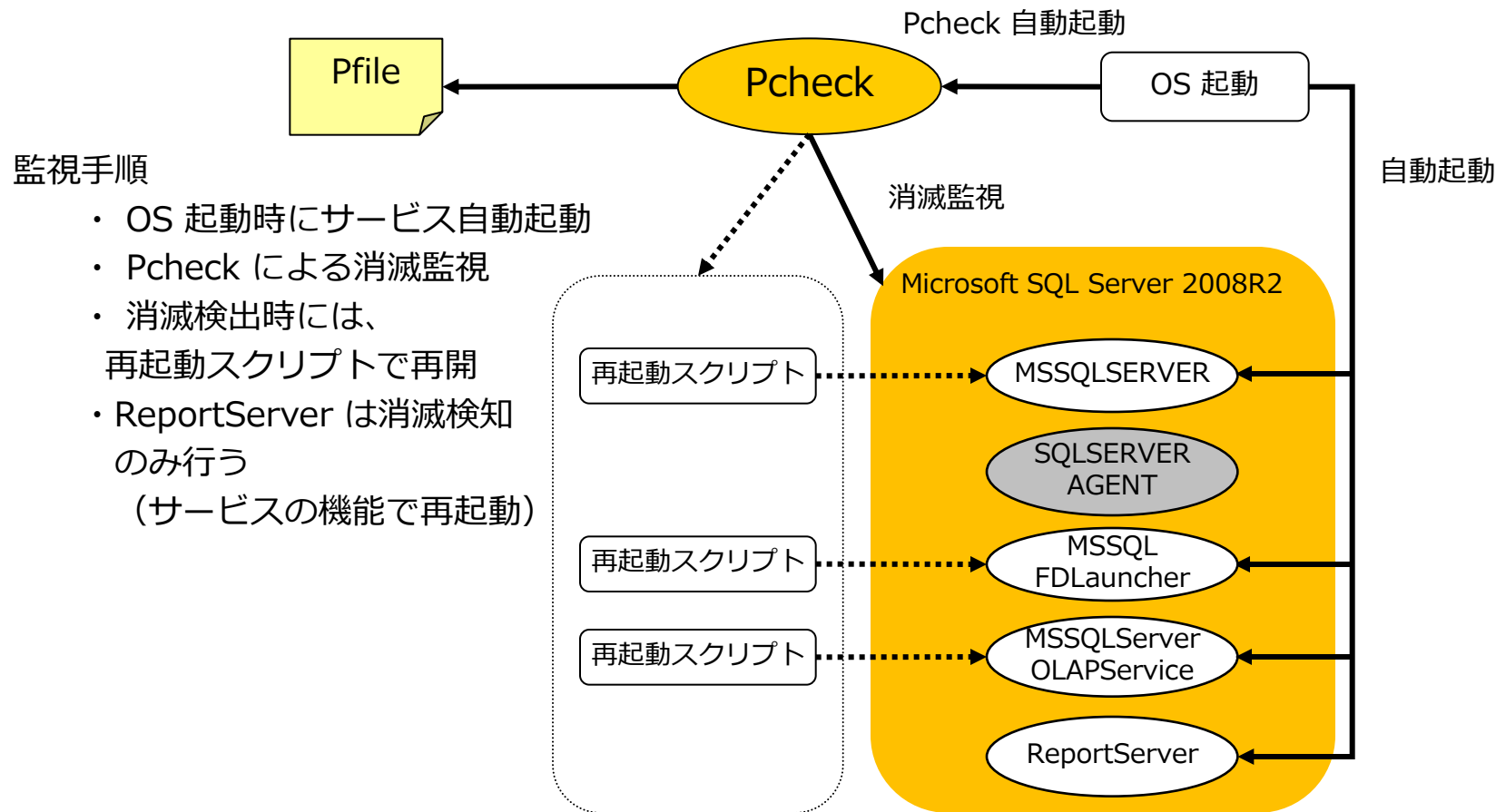
2-1. (続き)

- ReportServer

タイプ		Service
表示名	既定インスタンスの場合	SQL Server Reporting Services (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Server Reporting Services (【インスタンス名】)
サービス名	既定インスタンスの場合	ReportServer
	名前つきインスタンスの場合	ReportServer\$【インスタンス名】
常駐プロセス		ReportingServicesService.exe
スタートアップ		自動
回復機能		サービスを再起動する
依存関係		なし

19. (続き)

SQLSERVERAGENT 未使用時



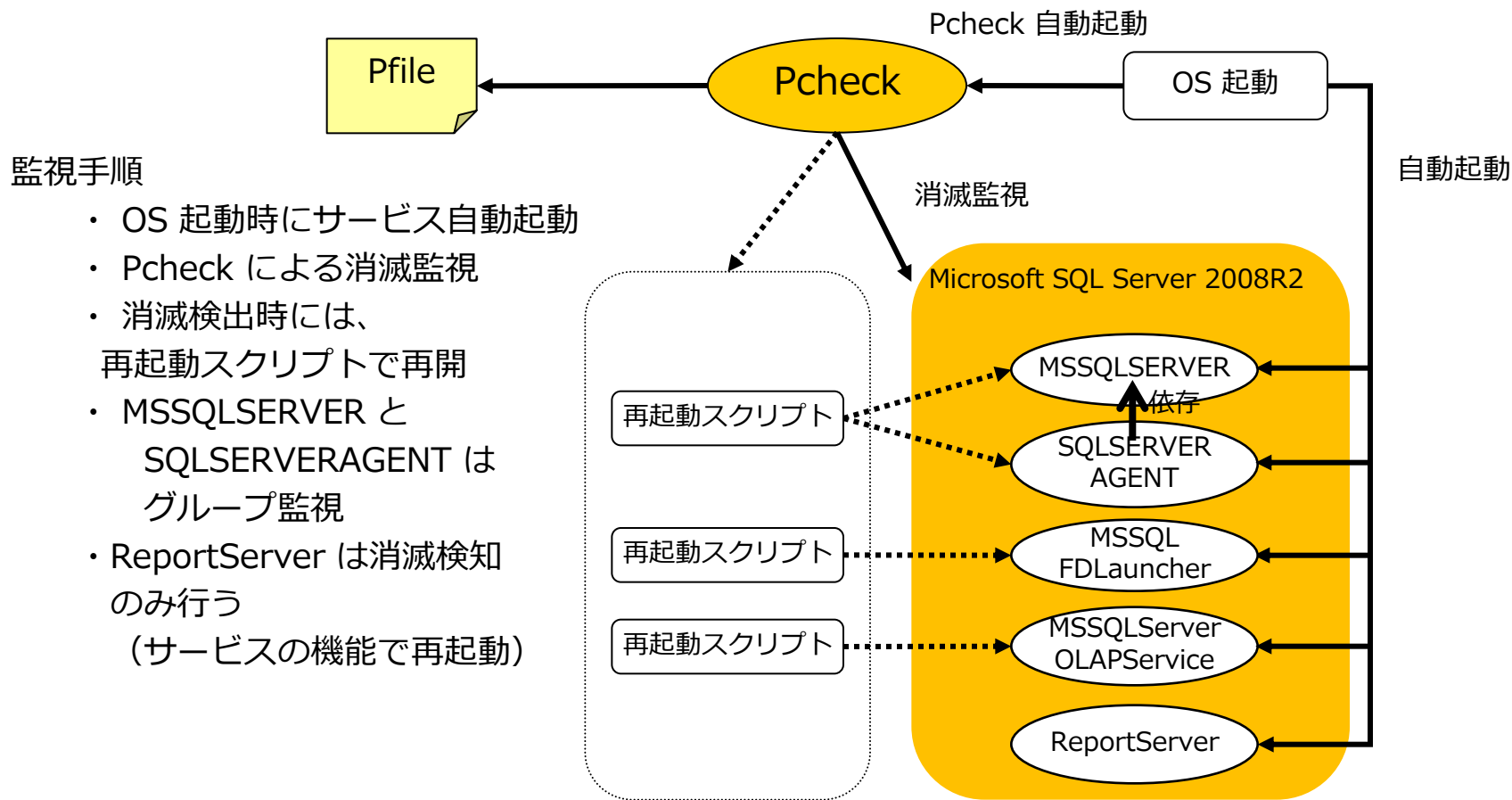
19. (続き)

SQLSERVERAGENT 使用時

SQLSERVERAGENT を使用している場合、SQLSERVERAGENT は、MSSQLSERVER に依存するため、MSSQLSERVER 異常終了時には、SQLSERVERAGENT も停止します。

ただし、SQLSERVERAGENT 異常終了時には、MSSQLSERVER は停止しません。

ProcessSaver にて各サービスを監視し、停止したサービスの再起動を行います。



2-2. Pfile

Pfile <sample_sqlserver2008r2_instance_notuse_agent>

本 Pfile は SQLSERVERAGENT を使用しない場合の Pfile です。
本 Pfile は使用する機能により、監視するサービスを削除する必要があります。
本 Pfile は既定のインスタンスで作成されたサービスを監視しています。
本 Pfile は ReportServer の監視はコメントアウトしていますので ReportServer を監視するためには Pfile を変更する必要があります。
その場合、「3. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####  
MsgCheckInterval      5  
MonitorInterval       10  
DumpFileName          "C:\Program Files\HA\ProcessSaver\log\sqlserver2008r2_instance_notuse_agent.dmp"  
UpMessageReduceMode   disable  
  
##### PENT #####  
##Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd  
;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch  
MSSQLSERVER,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_sqlserver.bat",86400,3,Continue,Service  
MSSQLFDLauncher,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_fdlauncher.bat",86400,3,Continue,Service  
MSSQLServerOLAPService,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_olapservice.bat",86400,3,Continue,Service  
#ReportServer,,86400,0,Continue,Service
```

ReportServer サービスの自動再起動は、サービス異常終了後、デフォルトで 1分後に再起動されるため、
本テンプレートの ReportServer の監視をコメントインして監視する場合、ProcessSaver で消滅検知後すぐに RetryOver となります。
ReportServer サービスを監視する場合で、消滅検知後、すぐに RetryOver にしたくない場合は、ReportServer サービスの [プロパティ] から
再起動時間を変更していただくか、ReportServer サービス再起動を待ち合わせる再起動スクリプトをご用意いただく、または、MonitorInterval の
値を変更する必要があります。
RetryOver になった場合は、サービス情報を再読み込み後にサービス監視の再開を行うために ReportServer サービスが 1 分後に起動したことを
確認してから、Restart ボタンを押してください。

19. (続き)

2-2. (続き)

Pfile <sample_sqlserver2008r2_instance_use_agent>

本 Pfile は SQLSERVERAGENT を使用する場合の Pfile です。

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

本 Pfile は既定のインスタンスで作成されたサービスを監視しています。

本 Pfile は ReportServer の監視はコメントアウトしていますので ReportServer を監視するためには Pfile を変更する必要があります。

その場合、「3. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver2008r2_instance_use_agent.dmp"
UpMessageReduceMode   disable

##### PENT #####
##Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd
;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
SQLSERVERAGENT,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
MSSQLFDLauncher,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_olapservice.bat",86400,3,Continue,Service
#ReportServer,,86400,0,Continue,Service
```

ReportServer サービスの自動再起動は、サービス異常終了後、デフォルトで 1分後に再起動されるため、本テンプレートの ReportServer の監視をコメントインして監視する場合、ProcessSaver で消滅検知後すぐに RetryOver となります。ReportServer サービスを監視する場合で、消滅検知後、すぐに RetryOver にしたくない場合は、ReportServer サービスの [プロパティ] から再起動時間を変更していただくか、ReportServer サービス再起動を待ち合わせる再起動スクリプトをご用意いただく、または、MonitorInterval の値を変更する必要があります。RetryOver になった場合は、サービス情報を再読み込み後にサービス監視の再開を行うために ReportServer サービスが 1 分後に起動したことを確認してから、Restart ボタンを押してください。

19. (続き)

2-2. (続き)

名前付きインスタンスで作成されたサービスを監視する場合、使用する Pfile を以下のように変更してください。

```
##### PARAM #####
MsgCheckInterval 5
MonitorInterval 10
DumpFileName "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver2008r2_instance_use_agent.dmp"
UpMessageReduceMode disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
SQLSERVERAGENT,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
MSSQLFDLauncher,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_olapservice.bat",86400,3,Continue,Service
#ReportServer,,86400,0,Continue,Service
```

名前つきインスタンスのサービスを監視する場合は
各行の Process/ServiceName をそれぞれ以下のように変更してください。

MSSQLSERVER	→	MSSQL\$【インスタンス名】
SQLSERVERAGENT	→	SQLAgent\$【インスタンス名】
MSSQLFDLauncher	→	MSSQLFDLauncher\$【インスタンス名】
MSSQLServerOLAPService	→	MSOLAP\$【インスタンス名】
ReportServer	→	ReportServer\$【インスタンス名】

19. (続き)

2-3. 再起動スクリプト

- MSSQLSERVER <restart_sqlserver.bat>
本スクリプトは Pfile <sample_sqlserver2008r2_instance_notuse_agent> で使用します。

```
@echo off
```

```
rem MSSQLSERVER の再起動スクリプトです。 }
```

```
sc start MSSQLSERVER
```

```
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は
MSSQL\$【インスタンス名】に変更してください。

19. (続き)

2-3. (続き)

- MSSQLSERVER <restart_sqlserver_useagent.bat>
本スクリプトは Pfile <sample_sqlserver2008r2_instance_use_agent> で使用します。

```
@echo off

rem SQLSERVERAGENT 使用時の MSSQLSERVER の再起動スクリプトです。

rem MSSQLSERVER が起動していない場合、MSSQLSERVER と SQLSERVERAGENT を起動する。
sc query MSSQLSERVER | find "STOPPED"
if %ERRORLEVEL% == 0 (
  sc start MSSQLSERVER
)

rem SQLSERVERAGENT が起動していない場合、SQLSERVERAGENT を起動する。
sc query SQLSERVERAGENT | find "STOPPED"
if %ERRORLEVEL% == 0 (
  sc start SQLSERVERAGENT
)

exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は、以下のように変更してください。
MSSQLSERVER → MSSQL\$【インスタンス名】
SQLSERVERAGENT → SQLAgent\$【インスタンス名】

19. (続き)

2-3. (続き)

- MSSQLFDLauncher <restart_fdlauncher.bat>

```
@echo off  
rem MSSQLFDLauncher の再起動スクリプトです。 } sc コマンドを使用してサービスの起動を行います。  
sc start MSSQLFDLauncher  
exit
```

名前つきインスタンスの場合は
MSSQLFDLauncher\$【インスタンス名】に変更してください。

- MSSQLServerOLAPService<restart_olapervice.bat>

```
@echo off  
rem MSSQLServerOLAPService の再起動スクリプトです。 } sc コマンドを使用してサービスの起動を行います。  
sc start MSSQLServerOLAPService  
exit
```

名前つきインスタンスの場合は
MSOLAP\$【インスタンス名】に変更してください。

3. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- ・ 監視しないサービスについては、Pfile から削除してください。
- ・ SQLBrowser、MSSQLServerADHelper100、ReportServer サービスの監視については、コメントアウトしています。SQLBrowser、MSSQLServerADHelper100、ReportServer サービスを使用、監視する場合は、Pfile を書き換えて、コメントインしてください。
- ・ SQLSERVERAGENT サービスを使用する場合、SQLSERVERAGENT サービスが MSSQLSERVER サービスに依存します。ProcessSaver による監視が不要な場合は Pfile から削除してください。

以降、システム一意のサービスの監視を削除する場合の Pfile の書き換え手順を説明します。

インスタンスのサービスの監視を削除する場合は、使用する Pfile

```
<C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver2008r2_instance_notuse_agent>
```

または

```
<C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver2008r2_instance_use_agent>
```

を「3-2. Pfile の読み込み」以降の手順にしたがい編集してください。

19. (続き)

Pfile 書き換え手順

3-1. Pfile の直接編集



3-2. Pfile の読み込み



3-3. 監視しないサービスの削除



3-4. Pfile の保存

3-1. Pfile の直接編集

- (1) SQLBrowser、MSSQLServerADHelper100 を監視する場合、テキストエディターなどで Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver2008r2_system> を開き、SQLBrowser、MSSQLServerADHelper100 の行の # を削除し、コメントインしてください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver2008r2_system.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace( Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
# SQLBrowser,,86400,0,Continue,Service
SQLWriter,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlwriter.bat",86400,3,Continue,Service
MsDtsServer100,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_msdtsserver.bat",86400,3,Continue,Service
# MSSQLServerADHelper100,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_adhelper.bat",86400,3,Continue,Service
```

を削除してコメントインしてください。

SQLBrowser サービスの自動再起動は、サービス異常終了後、デフォルトで 1分後に再起動されるため、本テンプレートの SQLBrowser の行をコメントインして監視する場合、ProcessSaver で消滅検知後すぐに RetryOver となります。SQLBrowser を監視する場合で、消滅検知後すぐに RetryOver にしたくない場合は、SQLBrowser サービスの [プロパティ] から再起動時間を変更していただくか、SQLBrowser サービス再起動を待ち合わせる再起動スクリプトをご用意いただく、または、MonitorInterval の値を変更する必要があります。

RetryOver になった場合は、サービス情報を再読み込み後にサービス監視の再開を行うために SQLBrowser サービスが 1分後に起動したことを確認してから、Restart ボタンを押してください。

3-2. Pfile の読み込み

(1) 『Create Pfile』 で *Read* を押して

Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_sqlserver2008r2_system>
の読み込みを行ってください。

下記画面にならなかった場合、または Pfile の読み込み時にエラーか警告が出力された場合は、
Page 18 の手順により Pfile を再取得し、もう一度やり直してください。

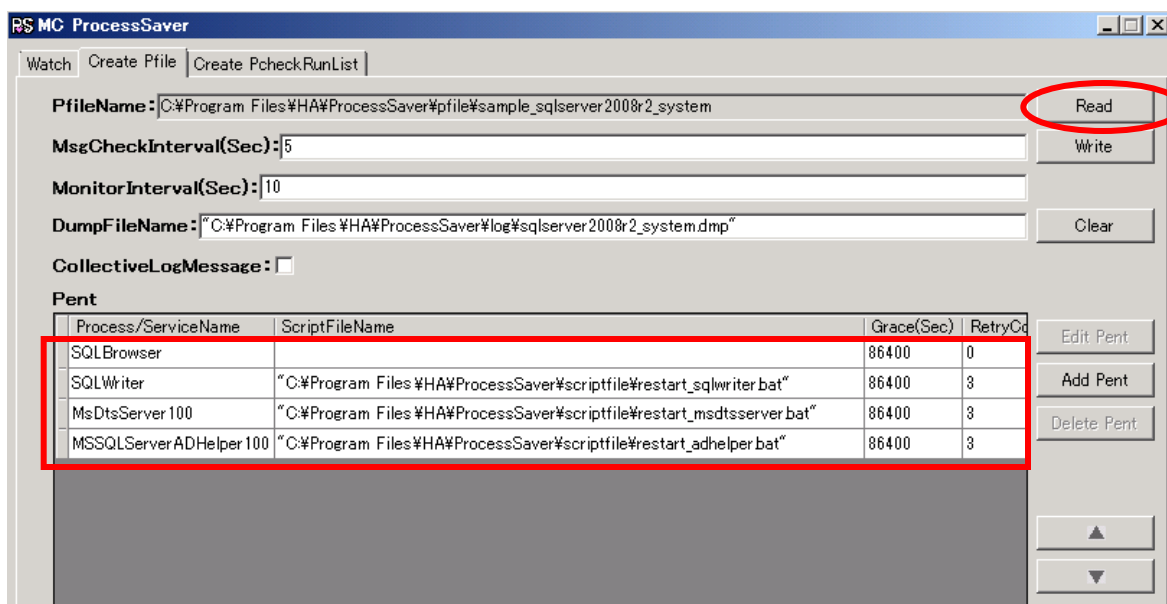


図18 『Create Pfile』

3-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から 監視しないサービスを選択し、*Delete Pent* を押して削除してください。

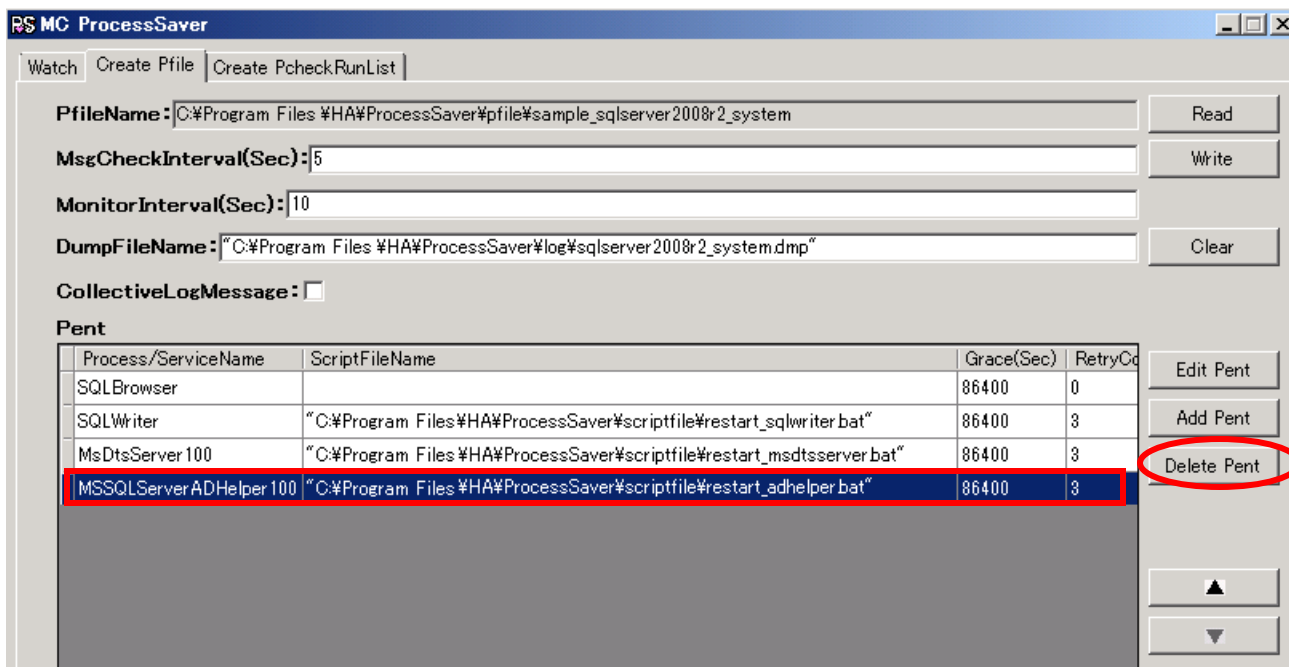


図19 『Create Pfile』

3-4. Pfile の保存

(1) 『Create Pfile』 で Write を押して

Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_sqlserver2008r2_system>
を保存してください。

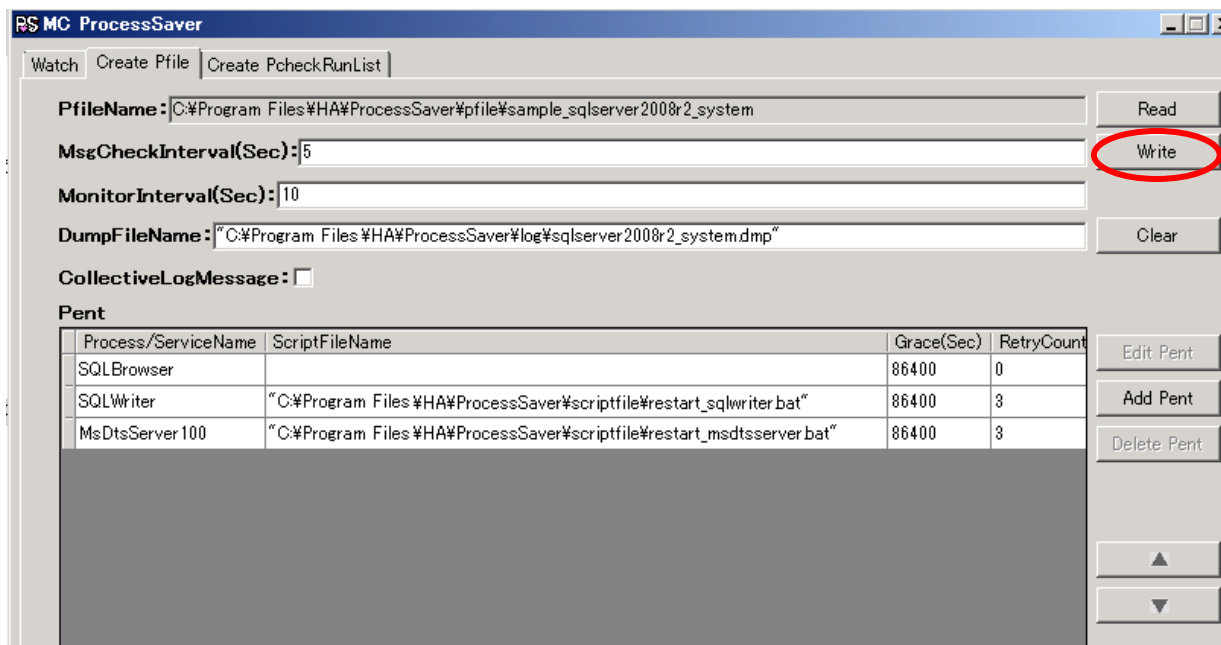


図20 『Create Pfile』

20. Microsoft SQL Server 2012 の監視事例

- ・ 本事例は Microsoft SQL Server 2012 ServicePack1 を使用しています。
Microsoft SQL Server にて提供されているサービスを監視する場合を紹介します。

Microsoft SQL Server で提供されるサービスは以下のとおりですが、インストール時に選択する機能によって、登録されないサービスがあります。

(1) システム一意

Microsoft SQL Server 一意のサービスのため、複数登録されません。

- ・ SQLBrowser
- ・ SQLWriter
- ・ MsDtsServer110
- ・ SQL Server Distributed Replay Client
- ・ SQL Server Distributed Replay Controller

(2) インスタンス

インスタンス数分サービスが登録されます。

- ・ MSSQLSERVER
- ・ SQLSERVERAGENT
- ・ MSSQLFDLauncher
- ・ MSSQLServerOLAPService
- ・ ReportServer

20. (続き)

(2) (続き)

インスタンスのサービス名はインストール時のインスタンス名の設定方法によって、以下ようになります。

既定のインスタンスの場合	名前つきインスタンスの場合
MSSQLSERVER	MSSQL\$【インスタンス名】
SQLSERVERAGENT	SQLAgent\$【インスタンス名】
MSSQLFDLauncher	MSSQLFDLauncher\$【インスタンス名】
MSSQLServerOLAPService	MSOLAP\$【インスタンス名】
ReportServer	ReportServer\$【インスタンス名】

* 本書ではインスタンス名は既定のインスタンスで設定したものとします。

1. システム一意

1-1. 概要

- SQLBrowser
- SQLWriter

タイプ	Service	Service
表示名	SQL Server Browser	SQL Server VSS Writer
サービス名	SQLBrowser	SQLWriter
常駐プロセス	sqlbrowser.exe	sqlwriter.exe
スタートアップ	無効	自動
回復機能	サービスを再起動する	何もしない
依存関係	なし	なし

- * スタートアップの値はインストールする機能の構成により変わる場合があります。
また、SQL Server インストール時に設定変更が可能となっています。

20. (続き)

1-1. (続き)

- MsDtsServer110

タイプ	Service
表示名	SQL Server Integration Services 11.0
サービス名	MsDtsServer110
常駐プロセス	MsDtsSrvr.exe
スタートアップ	自動
回復機能	何もしない
依存関係	なし

- * スタートアップの値はインストールする機能の構成により変わる場合があります。また、SQL Server インストール時に設定変更が可能となっています。

20. (続き)

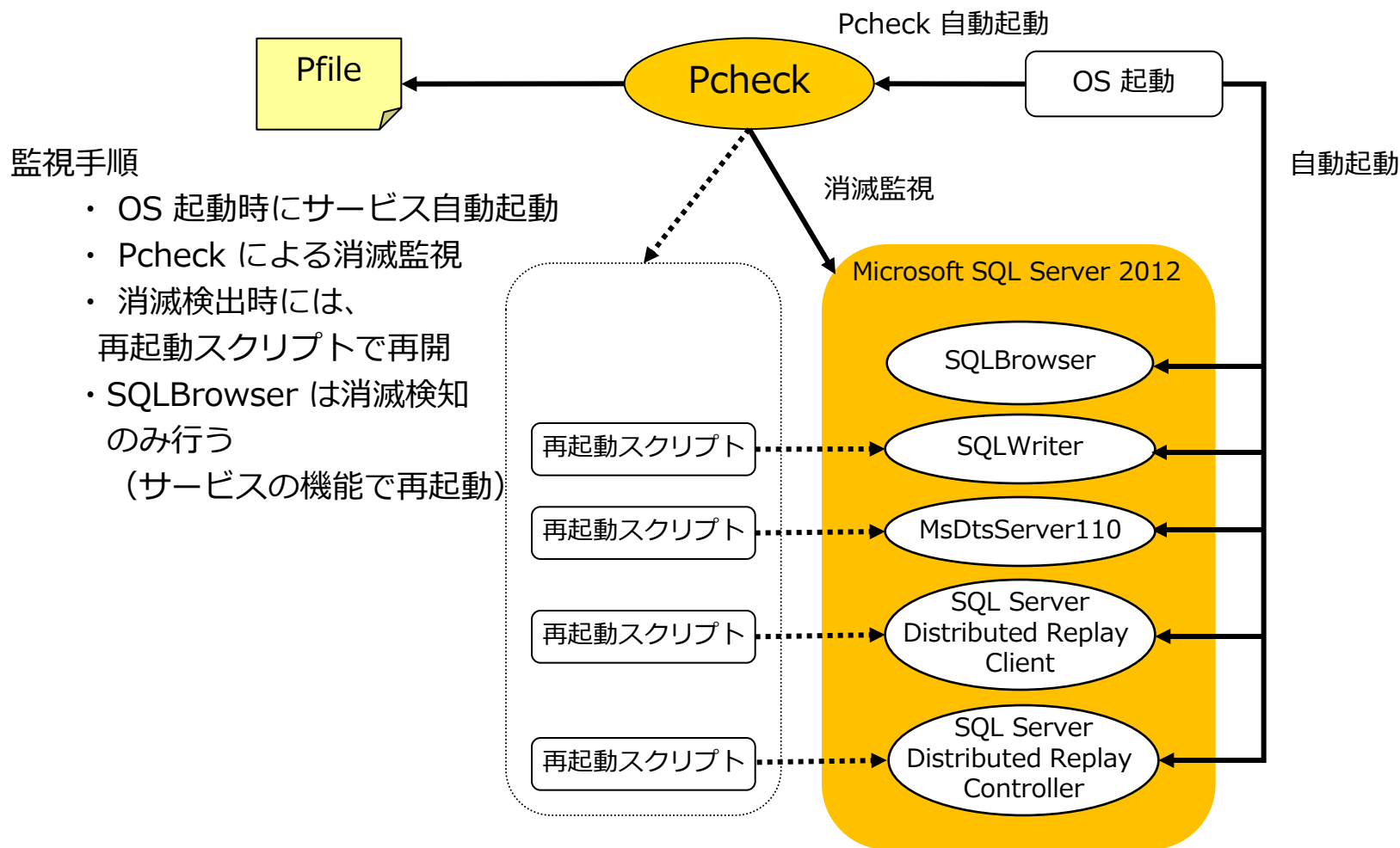
1-1. (続き)

- SQL Server Distributed Replay Client
- SQL Server Distributed Replay Controller

タイプ	Service	Service
表示名	SQL Server Distributed Replay Client	SQL Server Distributed Replay Controller
サービス名	SQL Server Distributed Replay Client	SQL Server Distributed Replay Controller
常駐プロセス	DReplayClient.exe	DReplayController.exe
スタートアップ	手動	手動
回復機能	何もしない	何もしない
依存関係	なし	なし

- * スタートアップの値はインストールする機能の構成により変わる場合があります。
また、SQL Server インストール時に設定変更が可能となっています。

20. (続き)



20. (続き)

1-2. Pfile

Pfile <sample_sqlserver2012_system>

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

本 Pfile は SQLBrowser、SQL Server Distributed Replay Client、SQL Server Distributed Replay Controller の監視はコメントアウトしていますので

SQLBrowser、SQL Server Distributed Replay Client、SQL Server Distributed Replay Controller を監視するためには Pfile を変更する必要があります。

その場合、「3. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver2012_system.dmp"
UpMessageReduceMode   disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
#SQLBrowser,,86400,0,Continue,Service
SQLWriter,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlwriter.bat",86400,3,Continue,Service
MsDtsServer110,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_msdtsserver110.bat",86400,3,Continue,Service
#SQL Server Distributed Replay Client,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_distributedreplayclient.bat",86400,3,Continue,Service
#SQL Server Distributed Replay Controller,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_distributedreplaycontroller.bat",86400,3,Continue,Service
```

1-3. 再起動スクリプト

- SQLWriter <restart_sqlwriter.bat>

```
@echo off
```

```
rem SQLWriter の再起動スクリプトです。  
sc start SQLWriter
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

- MsDtsServer110 <restart_msdtsserver110.bat>

```
@echo off
```

```
rem MsDtsServer110 の再起動スクリプトです。  
sc start MsDtsServer110
```

```
}
```

```
sc コマンドを使用してサービスの起動を行います。
```

```
exit
```

1-3. 再起動スクリプト (続き)

- SQL Server Distributed Replay Client <restart_distributedreplayclient.bat>

```
@echo off
```

```
rem SQL Server Distributed Replay Client の再起動スクリプトです。 }  
sc start "SQL Server Distributed Replay Client"  
exit
```

sc コマンドを使用して
サービスの起動を行います。

- SQL Server Distributed Replay Controller <restart_distributedreplaycontroller.bat>

```
@echo off
```

```
rem SQL Server Distributed Replay Controller の再起動スクリプトです。 }  
sc start "SQL Server Distributed Replay Controller"  
exit
```

sc コマンドを使用して
サービスの起動を行います。

2. インスタンス

2-1. 概要

- MSSQLSERVER
- SQLSERVERAGENT

タイプ		Service	Service
表示名	既定インスタンスの場合	SQL Server (MSSQLSERVER)	SQL Server エージェント (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Server (【インスタンス名】)	SQL Server エージェント (【インスタンス名】)
サービス名	既定インスタンスの場合	MSSQLSERVER	SQLSERVERAGENT
	名前つきインスタンスの場合	MSSQL\$【インスタンス名】	SQLAgent\$【インスタンス名】
常駐プロセス		sqlserver.exe	SQLAGENT.EXE
スタートアップ		自動	手動
回復機能		何もしない	何もしない
依存するサービス		なし	SQL Server (【インスタンス名】)
依存されるサービス		SQL Server エージェント (【インスタンス名】)	なし

20. (続き)

2-1. (続き)

- MSSQLFDLauncher
- MSSQLServerOLAPService

タイプ		Service	Service
表示名	既定インスタンスの場合	SQL Full-text Filter Daemon Launcher (MSSQLSERVER)	SQL Server Analysis Services (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Full-text Filter Daemon Launcher (【インスタンス名】)	SQL Server Analysis Services (【インスタンス名】)
サービス名	既定インスタンスの場合	MSSQLFDLauncher	MSSQLServerOLAPService
	名前つきインスタンスの場合	MSSQLFDLauncher\$【インスタンス名】	MSOLAP\$【インスタンス名】
常駐プロセス		fdlauncher.exe fdhost.exe	msmdsrv.exe
スタートアップ		手動	自動
回復機能		何もしない	何もしない
依存関係		なし	なし

* MSSQLFDLauncher は MSSQLSERVER 起動時に起動されます。

20. (続き)

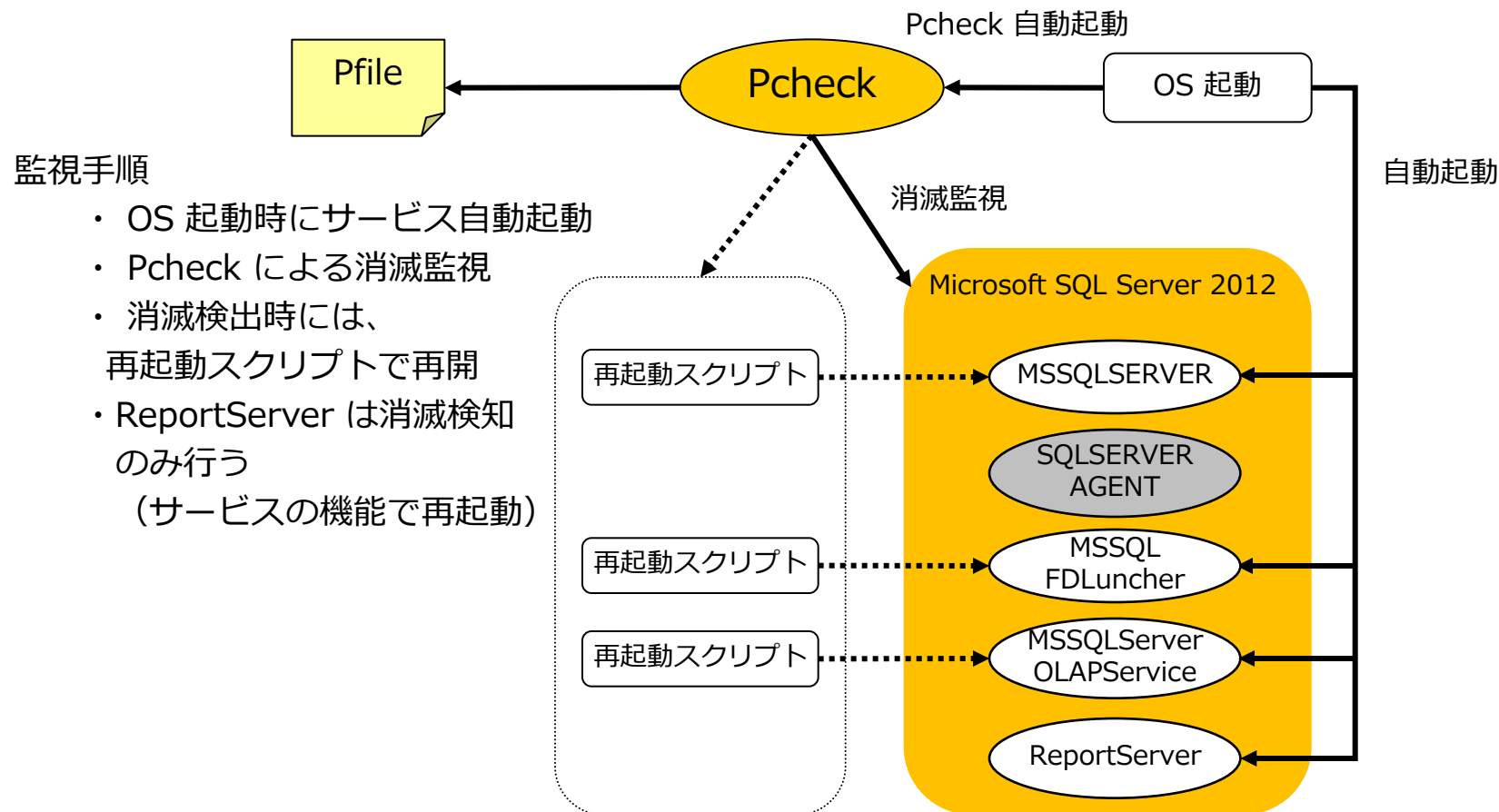
2-1. (続き)

- ReportServer

タイプ	Service	
表示名	既定インスタンスの場合	SQL Server Reporting Services (MSSQLSERVER)
	名前つきインスタンスの場合	SQL Server Reporting Services (【インスタンス名】)
サービス名	既定インスタンスの場合	ReportServer
	名前つきインスタンスの場合	ReportServer\$【インスタンス名】
常駐プロセス	ReportingServicesService.exe	
スタートアップ	自動	
回復機能	サービスを再起動する	
依存関係	なし	

20. (続き)

SQLSERVERAGENT 未使用時



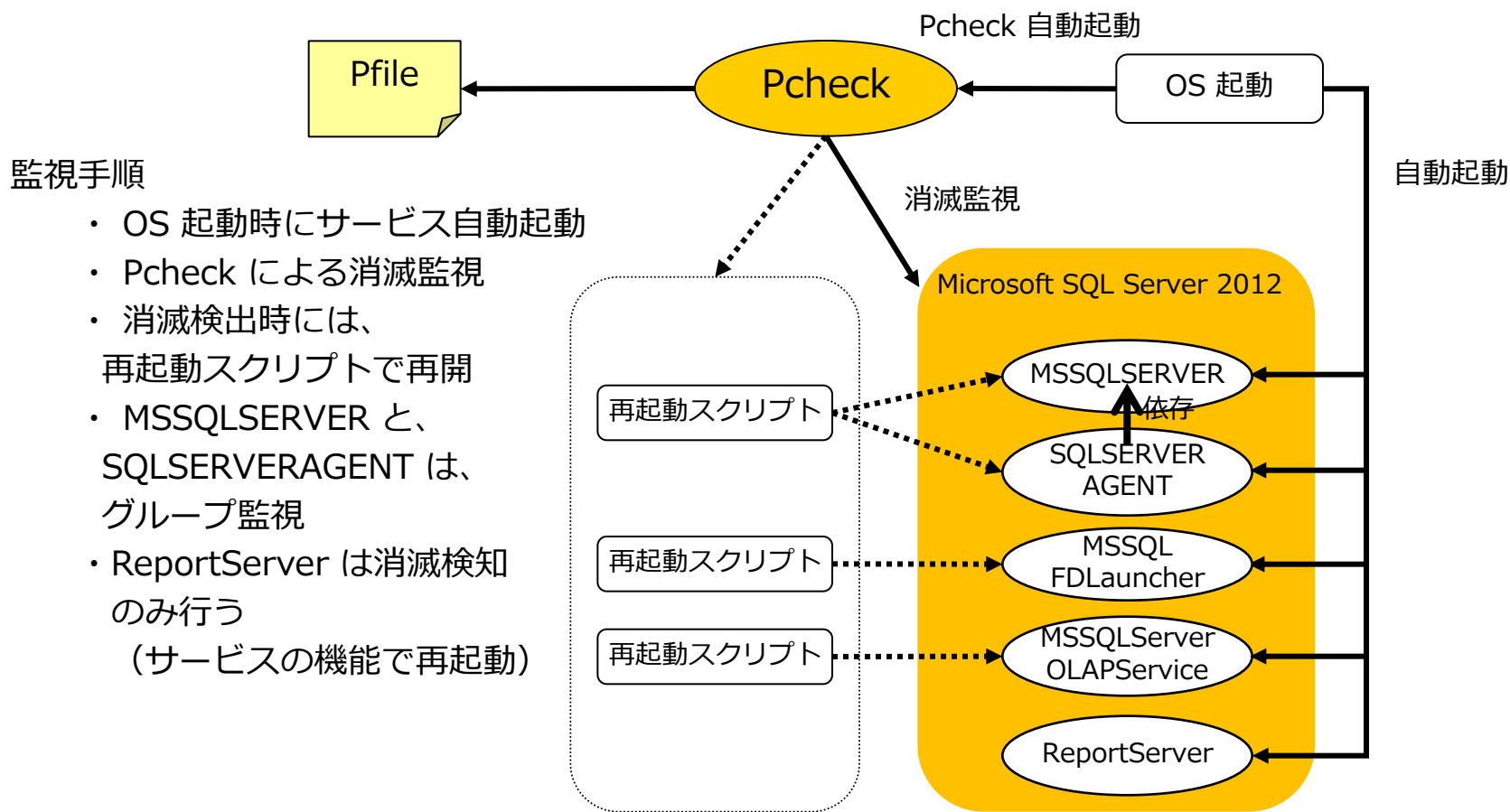
20. (続き)

SQLSERVERAGENT 使用時

SQLSERVERAGENT を使用している場合、SQLSERVERAGENT は MSSQLSERVER に依存するため、MSSQLSERVER 異常終了時に SQLSERVERAGENT も停止します。

ただし、SQLSERVERAGENT 異常終了時には、MSSQLSERVER は停止しません。

ProcessSaver にて各サービスを監視し、停止したサービスの再起動を行います。



20. (続き)

2-2. Pfile

Pfile <sample_sqlserver2012_instance_notuse_agent>

本 Pfile は SQLSERVERAGENT を使用しない場合の Pfile です。
本 Pfile は使用する機能により、監視するサービスを削除する必要があります。
本 Pfile は既定のインスタンスで作成されたサービスを監視しています。
本 Pfile は ReportServer の監視はコメントアウトしていますので ReportServer を監視するためには Pfile を変更する必要があります。
その場合、「3. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName           "C:\Program Files\HA\ProcessSaver\log\sqlserver2012_instance_notuse_agent.dmp"
UpMessageReduceMode   disable

##### PENT #####
##Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd
;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_sqlserver.bat",86400,3,Continue,Service
MSSQLFDLauncher,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_olapservice.bat",86400,3,Continue,Service
#ReportServer,,86400,0,Continue,Service
```

ReportServer サービスの自動再起動は、サービス異常終了後、デフォルトで 1分後に再起動されるため、本テンプレートの ReportServer の監視をコメントインして監視する場合、ProcessSaver で消滅検知後すぐに RetryOver となります。ReportServer サービスを監視する場合、消滅検知後、すぐに RetryOver にしたくない場合は、ReportServer サービスの [プロパティ] から再起動時間を変更していただくか、ReportServer サービス再起動を待ち合わせる再起動スクリプトをご用意いただく、または、MonitorInterval の値を変更する必要があります。
RetryOver になった場合は、サービス情報を再読み込み後にサービス監視の再開を行うために ReportServer サービスが 1 分後に起動したことを確認してから、Restart ボタンを押してください。

20. (続き)

2-2. (続き)

Pfile <sample_sqlserver2012_instance_use_agent>

本 Pfile は SQLSERVERAGENT を使用する場合の Pfile です。

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

本 Pfile は既定のインスタンスで作成されたサービスを監視しています。

本 Pfile は ReportServer の監視はコメントアウトしていますので ReportServer を監視するためには Pfile を変更する必要があります。

その場合、「3. Pfile 書き換え」の手順で Pfile を変更してください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:\Program Files\HA\ProcessSaver\log\sqlserver2012_instance_use_agent.dmp"
UpMessageReduceMode   disable

##### PENT #####
##Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd
;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:\Program
Files\HA\ProcessSaver\scriptfile\restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
SQLSERVERAGENT,"C:\Program
Files\HA\ProcessSaver\scriptfile\restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
MSSQLFDLauncher,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_olapservice.bat",86400,3,Continue,Service
#ReportServer,,86400,0,Continue,Service
```

ReportServer サービスの自動再起動は、サービス異常終了後、デフォルトで 1分後に再起動されるため、本テンプレートの ReportServer の監視をコメントインして監視する場合、ProcessSaver で消滅検知後すぐに RetryOver となります。ReportServer サービスを監視する場合で、消滅検知後、すぐに RetryOver にしたくない場合は、ReportServer サービスの [プロパティ] から再起動時間を変更していただくか、ReportServer サービス再起動を待ち合わせる再起動スクリプトをご用意いただく、または、MonitorInterval の値を変更する必要があります。RetryOver になった場合は、サービス情報を再読み込み後にサービス監視の再開を行うために ReportServer サービスが 1 分後に起動したことを確認してから、Restart ボタンを押してください。

20. (続き)

2-2. (続き)

名前付きインスタンスで作成されたサービスを監視する場合、使用する Pfile を以下のように変更してください。

```
##### PARAM #####
MsgCheckInterval 5
MonitorInterval 10
DumpFileName "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver2012_instance_use_agent.dmp"
UpMessageReduceMode disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
MSSQLSERVER,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
SQLSERVERAGENT,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlserver_useagent.bat",86400,3,Continue,Service;GroupTag=sqlserver2008r2group
MSSQLFDLauncher,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_fdlauncher.bat",86400,3,Continue,Service
MSSQLServerOLAPService,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_olapservice.bat",86400,3,Continue,Service
#ReportServer,,86400,0,Continue,Service
```

名前つきインスタンスのサービスを監視する場合は
各行の Process/ServiceName をそれぞれ以下のように変更してください。

MSSQLSERVER	→	MSSQL\$【インスタンス名】
SQLSERVERAGENT	→	SQLAgent\$【インスタンス名】
MSSQLFDLauncher	→	MSSQLFDLauncher\$【インスタンス名】
MSSQLServerOLAPService	→	MSOLAP\$【インスタンス名】
ReportServer	→	ReportServer\$【インスタンス名】

20. (続き)

2-3. 再起動スクリプト

本 スクリプトは既定のインスタンスで作成されたサービスを起動します。

名前つきインスタンスで作成されたサービスを起動するにはスクリプトを変更する必要があります。

- MSSQLSERVER <restart_sqlserver.bat>
本スクリプトは Pfile <sample_sqlserver2012_instance_notuse_agent> で使用します。

```
@echo off  
  
rem MSSQLSERVER の再起動スクリプトです。 }  
sc start MSSQLSERVER  
  
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は
MSSQL\$【インスタンス名】 に変更してください。

20. (続き)

2-3. (続き)

- MSSQLSERVER <restart_sqlserver_useagent.bat>
本スクリプトは Pfile <sample_sqlserver2012_instance_use_agent> で使用します。

```
@echo off

rem SQLSERVERAGENT 使用時の MSSQLSERVER の再起動スクリプトです。

rem MSSQLSERVER が起動していない場合、MSSQLSERVER と SQLSERVERAGENT を起動する。
sc query MSSQLSERVER | find "STOPPED"
if %ERRORLEVEL% == 0 (
  sc start MSSQLSERVER
)

rem SQLSERVERAGENT が起動していない場合、SQLSERVERAGENT を起動する。
sc query SQLSERVERAGENT | find "STOPPED"
if %ERRORLEVEL% == 0 (
  sc start SQLSERVERAGENT
)

exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は、以下のように変更してください。

MSSQLSERVER	→	MSSQL\$【インスタンス名】
SQLSERVERAGENT	→	SQLAgent\$【インスタンス名】

20. (続き)

2-3. (続き)

- MSSQLFDLauncher <restart_fdlauncher.bat>

```
@echo off
rem MSSQLFDLauncher の再起動スクリプトです。 }
sc start MSSQLFDLauncher
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は
MSSQLFDLauncher\$【インスタンス名】に変更してください。

- MSSQLServerOLAPService<restart_olapervice.bat>

```
@echo off
rem MSSQLServerOLAPService の再起動スクリプトです。 }
sc start MSSQLServerOLAPService
exit
```

sc コマンドを使用してサービスの起動を行います。

名前つきインスタンスの場合は
MSOLAP\$【インスタンス名】に変更してください。

3. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- ・ 監視しないサービスについては、Pfile から削除してください。
- ・ SQLBrowser、SQL Server Distributed Replay Client、SQL Server Distributed Replay Controller ReportServer サービスの監視については、コメントアウトしています。SQLBrowser、SQL Server Distributed Replay Client、SQL Server Distributed Replay Controller ReportServer サービスを使用、監視する場合は、Pfile を書き換えて、コメントインしてください。
- ・ SQLSERVERAGENT サービスを使用する場合、SQLSERVERAGENT サービスが MSSQLSERVER サービスに依存します。ProcessSaver による監視が不要な場合は Pfile から削除してください。

以降、システム一意のサービスの監視を削除する場合の Pfile の書き換え手順を説明します。

インスタンスのサービスの監視を削除する場合は、使用する Pfile

```
<C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver2012_instance_notuse_agent>
```

または

```
<C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver2012_instance_use_agent>
```

を「3-2. Pfile の読み込み」以降の手順にしたがい編集してください。

20. (続き)

Pfile 書き換え手順

3-1. Pfile の直接編集



3-2. Pfile の読み込み



3-3. 監視しないサービスの削除



3-4. Pfile の保存

20. (続き)

3-1. Pfile の直接編集

- (1) SQLBrowser、SQL Server Distributed Replay Client、SQL Server Distributed Replay Controllerを監視する場合、テキストエディターなどで Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver2012_system> を開き、SQLBrowser、SQL Server Distributed Replay Client、SQL Server Distributed Replay Controller の行の # を削除し、コメントインしてください。

```
##### PARAM #####
MsgCheckInterval      5
MonitorInterval       10
DumpFileName          "C:¥Program Files¥HA¥ProcessSaver¥log¥sqlserver2012_system.dmp"
UpMessageReduceMode  disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
# SQLBrowser,,86400,0,Continue,Service
SQLWriter,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_sqlwriter.bat",86400,3,Continue,Service
MsDtsServer110,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_msdtsserver110.bat",86400,3,Continue,Service
# SQL Server Distributed Replay Client,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_distributedreplayclient.bat",86400,3,Continue,Service
# SQL Server Distributed Replay Controller,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_distributedreplaycontroller.bat",86400,3,Continue,Service
```

を削除してコメントインしてください。

SQLBrowser サービスの自動再起動は、サービス異常終了後、デフォルトで 1分後に再起動されるため、本テンプレートの SQLBrowser の行をコメントインして監視する場合、ProcessSaver で消滅検知後すぐに RetryOver となります。SQLBrowser を監視する場合で、消滅検知後すぐに RetryOver にしたくない場合は、SQLBrowser サービスの [プロパティ] から再起動時間を変更していただくか、SQLBrowser サービス再起動を待ち合わせる再起動スクリプトをご用意いただく、または、MonitorInterval の値を変更する必要があります。

RetryOver になった場合は、サービス情報を再読み込み後にサービス監視の再開を行うために SQLBrowser サービスが 1分後に起動したことを確認してから、Restart ボタンを押してください。

3-2. Pfile の読み込み

(1) 『Create Pfile』で *Read* を押して

Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_sqlserver2012_system>
の読み込みを行ってください。

下記画面にならなかった場合、または Pfile の読み込み時にエラーか警告が出力された場合は、
Page 18 の手順により Pfile を再取得し、もう一度やり直してください。

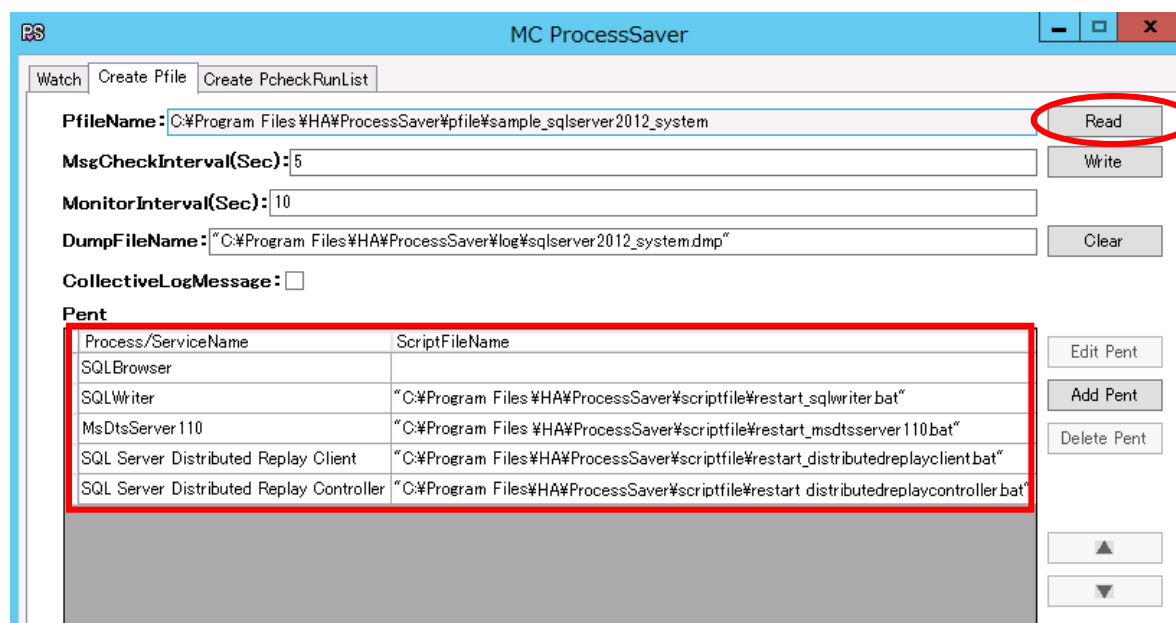


図21 『Create Pfile』

3-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から 監視しないサービスを選択し、*Delete Pent* を押して削除してください。

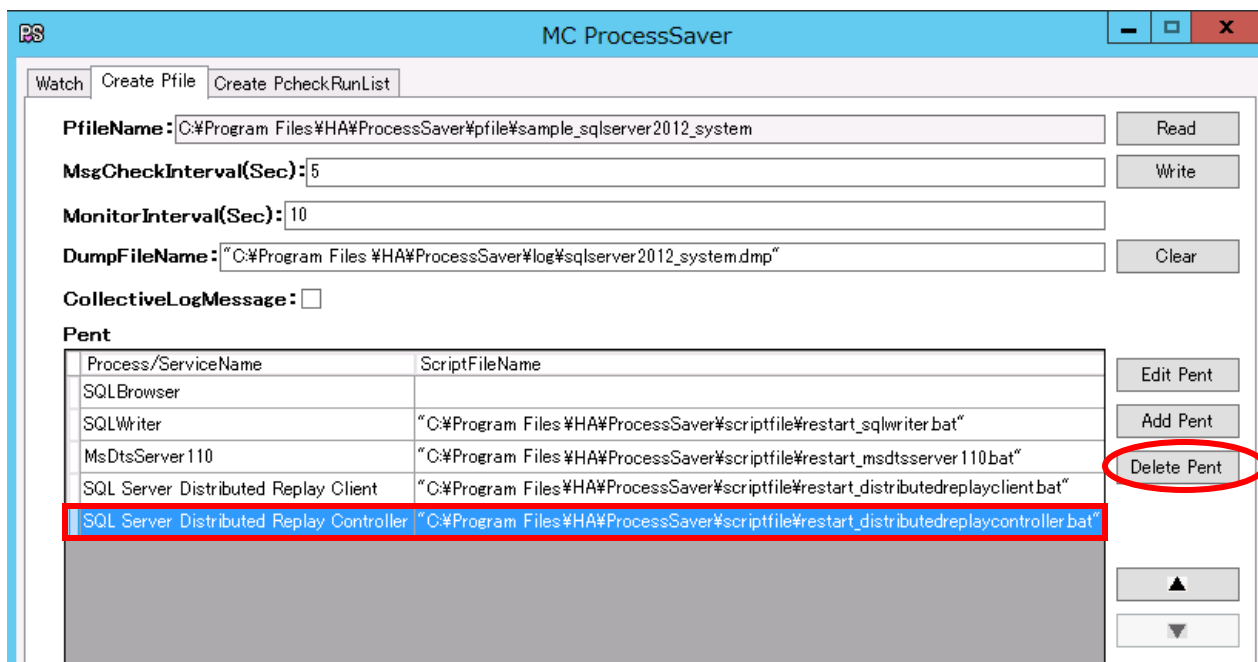


図22 『Create Pfile』

3-4. Pfile の保存

(1) 『Create Pfile』 で Write を押して

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_sqlserver2012_system>
を保存してください。

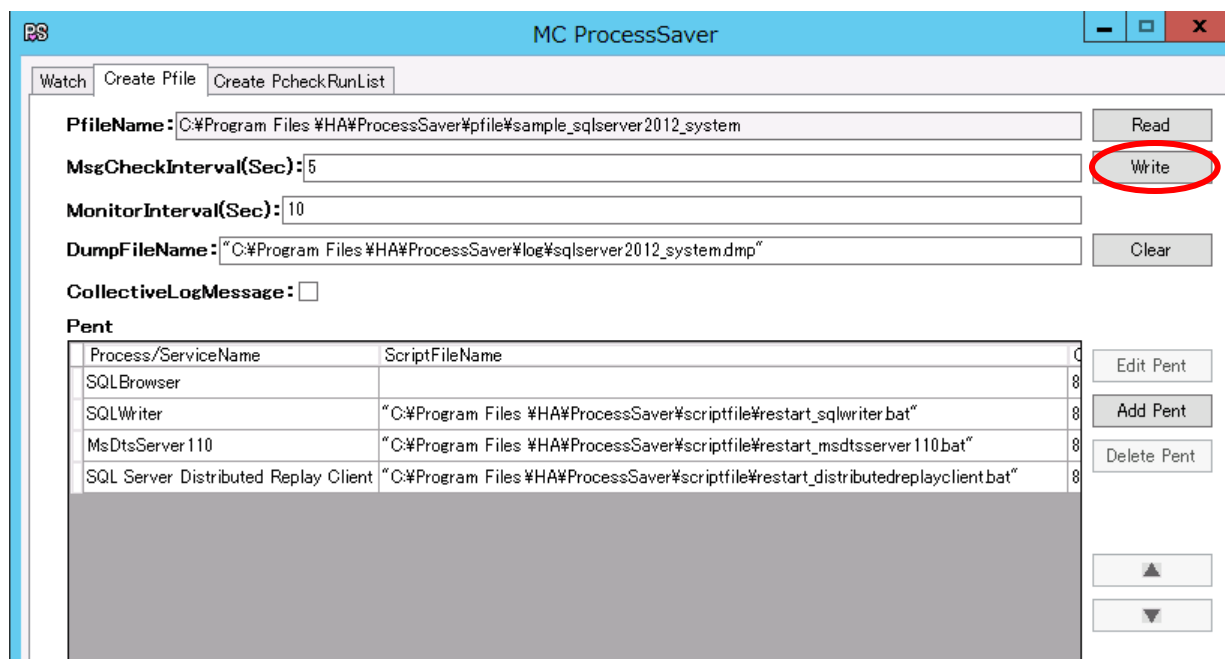


図23 『Create Pfile』

21. StarOffice の監視事例

1. 概要

本事例は StarOffice X Standard V4.0 を使用しています。
StarOffice X にて提供されている以下のサービスを監視する場合を紹介します。

表示名	サービス名
EDS Manager	EDS Manager
EDS Protocol Server	EDS Protocol Server
PERCIO-CCM Service	PERCIO-CCM service
SO-X Cabinet Server	ObjectManager
SO-X Mail LOG Service	mlog
SO-X Mail CGI Service	cgid
SO-X Mail INET Service	minetd
SO-X Mail SHELL Service	mshd
SO-X Mail Agent Queue Service	somail_agd
SO-X ScheduleServer	StarOffice FormServer
SO-X Presence Server Management Service	PMManager
SO-X Presence Server Distribution Service	PMDistributor
SO-X FTS BcabCrawler	SOXFTSBcabCrawler
SO-X FTS Maintenance	SOXFTSMaintenance
SO-X FTS Server	SOXFTSServer
SO-X FTS Broker	SOXFTSBroke
SO-X BizDirSync Service	so_edsyncd
SO-X BizDirSync Slave Service	BDSyncSlave_Service

21. (続き)

2. Pfile

Pfile <sample_staroffice>

本 Pfile は すべてのサービスを監視する設定となっています。

本 Pfile は使用する機能により、監視するサービスを削除する必要があります。

```
##### PARAM #####
MsgCheckInterval    5
MonitorInterval     10
DumpFileName        "C:¥Program Files¥HA¥ProcessSaver¥log¥staroffice_dump"
UpMessageReduceMode  disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch

EDS Manager,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_EDSManager.bat",86400,3,Continue,Service
EDS Protocol Server,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_EDSProtocol.bat",86400,3,Continue,Service
PERCIO-CCM service,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_PERCIO.bat",86400,3,Continue,Service
ObjectManager,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_Cabinet.bat",86400,3,Continue,Service
mlog,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailLOG.bat",86400,3,Continue,Service
cgid,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailCGI.bat",86400,3,Continue,Service
minetd,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailINET_Service.bat",86400,3,Continue,Service
```

次ページへ続く

21. (続き)

続き

```
mshd,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_MailSHELL.bat",86400,3,Continue,Service
somail_agd,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_MailAgentQueue.bat",86400,3,Continue,Service
StarOffice_FormServer,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_Schedule.bat",86400,3,Continue,Service
PMManager,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_PSMS.bat",86400,3,Continue,Service
PMDistributor,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_PSDS.bat",86400,3,Continue,Service
SOXFTSBcabCrawler,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_FTSBC.bat",86400,3,Continue,Service
SOXFTSMaintenance,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_FTSMaintenance.bat",86400,3,Continue,Service
SOXFTSServer,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_FTSServer.bat",86400,3,Continue,Service
SOXFTSBroker,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_FTSBroker.bat",86400,3,Continue,Service
so_edsyncd,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_BDS.bat",86400,3,Continue,Service
BDSSyncSlave_Service,"C:\Program Files\HA\ProcessSaver\scriptfile\restart_SO_BDSSlave.bat",86400,3,Continue,Service
```

3. 再起動スクリプト

再起動スクリプト

- EDS Manager <restart_SO_EDSManager.bat>

```
@echo off  
  
rem EDS Manager の再起動スクリプトです。  
net start "EDS Manager"  
  
exit
```

- EDS Protocol Server <restart_SO_EDSProtocol.bat>

```
@echo off  
  
rem EDS Protocol Server の再起動スクリプトです。  
net start "EDS Protocol Server"  
  
exit
```

- PERCIO-CCM Service <restart_SO_PERCIO.bat>

```
@echo off  
  
rem PERCIO-CCM Service の再起動スクリプトです。  
net start "PERCIO-CCM Service"  
  
exit
```

21. (続き)

- ObjectManager (SO-X Cabinet Server) <restart_SO_Cabinet.bat>

```
@echo off  
  
rem SO-X Cabinet Server の再起動スクリプトです。  
net start "SO-X Cabinet Server"  
  
exit
```

- mlog (SO-X Mail LOG Service) <restart_SO_MailLOG.bat>

```
@echo off  
  
rem SO-X Mail LOG Service の再起動スクリプトです。  
net start "SO-X Mail LOG Service"  
  
exit
```

- cgid (SO-X Mail CGI Service) <restart_SO_MailCGI.bat>

```
@echo off  
  
rem SO-X Mail CGI Service の再起動スクリプトです。  
net start "SO-X Mail CGI Service"  
  
exit
```

21. (続き)

- minetd (SO-X Mail INET Service) <restart_SO_MailINET_Service.bat>

```
@echo off  
  
rem SO-X Mail INET Service の再起動スクリプトです。  
net start "SO-X Mail INET Service"  
  
exit
```

- mshd (SO-X Mail SHELL Service) <restart_SO_MailSHELL.bat>

```
@echo off  
  
rem SO-X Mail SHELL Service の再起動スクリプトです。  
net start "SO-X Mail SHELL Service"  
  
exit
```

- somail_agd (SO-X Mail Agent Queue Service) <restart_SO_MailAgentQueue.bat>

```
@echo off  
  
rem SO-X Mail Agent Queue Service の再起動スクリプトです。  
net start "SO-X Mail Agent Queue Service"  
  
exit
```

21. (続き)

- StarOffice FormServer (SO-X ScheduleServer) <restart_SO_Schedule.bat>

```
@echo off  
  
rem SO-X ScheduleServer の再起動スクリプトです。  
net start "SO-X ScheduleServer"  
  
exit
```

- PMManager (SO-X Presence Server Management Service) <restart_SO_PSMS.bat>

```
@echo off  
  
rem SO-X Presence Server Management Service の再起動スクリプトです。  
net start "SO-X Presence Server Management Service"  
  
exit
```

- PMDistributor (SO-X Presence Server Distribution Service) <restart_SO_PSDS.bat>

```
@echo off  
  
rem SO-X Presence Server Distribution Service の再起動スクリプトです。  
net start "SO-X Presence Server Distribution Service"  
  
exit
```

21. (続き)

- SOXFTSBcabCrawler (SO-X FTS BcabCrawler) <restart_SO_FTSSBC.bat>

```
@echo off  
  
rem SO-X FTS BcabCrawler の再起動スクリプトです。  
net start "SO-X FTS BcabCrawler"  
  
exit
```

- SOXFTSMaintenance (SO-X FTS Maintenance) <restart_SO_FTSMaintenance.bat>

```
@echo off  
  
rem SO-X FTS Maintenance の再起動スクリプトです。  
net start "SO-X FTS Maintenance"  
  
exit
```

- SOXFTSServer (SO-X FTS Server) <restart_SO_FTSServer.bat>

```
@echo off  
  
rem SO-X FTS Server の再起動スクリプトです。  
net start "SO-X FTS Server"  
  
exit
```

21. (続き)

- SOXFTSBroker (SO-X FTS Broker) <restart_SO_FTSBroker.bat>

```
@echo off  
  
rem SO-X FTS Broker の再起動スクリプトです。  
net start "SO-X FTS Broker"  
  
exit
```

- so_edsyncd (SO-X BizDirSync Service) <restart_SO_BDS.bat>

```
@echo off  
  
rem SO-X BizDirSync Service の再起動スクリプトです。  
net start "SO-X BizDirSync Service"  
  
exit
```

- BDSyncSlave_Service (SO-X BizDirSync Slave Service) <restart_SO_BDSSlave.bat>

```
@echo off  
  
rem SO-X BizDirSync Slave Service の再起動スクリプトです。  
net start "SO-X BizDirSync Slave Service"  
  
exit
```

4. Pfile 書き換え

提供している Pfile を下記の観点で書き換えてください。

- ・ 監視しないサービスについては、Pfile から削除してください。

Pfile 書き換え手順

4-1. Pfile の直接編集



4-2. Pfile の読み込み



4-3. 監視しないサービスの削除



4-4. Pfile の保存

21. (続き)

4-1. Pfile の直接編集

- (1) テキストエディターなどで Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_staroffice> を開き、編集してください。

```
.
.
##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearCmd;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch

EDS Manager,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_EDSManager.bat",86400,3,Continue,Service
EDS Protocol Server,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_EDSProtocol.bat",86400,3,Continue,Service
PERCIO-CCM service,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_PERCIO.bat",86400,3,Continue,Service
ObjectManager,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_Cabinet.bat",86400,3,Continue,Service
mlog,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailLOG.bat",86400,3,Continue,Service
cgid,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailCGI.bat",86400,3,Continue,Service
minetd,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailINET_Service.bat",86400,3,Continue,Service
mshd,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailSHELL.bat",86400,3,Continue,Service
somain_agd,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_MailAgentQueue.bat",86400,3,Continue,Service
StarOffice FormServer,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_Schedule.bat",86400,3,Continue,Service
PMManager,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_PSMS.bat",86400,3,Continue,Service
PMDistributor,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_PSDS.bat",86400,3,Continue,Service
SOXFTSBcabCrawler,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_FTSBC.bat",86400,3,Continue,Service
SOXFTSMaintenance,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_FTSMaintenance.bat",86400,3,Continue,Service
SOXFTSServer,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_FTSServer.bat",86400,3,Continue,Service
SOXFTSBroker,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_FTSBroker.bat",86400,3,Continue,Service
so_edsyncd,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_BDS.bat",86400,3,Continue,Service
BDSyncSlave_Service,"C:¥Program Files¥HA¥ProcessSaver¥scriptfile¥restart_SO_BDSSlave.bat",86400,3,Continue,Service
```

4-2. Pfile の読み込み

(1) 『Create Pfile』で Read を押して

Pfile <C:\Program Files\HA\ProcessSaver\pfile\sample_staroffice> の読み込みを行ってください。

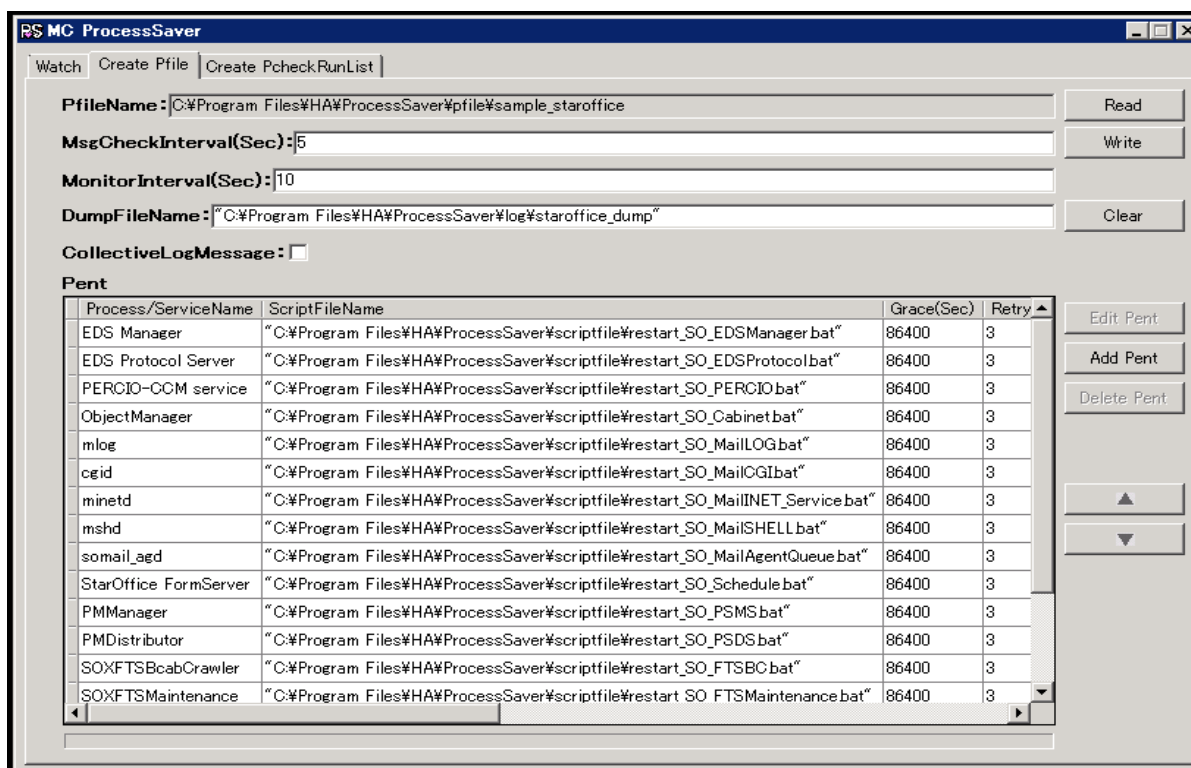


図24 『Create Pfile』

4-3. 監視しないサービスの削除

- (1) 『Create Pfile』の [Pent] から監視しないサービスを選択し、*Delete Pent* を押して削除してください。

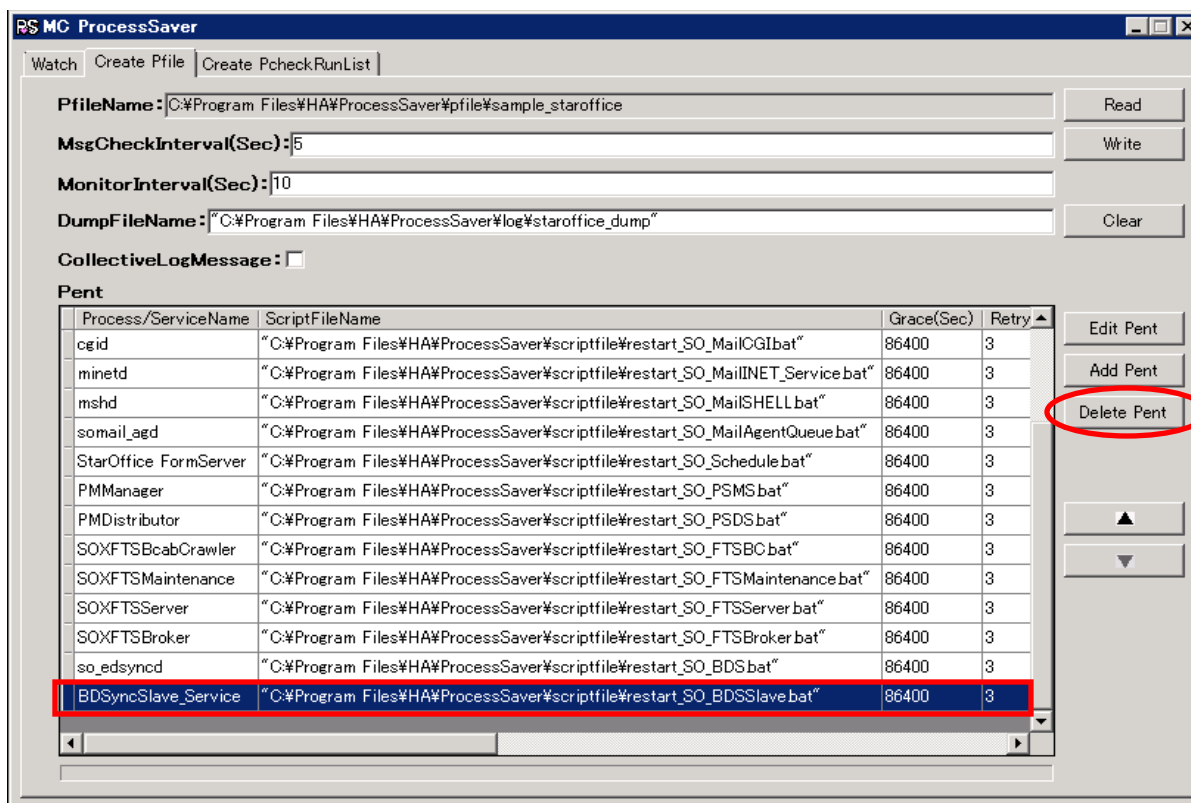


図25 『Create Pfile』

4-4. Pfile の保存

(1) 『Create Pfile』で *Write* を押して

Pfile <C:¥Program Files¥HA¥ProcessSaver¥pfile¥sample_staroffice> を保存してください。

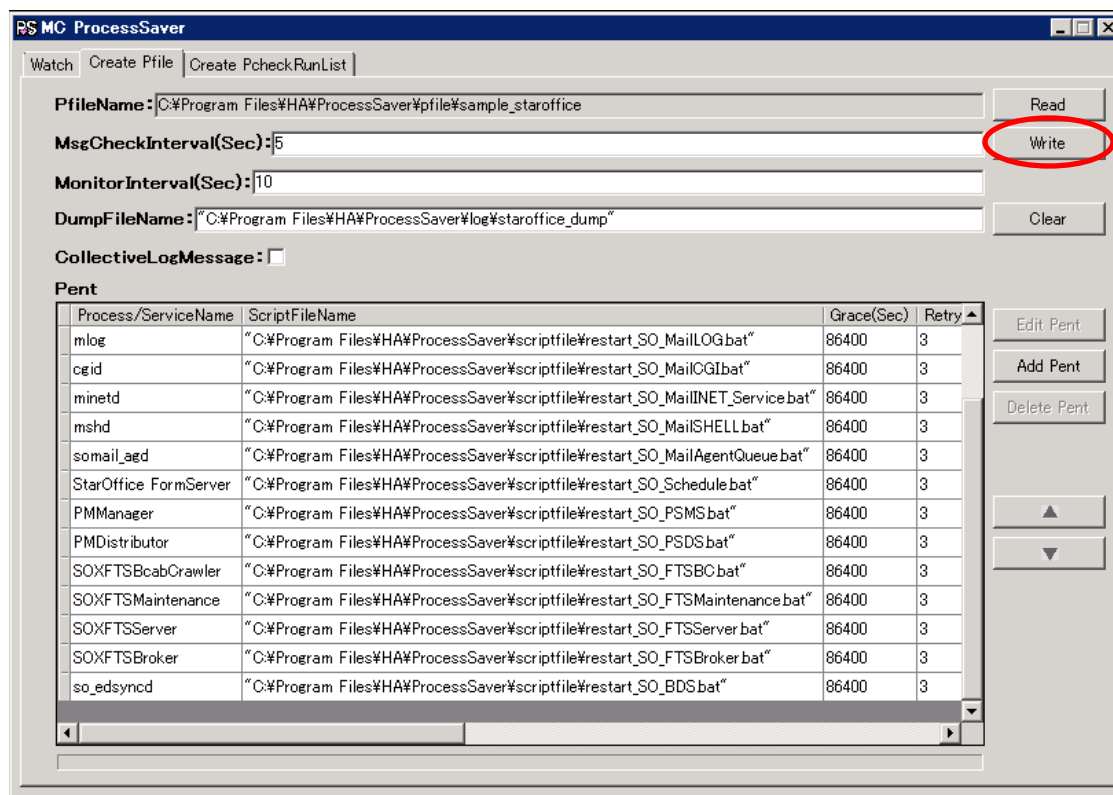


図26 『Create Pfile』

22. IIS のストール監視事例

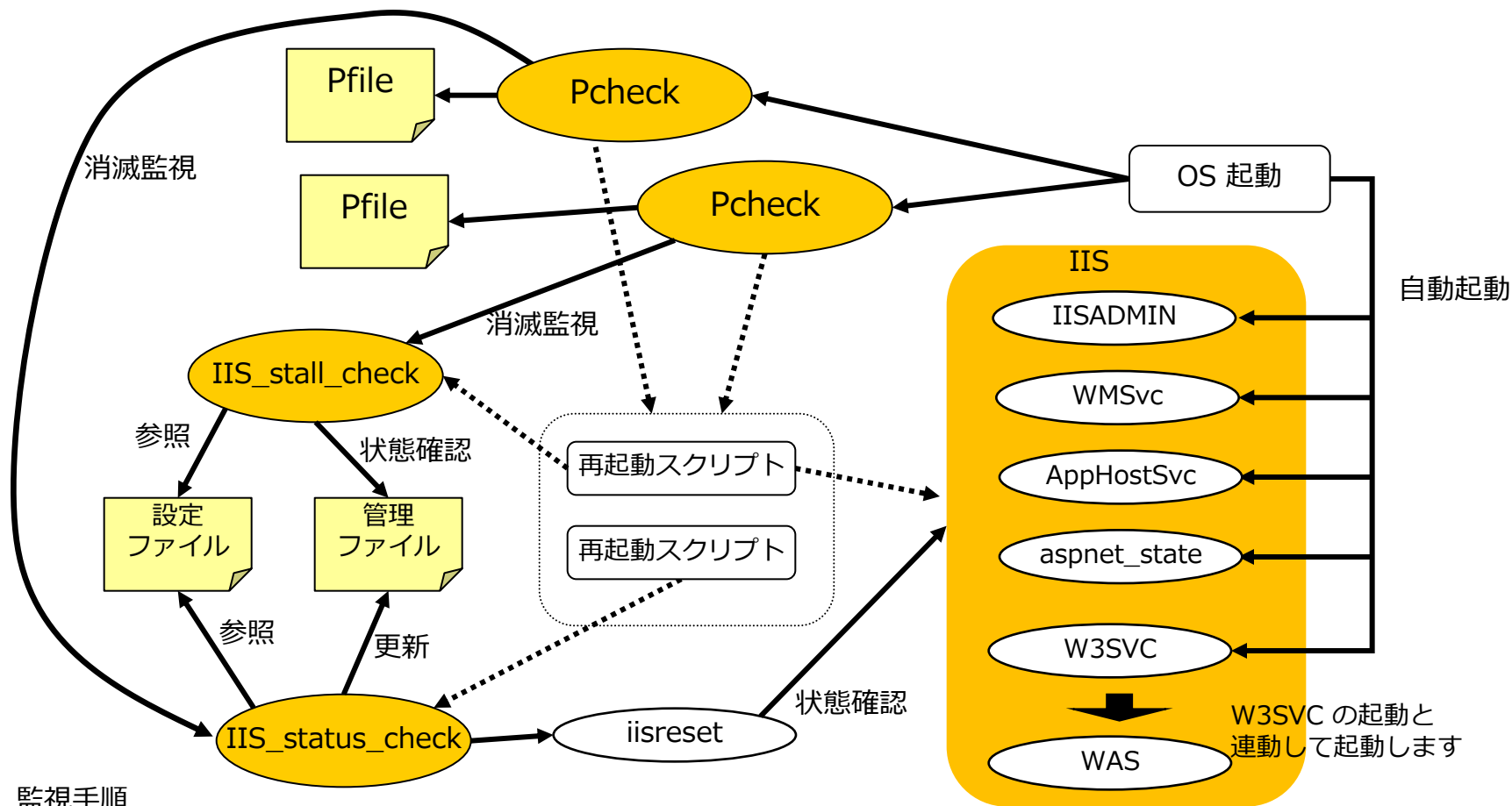
1. 概要

IIS のストール監視を以下の方式で行います。

IIS のステータス参照コマンド (iisreset.exe) を用いて、定期的に IIS の運用状態を確認し、管理ファイルを更新するスクリプト(IIS_status_check.ps1)と管理ファイルの更新を定期的に監視し、一定時間更新されていない場合に終了するスクリプト(IIS_stall_check.ps1)を起動します。

ProcessSaver は IIS_stall_check.ps1 の消滅を監視することでストール監視を行います。また、IIS_status_check.ps1 が何らかの原因により消滅した場合に、誤ってストールと判断しないよう IIS_status_check.ps1の消滅も監視します。

22. (続き)



2. Pfile

Pfile <sample_pfile_IIS_stall>

本 Pfile は IIS_stall_check.ps1 を監視する Pfile です。

```
##### PARAM #####
MsgCheckInterval    5
MonitorInterval     10
DumpFileName        "C:¥Program Files¥HA¥ProcessSaver¥log¥pfile_IIS_stall_dump"
UpMessageReduceMode  disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearC
md;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
powershell.exe,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS_stall_check.bat",86400,3,Continue,Process;IncludeStrings="C:¥Program
Files¥HA¥ProcessSaver¥bin¥IIS_stall_check.ps1";PnameFullMatch=disable
```

22. (続き)

Pfile <sample_pfile_IIS_script>

本 Pfile は IIS_status_check.ps1 を監視する Pfile です。

```
##### PARAM #####
MsgCheckInterval    5
MonitorInterval     10
DumpFileName        "C:¥Program Files¥HA¥ProcessSaver¥log¥pfile_IIS_script_dump"
UpMessageReduceMode disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearC
md;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
powershell.exe,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_IIS_status_check.bat",1,1,Continue,Process;IncludeStrings="C:¥Program
Files¥HA¥ProcessSaver¥bin¥IIS_status_check.ps1";PnameFullMatch=disable
```

3. 再起動スクリプト

再起動スクリプト

- IIS_stall_check.ps1 <restart_IIS_stall_check.bat>
 - 本スクリプトは IIS_stall_check.ps1 の消滅を検知した場合に実行します。
 - INSTALLPATH には ProcessSaver のインストールフォルダーを設定してください。
デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。
 - PFILE_IIS には IIS のサービス死活監視の設定ファイルを設定してください。
監視しない場合、設定変更は不要です。
 - PFILE_IIS_SCRIPT には IIS_status_check.ps1 の監視設定ファイルを設定してください。

```
@echo off
```

```
rem ProcessSaver のインストールフォルダーを設定する
```

```
set INSTALLPATH=C:¥Program Files¥HA¥ProcessSaver
```

```
set PFILE_IIS=%INSTALLPATH%¥pfile¥pfile_IIS
```

```
set PFILE_IIS_SCRIPT=%INSTALLPATH%¥pfile¥sample_pfile_IIS_script
```

ファイル名に合わせて
設定を変更します。

```
set PADMIN=%INSTALLPATH%¥bin¥Padmin.exe
```

```
set PSSLEEP=%INSTALLPATH%¥bin¥PSSleep.exe
```

```
set STALL_CHECK=%INSTALLPATH%¥bin¥IIS_stall_check.ps1
```

```
set STATUS_CHECK=%INSTALLPATH%¥bin¥IIS_status_check.ps1
```

```
次ページへ続く
```

22. (続き)

```
rem IIS のサービスを再起動するため IIS の死活監視を行っている場合
```

```
rem 再起動処理中は監視を停止する
```

```
:RETRY_STOP
```

```
"%PADMIN%" -f "%PFILE_IIS%" -c Stop
```

```
if "%ERRORLEVEL%" == "-1" (
```

```
    "%PSSLEEP%" 1
```

```
    goto RETRY_STOP
```

```
)
```

```
rem 念のため MsgCheckInterval秒待つ
```

```
"%PSSLEEP%" 5
```

```
rem IIS 関連サービスを再起動
```

```
iisreset /restart
```

```
net stop aspnet_state
```

```
net start aspnet_state
```

```
net stop AppHostSvc
```

```
net start AppHostSvc
```

```
net stop WMSVC
```

```
net start WMSVC
```

```
rem 停止した IIS の死活監視を再開する
```

```
rem 受付済み命令の実行待ちにより、再開に失敗する場合がありますので失敗時はリトライする
```

```
:RETRY_START
```

```
"%PADMIN%" -f "%PFILE_IIS%" -c Start
```

```
if "%ERRORLEVEL%" == "-1" (
```

```
    "%PSSLEEP%" 1
```

```
    goto RETRY_START
```

```
)
```

```
次ページへ続く
```

ProcessSaver で IIS サービスの
死活監視を行っていない場合は
コメントアウト or 削除します。

ProcessSaver で IIS サービスの
死活監視を行っていない場合は
コメントアウト or 削除します。

22. (続き)

```
rem IIS ステータス確認スクリプトを再起動するため、スクリプトの監視を一時停止
:RETRY_STOP_2
"%PADMIN%" -f "%PFILE_IIS_SCRIPT%" -c Stop
if "%ERRORLEVEL%" == "-1" (
    "%PSSLEEP%" 1
    goto RETRY_STOP_2
)

rem IIS ステータス確認スクリプトを再起動
FOR /F "usebackq tokens=2 delims=: " %%A IN (`powershell "Get-WmiObject -Class Win32_Process |
Where-Object { $_.CommandLine -like ¥"*IIS_status_check.ps1*¥" } | Sort-Object -Property CreationDate |
Select-Object ProcessID -First 1 | Format-List"`) DO SET PROCESSID=%%A
powershell "Stop-Process -Id %PROCESSID% -Force"

powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -
ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%STATUS_CHECK%¥"' -WindowStyle Hidden"

rem IIS stall 監視スクリプトを実行
powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -
ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%STALL_CHECK%¥"' -WindowStyle Hidden"

rem スクリプト起動と監視のすれ違いを防ぐため、少し待機
"%PSSLEEP%" 10

rem IIS ステータス確認スクリプトの監視を再開
:RETRY_RESTART
"%PADMIN%" -f "%PFILE_IIS_SCRIPT%" -c Restart
if "%ERRORLEVEL%" == "-1" (
    "%PSSLEEP%" 1
    goto RETRY_RESTART
)

exit
```

22. (続き)

- IIS_status_check.ps1 <restart_IIS_status_check.bat>
 - 本スクリプトは IIS_status_check.ps1 の消滅を検知した場合に実行します。
 - INSTALLPATH には ProcessSaver のインストールフォルダーを設定してください。デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。

```
@echo off

rem ProcessSaver のインストールフォルダーを設定する
set INSTALLPATH=C:¥Program Files¥HA¥ProcessSaver

set STATUS_CHECK=%INSTALLPATH%¥bin¥IIS_status_check.ps1
set PSSLEEP=%INSTALLPATH%¥bin¥PSSleep.exe

rem IIS ステータス確認スクリプトを再起動(念のため、プロセス停止処理も実行)
FOR /F "usebackq tokens=2 delims=: " %%A IN (`powershell "Get-WmiObject -Class Win32_Process | Where-Object { $_.CommandLine -like ¥"*IIS_status_check.ps1*¥" } | Sort-Object -Property CreationDate | Select-Object ProcessID -First 1 | Format-List"`) DO SET PROCESSID=%%A
powershell "Stop-Process -Id %PROCESSID% -Force"

powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%STATUS_CHECK%¥"' -WindowStyle Hidden"

rem スクリプト起動と監視のすれ違いを防ぐため、少し待機
"%PSSLEEP%" 10

exit
```

4. ストール監視スクリプトの設定

- IIS_check.conf

```
# ファイルの更新をチェックする間隔
STALL_CHECK_INTERVAL 600

# ステータスをチェックしファイルを更新する間隔
# STALL_CHECK_INTERVAL より短い時間を設定すること
STATUS_CHECK_INTERVAL 10
```

-STALL_CHECK_INTERVAL

IIS_stall_check.ps1 が管理ファイルの更新を確認する間隔です。この時間、管理ファイルが更新されていない場合にストールしていると判断します。

デフォルトは 600秒です。

-STATUS_CHECK_INTERVAL

IIS_status_check.ps1 が IIS のステータスを確認し管理ファイルを更新する間隔です。

デフォルトは 10秒です。

22. (続き)

- IIS_stall_check.ps1

管理ファイルの更新を定期的に監視するスクリプトです。

- installPath には ProcessSaver のインストールフォルダーを指定してください。
デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。

```
Set-PSDebug -Strict
$errorActionPreference = "SilentlyContinue"

# 各ファイル名の設定
$installPath = "C:¥Program Files¥HA¥ProcessSaver"
$configFile = $installPath + "¥bin¥IIS_check.conf"
$updateFile = $installPath + "¥bin¥IIS_staus"
$logFile = $installPath + "¥log¥IIS_stall.log"
$rotateLogFile = $installPath + "¥log¥IIS_stall.log.save1"

$interval = 600 * 1000
$binFirst = $true

以下省略(詳細は実ファイルを確認してください)
```

22. (続き)

- IIS_status_check.ps1

定期的に IIS の運用状態を確認し、管理ファイルを更新するスクリプトです。

- installPath には ProcessSaver のインストールフォルダーを指定してください。

デフォルトは、C:¥Program Files¥HA¥ProcessSaver です。

```
Set-PSDebug -Strict
$errorActionPreference = "SilentlyContinue"

# 各ファイル名の設定
$installPath = "C:¥Program Files¥HA¥ProcessSaver"
$configFile = $installPath + "¥bin¥IIS_check.conf"
$updateFile = $installPath + "¥bin¥IIS_staus"
$logFile = $installPath + "¥log¥IIS_status.log"
$rotateLogFile = $installPath + "¥log¥IIS_status.log.save1"

$interval = 10 * 1000

以下省略(詳細は実ファイルを確認してください)
```

22. (続き)

- IIS_script_start.bat

ストール監視を開始するためのバッチファイルです。

OS 起動後、自動で監視を行いたい場合は、タスクスケジューラに本バッチを登録してください。

- INSTALLPATH には ProcessSaver のインストールフォルダーを設定してください。

デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。

```
@echo off
```

```
rem ProcessSaver のインストールフォルダーを設定する
```

```
set INSTALLPATH=C:¥Program Files¥HA¥ProcessSaver
```

```
set PSSLEEP=%INSTALLPATH%¥bin¥PSSleep.exe
```

```
REM ステータス確認スクリプト実行
```

```
powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -  
ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%INSTALLPATH%¥bin¥IIS_status_check.ps1¥"' -  
WindowStyle Hidden"
```

```
REM 念のため 10秒待機
```

```
"%PSSLEEP%" 10
```

```
REM ストール監視スクリプト実行
```

```
powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -  
ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%INSTALLPATH%¥bin¥IIS_stall_check.ps1¥"' -  
WindowStyle Hidden"
```

22. (続き)

- IIS_stall_check_stop.bat

IIS_stall_check.ps1 を停止するバッチファイルです。

```
@echo off

rem IIS stall 監視スクリプトを停止する(プロセス停止)
FOR /F "usebackq tokens=2 delims=: " %%A IN (` powershell "Get-WmiObject -Class Win32_Process |
Where-Object { $_.CommandLine -like ¥"*IIS_stall_check.ps1*¥" } | Sort-Object -Property
CreationDate | Select-Object ProcessID -First 1 | Format-List"`) DO SET PROCESSID=%%A
powershell "Stop-Process -Id %PROCESSID% -Force"
```

- IIS_status_check_stop.bat

IIS_status_check.ps1 を停止するバッチファイルです。

```
@echo off

rem IIS ステータス確認スクリプトを停止する(プロセス停止)
FOR /F "usebackq tokens=2 delims=: " %%A IN (` powershell "Get-WmiObject -Class Win32_Process |
Where-Object { $_.CommandLine -like ¥"*IIS_status_check.ps1*¥" } | Sort-Object -Property
CreationDate | Select-Object ProcessID -First 1 | Format-List"`) DO SET PROCESSID=%%A
powershell "Stop-Process -Id %PROCESSID% -Force"
```

23. Microsoft SQLServer のストール監視事例

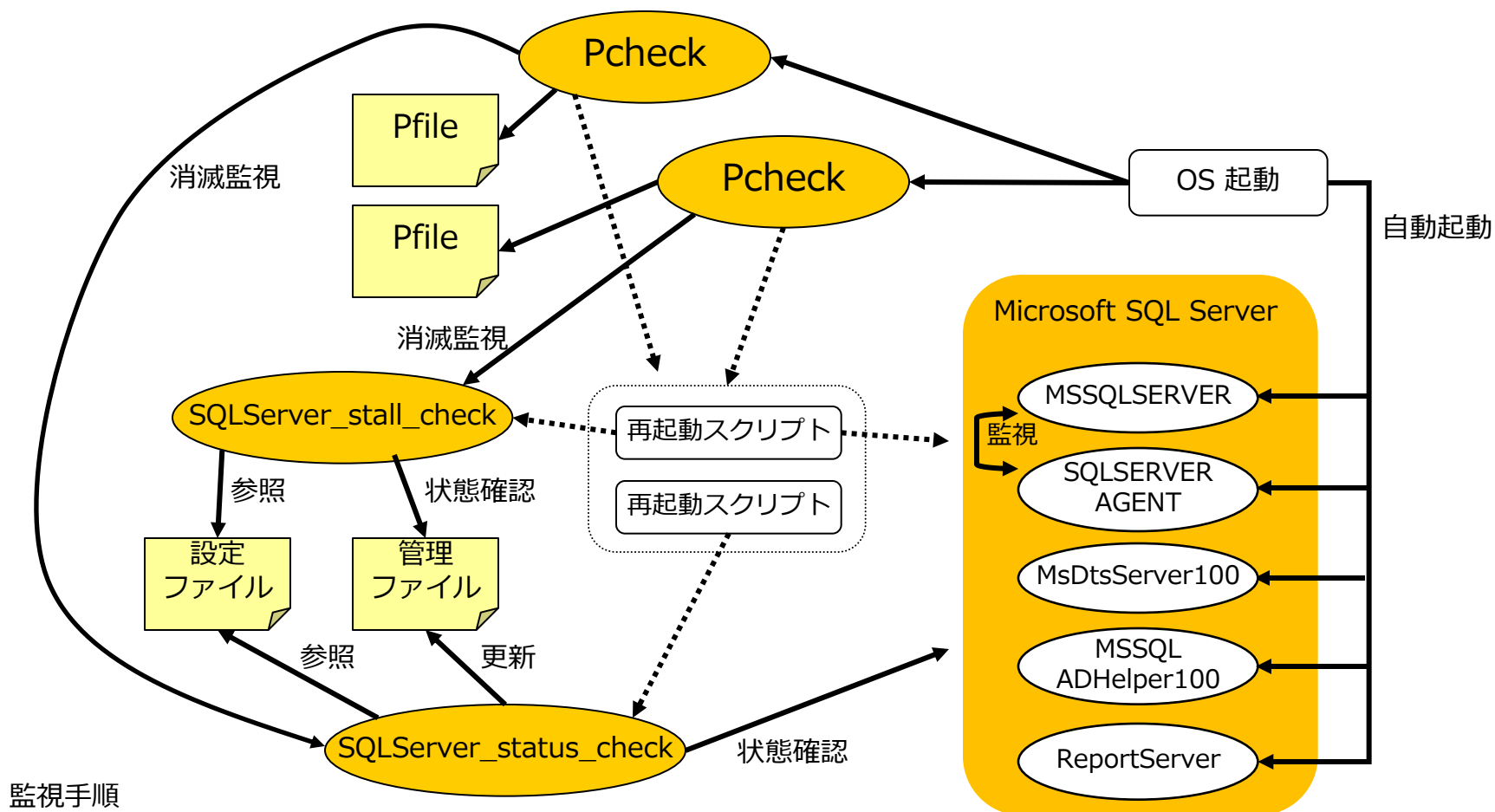
1. 概要

Microsoft SQL Server のストール監視を以下の方式で行います。

SQLServer の管理テーブル(sys.databases 表)を参照してデータベースの状態を定期的に確認し、管理ファイルを更新するスクリプト(SQLServer_status_check.ps1)と管理ファイルの更新を定期的に監視し、一定時間更新されていない場合に終了するスクリプト(SQLServer_stall_check.ps1)を起動します。

ProcessSaver は SQLServer_stall_check.ps1 の消滅を監視することでストール監視を行います。また、SQLServer_status_check.ps1 が何らかの原因により消滅した場合に、誤ってストールと判断しないよう SQLServer_status_check.ps1 の消滅も監視します。

23. (続き)



2. Pfile

Pfile <sample_pfile_SQLServer_stall>

本 Pfile は SQLServer_stall_check.ps1 を監視する Pfile です。

```
##### PARAM #####
MsgCheckInterval    5
MonitorInterval     10
DumpFileName        "C:¥Program Files¥HA¥ProcessSaver¥log¥pfile_SQLServer_stall_dump"
UpMessageReduceMode disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearC
md;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
powershell.exe,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_SQLServer_stall_check.bat",86400,3,Continue,Process;IncludeStrings="C:¥Program
Files¥HA¥ProcessSaver¥bin¥SQLServer_stall_check.ps1";PnameFullMatch=disable
```

23. (続き)

Pfile <sample_pfile_SQLServer_script>

本 Pfile は SQLServer_status_check.ps1 を監視する Pfile です。

```
##### PARAM #####
MsgCheckInterval    5
MonitorInterval     10
DumpFileName        "C:¥Program Files¥HA¥ProcessSaver¥log¥pfile_SQLServer_script_dump"
UpMessageReduceMode disable

##### PENT #####
##
Process/ServiceName,ScriptFileName,Grace(Sec),RetryCountMax,RetryOverAction,CheckType;MinProcCount;GroupTag;ClearC
md;IncludeStrings;RestartWaitTime;RestartTimeOut;PnameFullMatch
powershell.exe,"C:¥Program
Files¥HA¥ProcessSaver¥scriptfile¥restart_SQLServer_status_check.bat",1,1,Continue,Process;IncludeStrings="C:¥Program
Files¥HA¥ProcessSaver¥bin¥SQLServer_status_check.ps1";PnameFullMatch=disable
```

3. 再起動スクリプト

再起動スクリプト

- SQLServer_stall_check.ps1 <restart_SQLServer_stall_check.bat>
 - 本スクリプトは SQLServer_stall_check.ps1 の消滅を検知した場合に実行します。
 - INSTALLPATH には ProcessSaver のインストールフォルダーを設定してください。
デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。
 - PFILE_SQLSERVER には SQLServer のサービス死活監視の設定ファイルを設定してください。
監視しない場合、設定変更は不要です。
 - PFILE_SQLSERVER_SCRIPT には SQLServer_status_check.ps1 の監視設定ファイルを設定してください。

```
@echo off
```

```
rem ProcessSaver のインストールフォルダーを設定する  
set INSTALLPATH=C:¥Program Files¥HA¥ProcessSaver
```

```
set PADMIN=%INSTALLPATH%¥bin¥Padmin.exe  
set PSSLEEP=%INSTALLPATH%¥bin¥PSSleep.exe  
set PFILE_SQLSERVER=%INSTALLPATH%¥pfile¥pfile_SQLServer  
set PFILE_SQLSERVER_SCRIPT=%INSTALLPATH%¥pfile¥pfile_SQLServer_script  
set STALL_CHECK=%INSTALLPATH%¥bin¥SQLServer_stall_check.ps1  
set STATUS_CHECK=%INSTALLPATH%¥bin¥SQLServer_status_check.ps1  
次ページへ続く
```

ファイル名に合わせて
設定を変更します。

23. (続き)

```
rem SQLServer サービスを再起動するため SQLServer の死活監視を行っている場合、
```

```
rem 再起動処理中は監視を停止する
```

```
:RETRY_STOP  
"%PADMIN%" -f "%PFILE_SQLSERVER%" -c Stop  
if "%ERRORLEVEL%" == "-1" (  
    "%PSSLEEP%" 1  
    goto RETRY_STOP  
)
```

```
rem 念のため MsgCheckInterval秒待つ
```

```
"%PSSLEEP%" 5
```

```
rem SQLServer サービスを再起動
```

```
net stop MSSQLSERVER
```

```
net start MSSQLSERVER
```

```
rem 停止した SQLServer の死活監視を再開する
```

```
rem 受付済み命令の実行待ちにより、再開に失敗する場合がありますので失敗時はリトライする
```

```
:RETRY_START  
"%PADMIN%" -f "%PFILE_SQLSERVER%" -c Start  
if "%ERRORLEVEL%" == "-1" (  
    "%PSSLEEP%" 1  
    goto RETRY_START  
)
```

```
次ページへ続く
```

ProcessSaver で SQLServer の
死活監視を行っていない場合は
コメントアウト or 削除します。

ProcessSaver で SQLServer の
死活監視を行っていない場合は
コメントアウト or 削除します。

23. (続き)

```
rem SQLServer ステータス確認スクリプトを再起動するため、スクリプトの監視を一時停止
:RETRY_STOP_2
"%PADMIN%" -f "%PFILE_SQLSERVER_SCRIPT%" -c Stop
if "%ERRORLEVEL%" == "-1" (
    "%PSSLEEP%" 1
    goto RETRY_STOP_2
)

rem SQLServer ステータス確認スクリプトを再起動
FOR /F "usebackq tokens=2 delims=: " %%A IN (`powershell "Get-WmiObject -Class Win32_Process |
Where-Object { $_.CommandLine -like ¥"*SQLServer_status_check.ps1*¥" } | Sort-Object -Property
CreationDate | Select-Object ProcessID -First 1 | Format-List`) DO SET PROCESSID=%%A
powershell "Stop-Process -Id %PROCESSID% -Force"

powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -
ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%STATUS_CHECK%¥"' -WindowStyle Hidden"

rem SQLServer stall 監視スクリプトを実行
powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -
ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%STALL_CHECK%¥"' -WindowStyle Hidden"

rem スクリプト起動と監視のすれ違いを防ぐため、少し待機
"%PSSLEEP%" 10

rem SQLServer ステータス確認スクリプトの監視を再開
:RETRY_RESTART
"%PADMIN%" -f "%PFILE_SQLSERVER_SCRIPT%" -c Restart
if "%ERRORLEVEL%" == "-1" (
    "%PSSLEEP%" 1
    goto RETRY_RESTART
)

exit
```

23. (続き)

- SQLServer_status_check.ps1 <restart_SQLServer_status_check.bat>
 - 本スクリプトは SQLServer_status_check.ps1 の消滅を検知した場合に実行します。
 - INSTALLPATH には ProcessSaver のインストールフォルダーを設定してください。
デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。

```
@echo off

rem ProcessSaver のインストールフォルダーを設定する
set INSTALLPATH=C:¥Program Files¥HA¥ProcessSaver

set STATUS_CHECK=%INSTALLPATH%¥bin¥SQLServer_status_check.ps1
set PSSLEEP=%INSTALLPATH%¥bin¥PSSleep.exe

rem SQLServer ステータス確認スクリプトを再起動(念のため、プロセス停止処理も実行)
FOR /F "usebackq tokens=2 delims=: " %%A IN (` powershell "Get-WmiObject -Class Win32_Process |
Where-Object { $_.CommandLine -like ¥"*SQLServer_status_check.ps1*¥" } | Sort-Object -Property
CreationDate | Select-Object ProcessID -First 1 | Format-List"`) DO SET PROCESSID=%%A
powershell "Stop-Process -Id %PROCESSID% -Force"

powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe' -
ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File ¥"%STATUS_CHECK%¥"' -WindowStyle Hidden"

rem スクリプト起動と監視のすれ違いを防ぐため、少し待機
"%PSSLEEP%" 10

exit
```

4. ストール監視スクリプトの設定

- SQLServer_check.conf

```
# ファイルの更新をチェックする間隔
STALL_CHECK_INTERVAL 600

# DATABASE のステータスをチェックしファイルを更新する間隔
# STALL_CHECK_INTERVAL より短い時間を設定すること
STATUS_CHECK_INTERVAL 10

# ステータスをチェックする DATABASE が存在するインスタンス名
# 既定のインスタンスの場合は、サーバー名 のみ
# その他の場合は、サーバー名¥インスタンス名 を設定すること
SERVER_NAME SERVER01

# ステータスをチェックするDATABASE のリスト
# 複数チェックする場合は、","を用いて設定する
# (例) Database1 とDatabase2 をチェックする場合
# DATABASE_LIST Database1,Database2
DATABASE_LIST Database1,Database2

# 異常として扱う DATABASE のステータス
# 複数指定する場合は、","を用いて設定する
# (例) OFFLINE とEMERGENCY をチェックする場合
# ERROR_DB_STATUS OFFLINE,EMERGENCY
#ERROR_DB_STATUS OFFLINE,EMERGENCY

# ログをローテートするサイズ(KB)
# 指定したサイズを超えるとログファイルをローテートする
LOG_LOTATE_SIZE 10000
```

23. (続き)

-STALL_CHECK_INTERVAL

IIS_stall_check.ps1 が管理ファイルの更新を確認する間隔です。この時間、管理ファイルが更新されていない場合にストールしていると判断します。デフォルトは 600秒です。

-STATUS_CHECK_INTERVAL

IIS_status_check.ps1 がデータベースのステータスを確認し管理ファイルを更新する間隔です。デフォルトは 10秒です。

-SERVER_NAME

監視するデータベースが存在するインスタンス名を指定します。

-DATABASE_LIST

ステータスを確認するデータベースを指定します。複数のデータベースを指定したい場合は、"/,(コンマ)で区切って指定してください。

-ERROR_DB_STATUS

取得したデータベースのステータスのうち異常と判定するステータスを指定します。

指定されたステータスを取得した場合、管理ファイルの更新を行いません。

異常と判定するステータスが複数ある場合は、"/,(コンマ)で区切って指定してください。

DATABASE_LIST にて複数のデータベースを監視する設定となっている場合、

どれか1個のデータベースが異常と判定されると管理ファイルの更新を行いません。

設定しない場合、ステータスが取得できれば正常と判定します。デフォルトではコメントアウトされています。

-LOG_ROTATE_SIZE

SQLServer_status_check.ps1 の内部ログのサイズを指定します。ログファイルが指定したサイズを超えた場合、1世代分バックアップを行います。既にバックアップファイルが存在する場合、そのファイルは削除されます。デフォルトは 10000KB です。

23. (続き)

- SQLServer_stall_check.ps1

管理ファイルの更新を定期的に監視するスクリプトです。

- installPath には ProcessSaver のインストールフォルダーを指定してください。

デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。

```
Set-PSDebug -Strict
$errorActionPreference = "SilentlyContinue"

#各ファイル名の設定
$installPath = "C:¥Program Files¥HA¥ProcessSaver"
$configFile = $installPath + "¥bin¥SQLServer_check.conf"
$updateFile = $installPath + "¥bin¥SQLServer_staus"
$logFile = $installPath + "¥log¥SQLServer_stall.log"
$rotateLogFile = $installPath + "¥log¥SQLServer_stall.log.save1"

$interval = 600 * 1000
$binFirst = $true
```

以下省略(詳細は実ファイルを確認してください)

23. (続き)

- SQLServer_status_check.ps1

定期的に SQLServer の管理テーブル(sys.databases 表)を参照してデータベースの状態を確認し、管理ファイルを更新するスクリプトです。

- installPath には ProcessSaver のインストールフォルダーを指定してください。

デフォルトは、 C:¥Program Files¥HA¥ProcessSaver です。

```
Set-PSDebug -Strict
$errorActionPreference = "SilentlyContinue"

# 各ファイル名の設定
$installPath = "C:¥Program Files¥HA¥ProcessSaver"
$configFile = $installPath + "¥bin¥SQLServer_check.conf"
$updateFile = $installPath + "¥bin¥SQLServer_staus"
$logFile = $installPath + "¥log¥SQLServer_status.log"
$rotatelogFile = $installPath + "¥log¥SQLServer_status.log.save1"

$serverName = $null
$dbListStr = $null
$rotateSize = $null
$errStatusStr = $null
$interval = 10 * 1000
$rotateSize = 10000 * 1024
```

以下省略(詳細は実ファイルを確認してください)

23. (続き)

- SQLServer_script_start.bat

SQLServer のストール監視を開始するためのバッチファイルです。

OS 起動後、自動で監視を行いたい場合は、タスクスケジューラに本バッチを登録してください。

- INSTALLPATH には ProcessSaver のインストールフォルダーを設定してください。

デフォルトは、C:¥Program Files¥HA¥ProcessSaver です。

- 本バッチは SQLServer のデータベースにアクセス可能なユーザーで実行してください。

```
@echo off

rem ProcessSaver のインストールフォルダーを設定する
set INSTALLPATH=C:¥Program Files¥HA¥ProcessSaver

set PSSLEEP=%INSTALLPATH%¥bin¥PSSleep.exe

REM ステータス確認スクリプト実行
powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe'
-ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File
¥"%INSTALLPATH%¥bin¥SQLServer_status_check.ps1¥'" -WindowStyle Hidden"

REM 念のため 10秒待機
"%PSSLEEP%" 10

REM ストール監視スクリプト実行
powershell -NoProfile -ExecutionPolicy Unrestricted -Command "Start-Process -FilePath 'powershell.exe'
-ArgumentList '-NoProfile -ExecutionPolicy Unrestricted -File
¥"%INSTALLPATH%¥bin¥SQLServer_stall_check.ps1¥'" -WindowStyle Hidden"
```

23. (続き)

- SQLServer_stall_check_stop.bat

SQLServer_stall_check_stop.bat を停止するバッチファイルです。

```
@echo off  
  
rem SQLServer stall 監視スクリプトを停止する(プロセス停止)  
FOR /F "usebackq tokens=2 delims=: " %%A IN (`powershell "Get-WmiObject -Class Win32_Process |  
Where-Object { $_.CommandLine -like ¥"*SQLServer_stall_check.ps1*¥" } | Sort-Object -Property  
CreationDate | Select-Object ProcessID -First 1 | Format-List"`) DO SET PROCESSID=%%A  
powershell "Stop-Process -Id %PROCESSID% -Force"
```

- SQLServer_status_check_stop.bat

SQLServer_status_check_stop.bat を停止するバッチファイルです。

```
@echo off  
  
rem SQLServer ステータス確認スクリプトを停止する(プロセス停止)  
FOR /F "usebackq tokens=2 delims=: " %%A IN (`powershell "Get-WmiObject -Class Win32_Process |  
Where-Object { $_.CommandLine -like ¥"*SQLServer_status_check.ps1*¥" } | Sort-Object -Property  
CreationDate | Select-Object ProcessID -First 1 | Format-List"`) DO SET PROCESSID=%%A  
powershell "Stop-Process -Id %PROCESSID% -Force"
```

 **Orchestrating** a brighter world

NEC