

# HP Performance Agent

Linux オペレーティング システム

ソフトウェア バージョン : 4.70

---

## インストール、設定ガイド

Manufacturing Part Number: B7491-99091

本書の発行日 : 2007 年 9 月

ソフトウェアのリリース日 : 2007 年 9 月



## ご注意

### 保証書

HP 製品およびサービスに対する保証は、それらの製品およびサービスに付属している保証規定に明記された条項に限られます。本書に記載された内容は、追加の保証を規定するものではありません。HP は、本書の技術的および編集上の誤りや不備について、その責任を負わないものとします。

この情報は予告なしに変更されることがあります。

### 権利の制限

本書で取り扱っているコンピュータ ソフトウェアは秘密情報であり、その保有、使用、または複製には、HP から使用許諾を得る必要があります。FAR 12.211 および 12.212 に従って、商業用コンピュータ ソフトウェア、コンピュータ ソフトウェア ドキュメンテーション、および商業用製品の技術データは、ベンダ標準の商業用ライセンスのもとで、米国政府にライセンスが付与されます。

### 著作権

© Copyright 1983-2007 Hewlett-Packard Development Company, L.P.

### 商標

UNIX® は、The Open Group の登録商標です。

Adobe®、Acrobat®、および PostScript® は Adobe Systems Incorporated の商標です。

Motif® は、米国およびその他の国々における Open Software Foundation の登録商標です。

その他の製品名は各社の商標またはサービス マークです。

## サポートについて

次の HP ソフトウェア サポート専用 Web サイトを参照することができます。

**<http://www.hp.com/go/hpsoftwaresupport>**

HP ソフトウェア オンライン サポートでは、対話型テクニカル サポート ツールをすぐにご利用いただけます。このサポート サイトでは、次の機能が用意されています。

- 技術情報の検索
- サポート ケースの登録とトラッキング、およびエンハンスメント要求の送信とトラッキング
- ソフトウェア パッチのダウンロード
- サポート 契約の管理
- HP サポート連絡先の検索
- 利用可能なサービスの参照
- ユーザー同士のディスカッション
- ソフトウェア トレーニングの検索と登録

ほとんどのサポート エリアでは、アクセスするために HP Passport ユーザーとして登録、サインインする必要があります。また、多くのサポート エリアでサポート契約が必要です。

アクセス レベルに関する詳細は以下を参照してください。

**[http://h20230.www2.hp.com/new\\_access\\_levels.jsp](http://h20230.www2.hp.com/new_access_levels.jsp)**

HP Passport ID の登録は以下で行うことができます。

**<http://h20229.www2.hp.com/passport-registration.html>**



# 目次

<b>1 HP Performance Agent のインストールまたはアップグレード</b> .....	7
HP Performance Agent について .....	7
インストール要件 .....	9
対応しているアーキテクチャおよびオペレーティング システムのバージョン .....	9
ディスク スペース .....	10
インストールまたはアップグレードの手順 .....	11
アップグレード .....	11
初回インストールの場合 .....	11
アクティブなパフォーマンス ツール プロセスの終了 .....	11
Performance Agent のインストール .....	12
Debian での Performance Agent のインストール .....	14
Operations Manager を使用する Performance Agent のインストール .....	14
仮想環境における Performance Agent .....	15
サービス コンソール上の Performance Agent .....	15
仮想マシン上の Performance Agent .....	16
Performance Agent の削除 .....	17
<b>2 HP Performance Agent の実行方法</b> .....	19
Performance Agent の起動と終了 .....	20
自動的な起動と終了 .....	21
ステータスの確認方法 .....	22
ディレクトリのサンプル .....	22
ファイアウォールを介した通信 .....	23
Performance Agent ポートの設定 .....	24
ファイアウォール環境での HTTP クライアントの設定 .....	27
ファイアウォール設定の確認 .....	29

複数の IP アドレスを持つシステムの設定 .....	29
安全な通信の設定 .....	30
証明書の使用 .....	30
クライアント認証の使用 .....	30
authip ファイルによる認証の有効化 .....	31
authip ファイルのフォーマット .....	32
データソースの設定 .....	33
datasources 設定ファイルのフォーマット .....	33
データソースの削除 .....	34
parm ファイル .....	35
アラームの定義 .....	36
<b>3 ドキュメント .....</b>	<b>37</b>
Performance Agent ドキュメント一覧 .....	38
Web 上でのドキュメントの参照 .....	39
Adobe Acrobat ファイル .....	40
<b>4 coda の設定 .....</b>	<b>41</b>
はじめに .....	41
coda 名前空間 .....	41
coda 通信名前空間 (coda.comm) .....	44
通信ブローカー名前空間 (bbc.cb) .....	48
通信ブローカー ポート名前空間 (bbc.cb.ports) .....	49
HTTP 名前空間 (bbc.http) .....	51
<b>用語集 .....</b>	<b>53</b>
<b>索引 .....</b>	<b>61</b>

# 1 HP Performance Agent のインストール またはアップグレード

## HP Performance Agent について

HP Performance Agent は、ご使用の Linux システムから、パフォーマンス、リソース、およびトランザクションデータを取り込みます。また、最小限のシステムリソースを使用して、システム上の現在および過去のリソース データに関するアラーム状態の連続的な収集、要約、タイム スタンプ作成、および検出を行います。

Performance Agent は、VMware ESX Server でサポートされるようになりました。詳細は、15 ページの「[仮想環境における Performance Agent](#)」を参照してください。

ローカル エクスポート機能、または HP Reporter や HP Performance Manager 4.0 以降などの相補的な HP ソフトウェア製品を使用して、データを分析することができます。さらに、Performance Agent は、HP Network Node Manager (NNM) および HP Operations Manager へのデータ アクセスとアラーム通知機能を備えています。



本書における Performance Manager という語は、UNIX および Windows プラットフォーム対応のバージョン 4.0 およびそれ以降のみを指します。Performance Manager 3.x (PerfView) は、Performance Agent for Linux には接続しません。

Performance Agent は、データ ソース統合 (DSI: Data Source Integration) 技術を使用して、アプリケーション、データベース、ネットワーク、その他のオペレーティング システムなどの外部データ ソースからデータを受信し、アラーム状態を検出し、ログを記録します。

システムから Performance Agent によって記録および保存された包括的なシステムパフォーマンス データを使用して、次のようなタスクが可能となります。

- 環境の作業負荷の特徴づけ
- リソースの使用量と負荷バランスの分析
- 過去のデータの傾向分析とボトルネックの分離および特定
- エラー状態への対応

- トランザクション応答時間に基づいたサービスレベルの管理
- 容量計画の実行
- システム管理問題の事前の解決

HP Performance Agent に関する一般的な説明は、『HP Performance Agent for UNIX ユーザー マニュアル』を参照してください。



HP Operations Agent、HP Operations Manager UNIX Management Server、HP Performance Manager、HP Performance Insight、および OV Internet Services などの HP ソフトウェア製品がインストールされているシステムでは、Performance Agent 4.70 のインストール後に、これらのソフトウェアを再起動することをお勧めします。



# インストール要件

HP Performance Agent をインストールする前に、システムがこのセクションに記載されている要件を満たしていることを確認してください。

## 対応しているアーキテクチャおよびオペレーティング システムのバージョン

以下の表は、対応しているアーキテクチャおよびオペレーティング システムのバージョンです。

アーキテクチャ	オペレーティング システム
32 ビット Intel Pentium (x86)	<ul style="list-style-type: none"><li>• RedHat Enterprise Linux (ES/AS/WS) 3、4 および 5</li><li>• SuSE Enterprise Server 9.x および 10.x</li><li>• Debian 3.0 r2/r3/r4、3.1</li><li>• Asianux 2.0 および 3.0</li><li>• TurboLinux Server 8 および 10</li><li>• VMware ESX Server 2.5.x および 3.0.x</li></ul>
64 ビット システム (x86_64)	<ul style="list-style-type: none"><li>• RedHat Enterprise Linux (ES/AS/WS) 3、4 および 5</li><li>• SuSE Enterprise Server 9.x および 10.x</li><li>• Asianux 2.0 および 3.0</li><li>• VMware ESX Server 2.5.x および 3.0.x</li></ul>
64 ビット システム (IA64)	<ul style="list-style-type: none"><li>• RedHat Enterprise Linux (ES/AS/WS) 4 Update 4</li><li>• RedHat Enterprise Linux (ES/AS/WS) 5</li><li>• SuSE Enterprise Server 10.x</li></ul>

## ディスクスペース

Performance Agent は /opt/perf/ および /opt/OV/ ディレクトリにインストールされており、ログファイルとステータスファイルを /var/opt/perf/ および /var/opt/OV/ ディレクトリに作成します。

- Performance Agent を初めてインストールする場合は、/opt/perf/ および /opt/OV/ ディレクトリに約 50 MB のディスクスペースが必要です。
- Performance Agent データベースおよびステータスファイルには、/var/opt/perf/ および /var/opt/OV/ ディレクトリに 125 MB のディスクスペースが必要です。

Performance Agent の初回のインストールの際、デフォルトの parm ファイルは、ログファイルのデータを最大で約 60 MB まで収集するように設定されています。

parm ファイルを編集して scopeux を再開することにより、ログファイルのサイズを変更できます。parm ファイルの編集、parm ファイルを使用してログファイルのデータ記憶領域を制限および構成する方法については、『HP Performance Agent for UNIX ユーザー マニュアル』の第 2 章を参照してください。



新しいハードウェアを追加したり、設定を変更する場合は、scopeux を停止し、再起動して変更内容を有効にすることをお勧めします。

# インストールまたはアップグレードの手順

Performance Agent は DVD-ROM インストール メディアに収録されています。ドキュメントを含む製品のサイズはおよそ 60 MB です。

## アップグレード

Performance Agent または GlancePlus をすでにシステムにインストールしている場合は、以下のタスクを実行します。

- 実行中と思われるパフォーマンス ツールまたはプロセスをすべて終了します。詳細は、「アクティブなパフォーマンス ツールプロセスの終了」を参照してください。
- Performance Agent をインストールします。Performance Agent のインストールの詳細は、「Performance Agent のインストール」を参照してください。

## 初回インストールの場合

Performance Agent を初めてインストールする場合は、「Performance Agent のインストール」を参照してください。

## アクティブなパフォーマンス ツール プロセスの終了

- 1 **root** でログインします。
- 2 アクティブなパフォーマンス ツールを確認するには、次のように入力して perfstat を実行します。

```
/opt/perf/bin/perfstat
```

perfstat によって、アクティブなパフォーマンス ツール プロセスが報告された場合は、そのプロセスを終了します (プロセスを終了する前に、それらのツールを使用していないことを確認してください)。

- 3 インストール済みの Performance Agent が動作中の場合は、次のように入力して終了します。

```
/opt/perf/bin/ovpa stop
```

▶ parm, alarmdef, ttd.conf, datasources などのカスタマイズした設定ファイルやその他のカスタマイズしたログ ファイルは、新しくインストールする際に上書きされることはありません。新しい設定ファイルは /opt/perf/newconfig/ ディレクトリにインストールされます。

- 4 midaemon (測定インターフェイス デモン) が終了していることを、次のように入力して確認します。

```
ps -ef | grep midaemon
```

midaemon がアクティブである場合は、次のように入力して終了します。

```
/opt/perf/bin/midaemon -T
```

- 5 ttd (トランザクション デモン) が終了していることを、次のように入力して確認します。

```
ps -ef | grep ttd
```

ttd がアクティブである場合は、次のように入力して終了します。

```
/opt/perf/bin/ttd k
```

▶ ttd を終了する場合は、ARM 装備のアプリケーションをすべて終了してから、ttd と Performance Agent のプロセスを再開してください。

- 6 perfstat コマンドを再度実行して、アクティブなパフォーマンス ツールやプロセスがないことを確認します。すべてのツールまたはプロセスを終了し、カスタマイズした設定ファイルをバックアップしたら、インストールプロセスに進みます。

## Performance Agent のインストール

▶ Debian にインストールする場合は、14 ページの「Debian での Performance Agent のインストール」を参照してください。

- 1 **root** でログインしていることを確認します。
- 2 インストール DVD-ROM をドライブに挿入します。

通常、Linux システムは、自動的に DVD-ROM をマウントするように設定されています。DVD-ROM を自動マウントするように設定されていないシステムの場合は、ファイル マネージャを使用するか、または以下のコマンドを入力して、手動でマウントする必要があります。

```
mount -t <タイプ> <デバイス> <ディレクトリ>
```

ここで、<タイプ> はファイル システムのタイプ、<デバイス> は DVD-ROM デバイス、<ディレクトリ> は DVD-ROM ディレクトリがマウントされる場所です。次に例を示します。

```
mount -t iso9660 /dev/dvdrom /mnt/dvdrom
```

- 3 ターミナル ウィンドウを開きます。コマンド ラインから `ovpa.install` スクリプトを実行します。
- 4 ターミナル ウィンドウで次を入力し、現在の作業ディレクトリを変更します。

```
cd /<ディレクトリ>
```

ここで、<ディレクトリ> は、ご使用の DVD-ROM ディレクトリです。

- 5 `ls` と入力して、正しいディレクトリに移動しているかどうかを確認します。README ファイル、`ovpa.install` および `glance.install` スクリプト、`paperdocs`、`packages`、および `ReleaseNotes` サブディレクトリが表示されます。
- 6 次のように入力し、インストール スクリプトを実行します。

```
./ovpa.install
```

`ovpa.install` スクリプトには、さらに高度なインストールに用いるコマンド ライン オプションが含まれています。コマンドの構文は次のとおりです。

```
ovpa.install [-n] [-f] [-v]
```

コマンド ライン オプションの意味は以下のとおりです。

- n インストールの最後に、`ovpa start` に関するメッセージを表示しません。
- f 強制インストール (`--force`、`--nodeps`)
- v 詳細インストール

Performance Agent のインストールが完了しました。

デフォルトでは、Performance Agent プロセスはシステムのリブート時に自動的に開始されるように設定されています。20 ページの「Performance Agent の起動と終了」を参照してください。

第 2 章「Performance Agent の実行方法」では、ご使用の環境に合わせて Performance Agent の設定を行う場合の作業を説明しています。



Performance Agent の新機能と変更点についての詳細は、`/opt/perf/ReleaseNotes/OVPA` にある「リリース ノート」を参照してください。

## Debian での Performance Agent のインストール

Debian システムでは、まず最初に `/usr/bin/rpm` (RedHat Package Manager) をインストールする必要があります ( インストールされていない場合 )。Performance Agent のインストールには rpm が必要です。

これまでに rpm を使用していない場合は、次の手順を実行します。

- 1 `root` でログインしていることを確認します。
- 2 次のように入力して、rpm データベースを作成します。

```
# mkdir /var/lib/rpm
# rpm --initdb --dbpath /var/lib/rpm
```

- 3 12 ページの「Performance Agent のインストール」の手順 2～手順 6 を行います。

## Operations Manager を使用する Performance Agent のインストール

Operations Manager for UNIX 8.x またはそれ以降を使用している場合、管理サーバーから Linux の管理ノードに、Performance Agent をインストールすることができます。

Operations Manager for UNIX バージョン 8.x でのインストールの方法については、『HP Operations Manager 管理リファレンス ガイド』の第 4 章「HP Performance Agent」を参照してください。

# 仮想環境における Performance Agent

このバージョンの Performance Agent は、VMware ESX Server でサポートされません。Performance Agent は VMware ESX Server のサービス コンソールにインストールする必要があります。詳細は、「サービス コンソール上の Performance Agent」を参照してください。

このバージョンの Performance Agent は、VMware ESX Server 2.5.2 以降にホストされた仮想マシン (VM) の監視にも対応しています。詳細は、「仮想マシン上の Performance Agent」を参照してください。

- ▶ VMware ESX Server では、デフォルトでポートが有効になっていません。VMware ESX Server で単一および 2 つの CODA ポートを有効にする方法については、24 ページの「Performance Agent ポートの設定」を参照してください。

## サービス コンソール上の Performance Agent

Performance Agent は VMware ESX Server のサービス コンソールおよび VMWare ESX Server にホストされたシステムにインストールできます。インストールの手順については、「インストールまたはアップグレードの手順」を参照してください。Performance Agent は以下のデータを取り込みます。

- システム全体のパフォーマンスおよび物理的なシステム リソース使用量に関するデータ
- 各ゲストのパフォーマンスおよび物理的なシステム リソース使用量に関するデータ

個々のゲスト オペレーティング システムのパフォーマンスとリソースに関するデータを記録するために、BYLS という新しいクラスのメトリックが導入されています。

- ▶ すべてのメトリックとその説明が記載されたリストは、『Dictionary of Performance Metrics』を参照してください。

## 仮想マシン上の Performance Agent

このバージョンの Performance Agent は、VMware ESX Server 2.5.2 以降にホストされた仮想マシン (VM) の監視に対応しています。インストール手順については、「インストールまたはアップグレードの手順」を参照してください。

仮想マシンに **VMWare Tools** をインストールします。仮想マシンにインストールされた Performance Agent (VMWare ESX Server 2.5.2 以降にホスト) では、VMWare Tools を使用して、仮想マシンをゲスト オペレーティング システムとして認識させる必要があります。

VMWare ESX Server 3.0.1 にホストされた仮想マシンでは、仮想マシンの **VMware Guest API** を有効にします。仮想マシンによるシステムの物理リソースの使用量データを取り込むために、Performance Agent では **VMWare Guest API** を使用します。**VMware Guest API** の詳細は、VMWare の **VMware Guest SDK** で利用できる最新の資料を参照してください。

 すべてのメトリックとその説明が記載されたリストは、リリース ノートおよび『Dictionary of Performance Metrics』を参照してください。

以下は、仮想環境におけるすべての機能のサポート状況の一覧です。

機能	Linux	VMware ESX Server
アラーム	利用可	利用可
データ ソース統合機能	利用可	利用可
アプリケーション メトリック クラス	利用可	利用不可
BYLS メトリック	利用不可	利用可
proc メトリック	利用可	利用不可
トランザクション メトリック クラス	利用可	利用不可
テーブル クラス	利用可	利用不可
ARM	利用可	利用不可



# Performance Agent の削除

システムから Performance Agent を削除する場合は、`/opt/perf/bin/` ディレクトリの `ovpa.remove` スクリプトを実行します。Performance Agent を削除する前に、作成されたログ ファイルおよびデータ ファイルを削除するかどうかの確認が、このスクリプトによって行われます。

`ovpa.remove` スクリプトは、次のコマンド ライン オプションを使用することで、柔軟に製品を削除できます。コマンドの構文は次のとおりです。

## **ovpa.remove [-f] [-h] [-r]**

- f 強制削除します。製品を削除する際、確認画面を表示しません ( 非インタラクティブ モード )。  
**メモ**：このオプションは、インストールされている GlancePlus も削除します。
- h ヘルプ：このコマンドの使用に関するメッセージを表示します。
- r Performance Agent の設定、データ、および製品ファイルを削除します。  
**メモ**：f オプションと r オプションを共に使用すると、確認画面を表示せずに、インストールされている GlancePlus を含め、すべての製品、設定、およびデータ ファイルを削除します。

他の HP ソフトウェア製品でも共有されていて、他のツールの使用に必要な一部の製品パッケージは、システムにインストールされたままになることがあります。関連するツールがすべて削除された時点で、これらのパッケージも削除されます。



## 2 HP Performance Agent の実行方法

この章では、Performance Agent をシステムにインストールした後のプログラムの起動と実行に関する作業について説明します。次のような内容を取り上げます。

- Performance Agent の起動と終了
- ファイアウォールを介した通信
- Performance Agent ポートの設定
- 安全な通信の設定
- データソースの設定
- アラームの定義



データソース統合 (DSI: Data Source Integration) 機能を使用して、他のソースのデータを記録する場合は、『HP Performance Agent for UNIX データソース統合ガイド』を参照してください。

Performance Agent は Linux での HTTP データ通信メカニズムに対応しています。

# Performance Agent の起動と終了

インストールの終了後、Performance Agent を起動できます。Performance Agent の終了、起動、または再起動が必要な場合は、ovpa スクリプトを使用して、現在実行中のプロセスを終了または再起動してください。

Performance Agent とそのプロセスを起動する手順は次のとおりです。

1 **root** でログインします。

2 次のように入力します。/opt/perf/bin/ovpa start

ovpa start スクリプトにより、Performance Agent と、scopeux (データコレクタ)、midaemon (測定インターフェイス デーモン)、ttd (トランザクショントラッキング デーモン)、coda、ovc、ovbbccb、アラームジェネレータなどのすべてのプロセスが起動します。スクリプトを実行すると、起動されたプロセスのステータスが画面に表示されます。

ovpa スクリプトとオプションを使用して、実行中の Performance Agent のプロセスを終了したり再起動したりすることができます。

- ovpa stop により、ttd (トランザクショントラッキング デーモン)、ovc、ovbbccb を除く、Performance Agent のすべてのプロセスが終了します。これらのプロセスは常に実行されている必要があります。Operations Manager エージェントがシステムで実行中の場合は、ovpa stop を入力しても coda デーモンは終了しません。



ttd を終了する必要がある場合は、実行中の ARM 装備のアプリケーションを終了してから、ttd と Performance Agent のプロセスを再起動してください。

- ovpa restart server は、coda を終了し、続いて再起動します。その際、一時的にアラームが無効になり、Performance Manager などのクライアントにアクセスできなくなります。その後、datasources ファイルを再度読み込みます。また、このオプションは、perfalarm プロセスを終了し、その後、再起動させて、alarmdef ファイルを再度読み込みます。
- ovpa restart は、scopeux およびサーバー プロセスを一時的に終了させ、その後、再起動させます。また、parm ファイルが読み込まれ、トランザクション デーモン ttd に設定ファイル ttd.conf を再度読み込ませます。
- ovpa restart alarm により、アラーム ジェネレータ プロセスは alarmdef ファイルを再度読み込みます。これにより、このファイルに変更を加えた場合に、すべての Performance Agent プロセスを再起動することなく、新しいアラーム定義を有効にすることができます。この動作によって他のプロセスが中断されることはありません。

## 自動的な起動と終了

システムの再起動時に自動で Performance Agent を起動するプロセス、およびシステムのシャットダウン時に Performance Agent を終了するプロセスは、Performance Agent 起動およびシャットダウン スクリプトと Performance Agent 起動設定ファイルにより制御されます。このスクリプトと設定ファイルは、ご使用の Linux ディストリビューションに固有の要件によって、以下のように異なります。

**表 1 Performance Agent 起動スクリプトおよび設定ファイル**

Linux ディストリビューション	Performance Agent 起動およびシャットダウン スクリプト	Performance Agent 起動設定ファイル
Red Hat	/etc/rc.d/init.d/ovpa	/etc/sysconfig/ovpa
SuSE	/etc/init.d/ovpa	/etc/ovpa.conf
TurboLinux	/etc/rc.d/init.d/ovpa	/etc/sysconfig/ovpa
Debian	/etc/init.d/ovpa	/etc/ovpa.conf
ESX Server	/etc/rc.d/init.d/ovpa	/etc/sysconfig/ovpa

Performance Agent 起動設定ファイルには、システム起動時の Performance Agent の動作を制御するさまざまな環境変数が含まれています。このファイルは、Performance Agent システム起動およびシャットダウン スクリプトのソースファイルです。

次の環境変数およびシェル変数を編集して、Performance Agent のデフォルトの動作を変更することができます。

- OVPA\_START は、システムを再起動したときの Performance Agent の自動起動を制御します。この変数は、以下の値のうちのいずれかをとりまします。
  - 0 システム起動時に Performance Agent を起動しません。
  - 1 システム起動時に Performance Agent を起動します。
- OVPA\_START\_COMMAND は、システムの再起動時に、Performance Agent を起動するスクリプト オプションを指定する変数です。通常、この変数は次のように設定されます。

```
/opt/perf/bin/ovpa start
```



Performance Agent 起動設定ファイルは、Performance Agent のパッチの適用時または更新時に上書きされないため、加えた変更はすべて保持されます。

## ステータスの確認方法

Performance Agent を起動すると、`/var/opt/perf/` および `/var/opt/OV/` ディレクトリに数個のステータス ファイルが作成されます。`perfstat` コマンドを使用して、Performance Agent のすべてまたは一部のプロセスのステータスを確認できます。

次のステータス ファイルには、Performance Agent のプロセスで生じる障害を解決するのに使用できる診断情報が含まれます。

```
/var/opt/perf/status.scope  
/var/opt/perf/status.perfalarm  
/var/opt/perf/status.ttd  
/var/opt/perf/status.mi  
/var/opt/OV/log/coda.txt
```

## ディレクトリのサンプル

`/opt/perf/examples/` ディレクトリには、設定ファイルや構文の例、および当社のパフォーマンス ツールをカスタマイズするために使用するプログラム ファイルのサンプルが含まれています。たとえば、`/opt/perf/examples/config/` サブディレクトリには、サンプルのアラーム定義と `parm` ファイルのアプリケーション指定パラメータの例が含まれています。詳細は、`/opt/perf/examples/README` ファイルを参照してください。

## ファイアウォールを介した通信

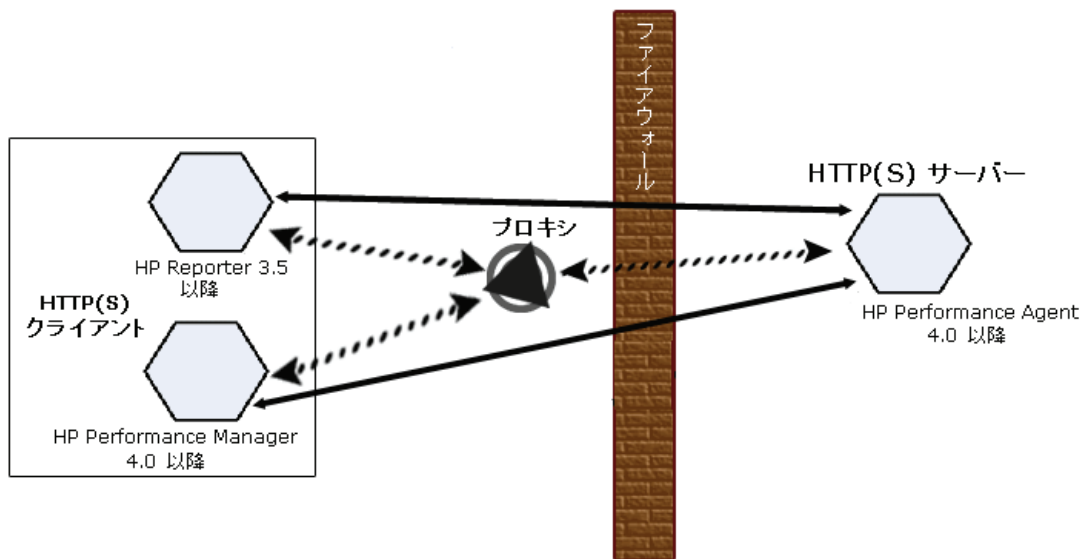
ファイアウォールは、あるネットワークと他のネットワークの間のデータの流れをフィルタリングする方法として定義することができます。Performance Agent では、クライアントとサーバー アプリケーション間のデータ アクセスに関して、以前からパケットフィルタリング ネットワーク ファイアウォールを介した通信メカニズムに対応していましたが、今回さらに、HTTP ベースの通信インターフェイスに対応しました。Performance Agent は内部で Black Box Communication (BBC) コンポーネントを用いて、HTTP ベースのデータ通信を実現します。BBC コンポーネントは、HTTP または HTTPS (BBC5) をベースとした、ファイアウォール対応の単一ポートの通信インフラを提供します。HTTP ベースのインターフェイスはプロキシが使用できるため、柔軟で、ポート数が少なく済み、ファイアウォールに適しています。

2 つのシステム間にネットワーク ファイアウォールがある場合は、Performance Manager および Reporter が Performance Agent システムからデータを取得します。



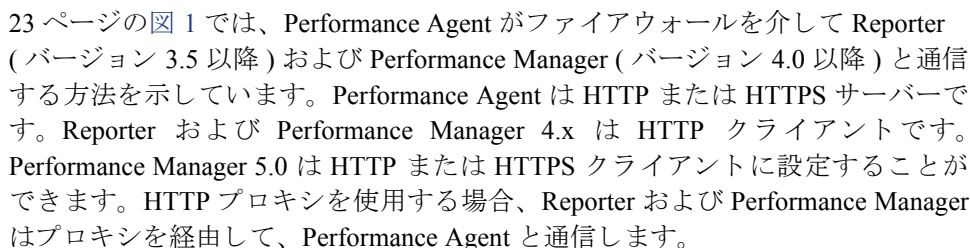
本書で使用されている Performance Manager という名前は、Performance Manager バージョン 4.0 以降を指します。Performance Manager 3.x は、Performance Agent for Linux には接続しません。


図 1 ファイアウォール環境での Performance Agent との通信



ファイアウォールを設定するためには、通信を開始するシステム (クライアント) および通信要求を受信するシステム (サーバー) を把握し、それに応じてファイアウォールのルールを設定することが重要です。典型的なリモート通信では、ソースポートを使用しているクライアントが、リモートシステムの宛先ポートをリスンしているサーバーに接続します。クライアントおよびサーバーのデータフローを含む、ファイアウォール環境を把握します。以下のタスクを実行し、ファイアウォール環境での Performance Agent との通信を設定します。

- 1 Performance Agent ポートの設定
- 2 ファイアウォール環境での HTTP クライアントの設定
- 3 ファイアウォール設定の確認

23 ページの  1 では、Performance Agent がファイアウォールを介して Reporter (バージョン 3.5 以降) および Performance Manager (バージョン 4.0 以降) と通信する方法を示しています。Performance Agent は HTTP または HTTPS サーバーです。Reporter および Performance Manager 4.x は HTTP クライアントです。Performance Manager 5.0 は HTTP または HTTPS クライアントに設定することができます。HTTP プロキシを使用する場合、Reporter および Performance Manager はプロキシを経由して、Performance Agent と通信します。

 Linux 環境で、Performance Manager 3.x を Performance Manager 4.0 またはそれ以降にアップグレードできない場合は、extract プログラムの export 機能を使用して、ローカルで Performance Agent for Linux のデータにアクセスする必要があります。

## Performance Agent ポートの設定

ファイアウォール環境の Performance Agent ポートは、次のいずれかの方法で設定し、確認することができます。

- 2 ポート通信の設定
- 単一ポート通信の設定
- ポート設定の確認

BBC5 を使用する Performance Agent システムでは、デフォルトで BBC 通信ブローカーがポート 383 を使用し、coda は動的に割り当てられたポートを使用します。



## BBC 通信ブローカー用のポート設定

BBC 通信ブローカーが使用するデフォルト ポートを設定できます。ovconfchg コマンドを使用して、Performance Agent システムでのポート設定を変更します。次のオプションのいずれかを使用します。

```
— ovconfchg -ns bbc.cb -set SERVER_PORT <ポート>
```

または、

```
— ovconfchg -ns bbc.cb.ports -set PORTS <ドメイン>:<ポート>
```

```
例: ovconfchg -ns bbc.cb.ports -set PORTS  
xyz.abc.com:50383
```

ポート変更には 2 番目のオプションのほうが適しています。

次のコマンドを入力して、Performance Agent を再起動します。

```
ovpa restart server
```

## 2 ポート通信の設定

デフォルトでは、coda デーモンは、BBC 通信ブローカーが使用するポート 383 に加えて、動的に選択される 2 つ目のポートを使用します。ovconfchg コマンドを使用して、任意の well known ポートをリッスンするよう coda のポートを設定することができます。次のコマンドを入力します。

```
ovconfchg -ns coda.comm -set SERVER_PORT <ポート番号>
```

```
ovpa restart server
```



ファイアウォールを介してリモートから Performance Agent に接続する場合、動的なポートの使用が難しいことがあります。これは、ファイアウォールのどのポートを開くべきか把握できない場合があるためです。

## 単一ポート通信の設定

Performance Agent システムでは、BBC 通信ブローカーがポート 383 を使用し、coda は動的に割り当てられたポートを使用します。ovconfchg コマンドを使用して、通信ブローカーが使用するポートを共有するよう coda のポートを設定することができます。次のオプションのいずれかを使用します。

```
ovconfchg -ns coda.com -set SERVER_BIND_ADDR localhost
ovpa restart server
```



単一ポート通信が設定されている状態で 2 ポート通信を有効にするには、次のコマンドを入力します。

```
ovconfchg -ns coda.com -set SERVER_BIND_ADDR
```

## ポート設定の確認

ポート設定を確認するには、次のコマンドを入力します。

```
perfstat -d
```

次の情報が出力されます。

- coda が使用するポートのポート番号
- BBC 通信ブローカーが使用するポートのポート番号
- ポート設定
- 安全性の高い通信が有効かどうか
- coda メトリック収集が有効かどうか

次に例を示します。

```
Datacomm configuration :
-----
Coda Port                               49552 (Dynamic)
                                         Two port Communication
BBC communication broker port           383
SSL security                             NONE
Coda Metric Collection(Prospector)     Disabled
```

## ファイアウォール環境での HTTP クライアントの設定

ファイアウォール環境で HTTP クライアントを設定する方法は 2 つあります。

- HTTP プロキシを用いた、HTTP クライアント (Reporter または Performance Manager) の設定
- HTTP プロキシを使用しない、HTTP クライアント (Reporter または Performance Manager) の設定

どちらの場合にも、Performance Agent ノードからデータにアクセスするには、HTTP サーバー (Performance Agent) 側のポートが 1 つだけ開いている必要があります。

### HTTP プロキシを用いた、HTTP クライアント (Reporter または Performance Manager) の設定

ファイアウォールを介して通信する場合は、HTTP プロキシの使用をお勧めします。この方法では、環境内ですでに使用されているプロキシを用いることにより、設定が簡略化されます。プロキシが双方向で使用されている場合、ファイアウォールでは厳密に 1 つのポートを開く必要があります。Performance Agent により収集されたデータにアクセスするには、HTTP サーバー (Performance Agent) 用のポートと HTTP クライアント (Reporter および Performance Manager) 用のポートが開かれている必要があります。



デフォルトの 383 ポートは変更しないことをお勧めします。

HTTP プロキシを使用する場合は、Windows または UNIX 対応の Reporter および Performance Manager に対して、Performance Agent に接続するために使用するプロキシを指定する必要があります。

Performance Manager バージョン 5.0 以降および Reporter 3.7 を設定するには、次の手順を実行します。

次のコマンドを入力します。

```
ovconfchg -ns bbc.http -set PROXY proxy:port+(a)-(b)
```

変数 *a* および変数 *b* は、プロキシに適用するホスト名、ネットワーク、および IP アドレスをコンマで区切った形式のリストです。区切り記号「;」または「,」を使用して、1 つの PROXY キーに対して複数のプロキシを定義することができます。リストの前の「-」は、それらの要素では対象のプロキシが使用されないことを示し、リストの前の「+」は、それらの要素で対象のプロキシが使用されることを示します。最初に適合したプロキシが使用されます。

Reporter バージョン 3.6 以前および Performance Manager 4.x を設定するには、次の手順を実行します。

`/var/opt/OV/conf/BBC/default.txt` 設定ファイルを編集します。

`default.txt` ファイルの [DEFAULT] セクションで、PROXY に関連する行を特定し、次のように PROXY パラメータを設定します。

**PROXY web-proxy.hp.com:8088-(localhost, \*.hp.com) + (\*)**

この例では、ローカルマシン (localhost) への要求、および HP 内 (たとえば、**www.hp.com** のような、\*.hp.com に合致するもの) への要求を除いて、web-proxy プロキシが、すべてのサーバー (\*) に対してポート 8088 で使用されます。

## HTTP プロキシを使用しない、HTTP クライアント (Reporter または Performance Manager) の設定

HTTP プロキシが利用できない場合は、HTTP クライアント (Reporter および Performance Manager システム) 上で、追加の設定が必要になります。

Reporter および Performance Manager for Windows が同一のシステムにインストールされていて、双方が同時に Performance Agent にアクセスする場合は、このセクションに記載されている方法でポート範囲を指定します。それらが異なるシステムで実行されている場合は、それぞれに単一のポートを指定することができます。使用している Performance Manager および Reporter のバージョンに合わせて、次のオプションを選択します。

Performance Manager 5.0 以降および Reporter 3.7 を設定するには、次の手順を実行します。

次のコマンドを入力します。

```
ovconfchg -ns bbc.http -set CLIENT_PORT <ポート範囲>
```

<ポート範囲> は、使用するポート範囲です。

次に例を示します。

```
ovconfchg -ns bbc.http -set CLIENT_PORT 14000-14003
```

Reporter バージョン 3.6 以前および Performance Manager 4.x を設定するには、次の手順を実行します。

`/var/opt/OV/conf/BBC/default.txt` ファイルを、以下のように編集します。

- 1 CLIENT\_PORT に適用される行を特定して、`;CLIENT_PORT =` の行をコメント解除します。
- 2 CLIENT\_PORT パラメータに、ポート範囲を指定します。次に例を示します。

CLIENT\_PORT = < ポート範囲 >

< ポート範囲 > は、使用するポート範囲です。次に例を示します。

CLIENT\_PORT = 14000-14003

## ファイアウォール設定の確認

設定を確認するには、次のコマンドを使用します。

```
ovcodautl -ping -n < システム名 >
```

このコマンドの出力には、通信設定の状態が表示されます。

## 複数の IP アドレスを持つシステムの設定

ご使用の環境に、複数のネットワーク インターフェイスおよび IP アドレスを持つシステムがあり、HTTP ベースの通信に専用のインターフェイスを使用する場合は、CLIENT\_BIND\_ADDR パラメータおよび SERVER\_BIND\_ADDR パラメータで、使用する IP アドレスを指定することができます。

- Performance Agent (サーバー) システム上に複数のネットワーク インターフェイスおよび IP アドレスがある場合は、SERVER\_BIND\_ADDR パラメータを次のように指定します。

```
ovconfchg -ns bbc.http -set SERVER_BIND_ADDR <IP Address>
```

- Performance Manager 5.0 (クライアント) システム上に複数のネットワーク インターフェイスおよび IP アドレスがある場合は、CLIENT\_BIND\_ADDR パラメータを次のように指定します。

```
ovconfchg -ns bbc.http -set CLIENT_BIND_ADDR <IP Address>
```

- Reporter/Performance Manager 4.x システム上に複数のネットワーク インターフェイスおよび IP アドレスがある場合は、CLIENT\_BIND\_ADDR パラメータを指定します。

/var/opt/OV/conf/BBC/default.txt ファイルを、以下のように編集します。

- a CLIENT\_BIND\_ADDR に適用される行を特定して、以下の行をコメント解除します。

```
;CLIENT_BIND_ADDR =
```

- b CLIENT\_BIND\_ADDR パラメータに、IP アドレスを指定します。

## 安全な通信の設定

Performance Agent は証明書ベースの安全性の高い通信、およびクライアント認証ベースの通信に対応しています。

### 証明書の使用

Performance Agent は、HP Operations Manager 8.x 環境でのみ、証明書ベースの安全なデータ通信に対応しています。

Operations Manager で安全性の高い通信を設定するには、『HP Operations Manager for UNIX Firewall Concepts and Configuration Guide』を参照してください。Operations Manager 8.x HTTPS エージェントについての詳細は、『HP Operations Manager HTTPS Agent Concepts and Configuration Guide』を参照してください。

Operations Manager 8.x 環境ですでに HTTPS 通信を設定している場合は、Performance Agent と Performance Manager 5.0 の間の安全な通信を設定するために、以下の変更を行います。



Reporter および Performance Manager 4.x は証明書ベースの安全性の高い通信には対応していません。

Performance Agent システムでは、coda に関して、SSL\_SECURITY に REMOTE を設定します。次のコマンドを入力します。

```
ovconfchg -ns coda -set SSL_SECURITY REMOTE
ovcodautl -config
```

### クライアント認証の使用

オプションとして、Performance Manager または Reporter などの製品からのクライアント接続の認証を、Performance Agent で実行することができます。この認証機能を利用することで、特定の Performance Agent インスタンスについて、そのインスタンスへのクライアント接続を許可するホストを指定できます。

クライアント認証機能は、あらゆるバージョンの Performance Manager および Reporter クライアントからの接続を有効または無効にします。この機能を利用するにあたって、クライアント ソフトウェアを更新する必要はありません。

認証プロセスは、認証されるクライアントに透過的に行われるため、クライアント接続は表面上、Performance Agent の過去のバージョンを使用していたときと同じように処理されます。未認証のクライアントには、サービス拒否を示すメッセージが表示されます。

## authip ファイルによる認証の有効化

authip というファイルが存在する場合、認証は有効になります。HTTP 通信が有効なシステムでは、authip ファイルは /var/opt/OV/conf/perf/ ディレクトリにあります。authip ファイルにはホストのリストが含まれ、そのホストからのクライアント接続が許可されます。

- authip ファイルが /var/opt/OV/conf/perf/ ディレクトリに存在する場合、クライアント接続が許可されるホストはこのファイルの内容によって決定されます。Performance Agent インスタンスと同じホストで実行されているクライアントは自動的に認証されるため、エントリを必要としません。長さがゼロの authip ファイルは、Performance Agent ホストで実行されているクライアントのみが接続できることを意味します。
- authip ファイルが /var/opt/OV/conf/perf/ ディレクトリに存在しない場合 ( デフォルト ) は、認証は行われず、Performance Agent の過去のバージョンと同様に、すべてのクライアントに接続が許可されます。

authip ファイルは、クライアントが Performance Agent によるサービスを要求して接続を試みるたびに照会されます。authip ファイルに変更が加えられた場合、変更内容を有効にするために Performance Agent を再起動する必要はありません。

Performance Agent クライアント認証機能では、使用するネットワークで authip ファイルに含まれるクライアント エントリを解決できることが必要です。エントリの性質によっては、DNS、NIS あるいは /etc/hosts/ ファイルによって提供されるようなネーム サービスを必要とする場合があります。

Performance Agent ホストから各 authip エントリへ「ping」が成功すれば、クライアント エントリは解決されています。また、ファイアウォールを介したクライアント認証も同様に、authip ファイル内のクライアント エントリが Performance Agent ホストからの ping に応答する場合に機能します。

## authip ファイルのフォーマット

authip ファイルは次のフォーマットに一致する必要があります。

- 1行あたり1つのクライアントホストをリストする
- クライアントエントリには、次のフォーマットのうち1つ(任意)を使用する
  - 完全修飾ドメイン名
  - エイリアス (alias)
  - IPアドレス(ドットで4つに区切ったIPv4アドレス)
- クライアントエントリにはスペースは使用できない
- 行の始めが「#」のラインは、コメントとして無視される
- ブランクあるいは長さがゼロの行は無視される
- IPアドレスは0から始めない。たとえば、23.10.10.10のIPアドレスを、023.10.10.10と表すことはできません。

/etc/hosts エントリは次のようになります。

```
123.456.789.1 testbox testbox.group1.thecompany.com
```

authip ファイル内で次のいずれかのエントリを使用することで、「testbox」ホストからのクライアント接続が可能となります。

```
#===== authip ファイルエントリの例 =====  
#  
# IP アドレスを使用  
123.456.789.1  
  
#  
# エイリアスを使用  
testbox  
  
#  
# 完全修飾ドメイン名を使用  
testbox.group1.thecompany.com  
  
#===== authip ファイルエントリの例の終わり =====
```



## データソースの設定

Performance Agent は、coda デーモンを使用して、収集したデータをアラーム ジェネレータおよび分析製品に送信します。各データソースには、1つのログファイルセットが含まれます。coda がアクセスするデータソースのリストは、`/var/opt/OV/conf/perf/` ディレクトリにある `datasources` 設定ファイルで管理されます。インストール後、初めて Performance Agent を実行する場合、デフォルトのデータソースである `SCOPE` が事前に設定されていて、アクティブな `scopeux` ログファイルセットへのアクセスが可能です。

他のデータソースを追加する場合は、`datasources` ファイルでそれらを設定することができます。その後、Performance Agent を再起動すると、coda デーモンにより `datasources` ファイルが読み込まれ、検出された各データソースの分析ツールへの通信リンクを経由して、データが利用可能になります。

### datasources 設定ファイルのフォーマット

`datasources` 設定ファイルの各エントリは、1つのログファイルセットで構成されるデータソースを示します。エントリは、データソースの名前とパスを示します。このフィールドでは、ログファイルのパス名以外は大文字小文字の区別はありません。構文は次のようになります。

```
datasource=datasource_name logfile=logfile_set
```

- **datasource** はキーワードです。**datasource\_name** はデータソースを識別するための名称です。たとえば、データソース名はアラーム定義や分析ソフトウェアで使用されます。また、データソース名は固有の名前である必要があります。これらの名前は大文字に変換されます。さらに、データソース名は最大 64 文字まで指定できます。
- **logfile** はキーワードです。また、**logfile\_set** はログファイル (`dsilog` または `scope` プロセスにより作成され、`.log` で終わるファイル) を識別する完全修飾名で、大文字小文字の区別があります。

次に `datasources` ファイルのデータソースエントリを 2 例示します。

```
datasource=SCOPE logfile=/var/opt/perf/datafiles/logglob  
datasource=ASTEX logfile=/tmp/dsidemo/log/astex/ASTEX_SDL
```

datasources の更新後、ovpa restart server コマンドを使用して、coda デーモンを再開始します。まず、ovpa restart server により、すべての coda プロセスが終了されます。次に、coda デーモンが開始され、datasources ファイルを再読み込みして、このファイルに加えられた変更内容を取り込みます。

coda デーモンプロセスを終了すると、Performance Manager の現行の接続がすべて切断されることに注意してください。たとえば、データソースでグラフを描いていて、さらに別のグラフを描く場合には、coda デーモンが再起動されたときに、Performance Manager のデータソースを再選択して、接続を確立しなおす必要があります。

/var/opt/OV/log/coda.txt ファイルの内容を見て、coda デーモンがアクティブになったことを確認するか、またはエラーメッセージを参照します。

## データソースの削除

Performance Manager からのログファイルデータを表示したり、またはログファイルのアラームを処理したりする必要がない場合は、Performance Agent 設定ディレクトリ /var/opt/OV/conf/perf/ の datasources ファイルを編集して、データソースやログファイルセットへのパスを削除することができます。その後、ovpa プログラムを実行して、前出の「データソースの設定」の手順で coda デーモンを再開始します。

データが不要になったときには、ログファイルセットを削除することもできます。datasources から、データソースは削除せずにログファイルセットを削除すると、coda により、そのデータソースは省略されます。

データのログファイルセットへの記録を停止する際に、coda デーモンを開いたままにしておくことが可能です。これにより、Performance Manager で履歴データを表示することができます。この場合は、datasources ファイルからデータソースを削除せずに、dsilog プロセスを停止します。

## parm ファイル

parm ファイルはテキスト ファイルです。ログ ファイルの最大サイズ、対象プロセスのしきい値の定義、およびアプリケーション定義といった scopeux データコレクタの設定を指定します。ファイル内のコメントで、各種設定の概要を確認することができます。

Performance Agent では、parm ファイルは /opt/perf/newconfig/ ディレクトリに存在し、インストール時に /var/opt/perf/ ディレクトリにコピーされます。parm ファイルとそのパラメータに関する詳細は、『HP Performance Agent for UNIX ユーザー マニュアル』の第 2 章、「parm ファイル」セクションを参照してください。

## アラームの定義

パフォーマンスをモニタするためにアラームを使用する場合、**Performance Agent** の `alarmdef` ファイルにあるアラーム定義セット内でアラーム発生条件を指定します。**Performance Agent** を最初にインストールしたときに、`alarmdef` ファイルにデフォルトのアラーム定義セットが含まれています。このデフォルトの定義を使用するか、または必要な場合はカスタマイズすることが可能です。

アラームを定義する手順については、『**HP Performance Agent for UNIX ユーザー マニュアル**』の「パフォーマンス アラーム」の章を参照してください。この章では、アラーム定義の構文、アラームの機能、パフォーマンスをモニタするためのアラームの使用方法についても説明しています。

---

## 3 ドキュメント

この章では、以下のトピックに関する Performance Agent のオンラインおよび印刷可能なドキュメントについて記載します。

- Performance Agent ドキュメント一覧
- Web 上でのドキュメントの参照
- Adobe Acrobat ファイル

# Performance Agent ドキュメント一覧

Performance Agent ソフトウェアには、表示および印刷可能なファイルフォーマットで、標準の Performance Agent ドキュメント一式が付属しています。Adobe Acrobat フォーマット (\*.pdf) のドキュメントはオンラインで表示され、必要な場合は印刷できます。ASCII テキスト (\*.txt) のドキュメントも印刷可能です。vi などの UNIX テキスト エディタを使用すると、画面上でテキスト ファイルを表示できます。次の表で、ドキュメント、使用されているファイル名、およびオンラインのパスを示します。

**表 2 Performance Agent ドキュメント一覧**

ドキュメント	ファイル名	パス
HP Performance Agent for Linux システム インストール、設定ガイド	ovpainst.pdf	/opt/perf/ paperdocs/ovpa/C
HP Performance Agent for UNIX ユーザー マニュアル	ovpausers.pdf	/opt/perf/ paperdocs/ovpa/C
HP Performance Agent for UNIX データ ソース統合ガイド	ovpadsi.pdf	/opt/perf/ paperdocs/ovpa/C
HP Performance Agent for UNIX および GlancePlus トランザクション 追跡	tyt.pdf	/opt/perf/ paperdocs/arm/C
アプリケーション応答測定 API ガイド	arm2api.pdf	/opt/perf/ paperdocs/arm/C
HP Performance Agent for Linux Metrics Definitions	metlinux.txt	/opt/perf/ paperdocs/ovpa/C
HP Performance Agent metrics list by Data Class for all operating systems	mettable.txt	/opt/perf/ paperdocs/ovpa/C

## Web 上でのドキュメントの参照

ここにリストしている .pdf および .htm ドキュメントは、以下の HP ソフトウェア マニュアルの Web サイトでも参照できます。

**[http://ovweb.external.hp.com/lpe/doc\\_serv](http://ovweb.external.hp.com/lpe/doc_serv)**

製品リスト ボックスから **Performance Agent** を選択し、リリース バージョン、OS、およびマニュアルのタイトルを選択します。**[Open]** をクリックして、オンラインでドキュメントを表示します。または **[Download]** をクリックして、ご使用のコンピュータにファイルを配置します。

# Adobe Acrobat ファイル

Adobe Acrobat ファイルは、Acrobat 7.0 で作成されています。表示する場合は、Adobe Acrobat Reader バージョン 4.0 以降を使用してください。Web ブラウザに Acrobat Reader がインストールされていない場合は、Adobe の Web サイトからダウンロードできます。

**<http://www.adobe.com>**

Acrobat Reader でドキュメントを表示している間は、1 ページ、複数ページ、または全ページ分のドキュメントを印刷できます。

Linux で .PDF ファイルを表示するには、次のように入力します。

**`acroread filename.pdf`**



## 4 coda の設定

### はじめに

coda は、HP Operations Agent 用の軽量なデータ収集エージェントです。coda は Performance Agent のサブセットで、Performance Agent の通信経路として動作します。すべての設定パラメータは、coda 名前空間および coda.comm 名前空間内でコマンド ovconfchg を使用して設定します。

### coda 名前空間

次の表で、coda 名前空間の設定パラメータの一覧を示します。

coda 名前空間のパラメータ	説明	デフォルト値
DISABLE_PROSPECTOR	HP Operations Agent と Performance Agent が共存する環境において、coda データ ソースからのデータ収集を無効にするために使用します。	false
ENABLE_PROSPECTOR	Performance Agent のスタンドアロン環境において、coda データ ソースからのデータ収集を有効にするために使用します。	false
SSL_SECURITY	coda からの安全性の高い通信を有効にする場合に使用します。	NONE
RESPONSE_SIZE_LIMIT	coda によって割り当てられるメモリの最大容量を指定するために使用します。	104857600 (100 メガバイト)

## DISABLE\_PROSPECTOR

このオプションでは、HP Operations Agent および Performance Agent の両方がインストールされている場合の、coda を介したデータ収集について設定します。デフォルト値は false です。フォーマットは次のようになります。

**ovconfchg -namespace coda -set DISABLE\_PROSPECTOR <値>**

- true: coda は、coda データ ソース用のデータを収集しません。
- false: coda は、coda データ ソース用のデータを収集します。

## ENABLE\_PROSPECTOR

このオプションでは、Performance Agent のみがインストールされている場合の、coda を介したデータ収集について設定します。デフォルト値は false です。このパラメータは、HP Operations Agent および Performance Agent の両方がインストールされている場合には無視されます。フォーマットは次のようになります。

**ovconfchg -namespace coda -set ENABLE\_PROSPECTOR < 値 >**

- true: coda は、coda データ ソース用のデータを収集します。
- false: coda は、coda データ ソース用のデータを収集しません。



coda の Prospector が有効かどうかを確認するには、次のコマンドを入力します。

**ovcodautl -dumpds coda**

## SSL\_SECURITY

このオプションは、coda を介した安全性の高い通信を有効にします。デフォルト値は NONE です。フォーマットは次のようになります。

**ovconfchg -namespace coda -set SSL\_SECURITY < 値 >**

- NONE: coda は、ローカルクライアントまたはリモートクライアントに対して SSL 接続を要求しません。
- REMOTE: coda は、すべてのリモート接続に対して SSL を要求します。
- ALL: coda は、すべての接続 (ローカルおよびリモート) に対して SSL を要求します。



このパラメータは、システムに証明書が存在する場合にのみ使用します。証明書は、HP Operations Agent 8.x がシステムにインストールされている場合にのみインストールされます。

## RESPONSE\_SIZE\_LIMIT

このオプションは、coda デーモンによってクエリ応答に割り当てられるメモリの最大容量を指定します。デフォルト値は、104857600 (100 メガバイト ) です。フォーマットは次のようになります。

```
ovconfchg -namespace coda -set RESPONSE_SIZE_LIMIT <値>
```



指定した制限を超えた場合は、次のエラーメッセージが表示されます。

```
Response exceeds memory limits, use several smaller requests
```

## coda 通信名前空間 (coda.comm)

次の表で、coda.comm 名前空間の設定パラメータの一覧を示します。

coda.comm 名前空間のパラメータ	説明	デフォルト値
SERVER_PORT	ポートを設定するために使用します。	0
SERVER_BIND_ADDR	サーバー ポートのバインド アドレスを指定するために使用します。	INADDR_ANY
LOG_SERVER_ACCESS	true に設定されている場合、coda はサーバーへのアクセスをすべて記録します。記録される情報には、送信側 IP アドレス、要求された HTTP アドレス、要求された HTTP メソッド、および応答ステータスが含まれます。	false

## SERVER\_PORT

coda が使用するデフォルト ポートを設定できます。ポートのデフォルト値は 0 です。ポートが 0 に設定されている場合、オペレーティングシステムによって、空いている最初のポート番号が割り当てられます。ovconfchg ツールを使用して、Performance Agent システムでのポート設定を変更します。以下のコマンドを入力します。

```
ovconfchg -namespace coda.comm -set SERVER_PORT <ポート番号>
```

## SERVER\_BIND\_ADDR

このオプションでは、サーバー ポートのバインド アドレスを指定します。この値が localhost に設定されると、coda サーバーへの通信はすべて ovbbccb を通して行われます。フォーマットは次のようになります。

```
ovconfchg -namespace coda.comm -set SERVER_BIND_ADDR <バインドアドレス>
```

## LOG\_SERVER\_ACCESS

このオプションを使用すると、サーバーへのアクセスを有効 / 無効にできます。このオプションが true に設定されている場合、coda はサーバーへのアクセスをすべて記録します。記録される情報には、送信側 IP アドレス、要求された HTTP アドレス、要求された HTTP メソッドおよび応答ステータスが含まれます。通常、この値は変更しません。

```
ovconfchg -namespace coda.comm -set LOG_SERVER_ACCESS <値>
```

## 単一ポート通信

これは、Performance Agent 4.70 のデフォルトの通信方式です。coda で単一ポート通信を有効にする場合、次のオプションを使用します。

```
ovconfchg -ns coda -set SSL_SECURITY REMOTE/ALL
ovconfchg -ns coda.comm -set SERVER_BIND_ADDR localhost
```

## 複数ポート通信

coda で複数ポート通信を有効にする場合、次のオプションを使用します。

```
ovconfchg -ns coda.comm -set SERVER_BIND_ADDR ""
```

coda で単一ポート通信と複数ポート通信のどちらが有効かを確認するには2つの方法があります。

- 1 ポートが単一ポート通信または複数ポート通信のどちらで使用されているかを、ローカルホストマシンから確認するには、次のオプションを使用します。

### **bbcutil -reg**

単一ポート通信の場合、以下のように表示されます。

```
BasePath=/Hewlett-Packard/OpenView/Coda/
Protocol=HTTPS
BindAddress=localhost
Port=59814
Authentication=NONE
```

返された値が localhost または 127.0.0.1 の場合は、単一ポート通信です。

複数ポート通信の場合、以下のように表示されます。

```
BasePath=/Hewlett-Packard/OpenView/Coda/
Protocol=HTTPS
BindAddress=ANY
Port=381
Authentication=NONE
```

返された値が localhost 以外の場合は、複数ポート通信です。



**bbcutil** は、<インストールディレクトリ>/binディレクトリにあります。

- 2 ポートが単一ポート通信または複数ポート通信のどちらで使用されているかを、ローカル ホスト以外のシステムから確認するには、次のオプションを使用します。

```
ovcodautil -n <ホスト名> -ping
```

次に例を示します。

```
ovcodautil -n ovphpt4 -ping
```

単一ポート通信の場合、以下のように表示されます。

```
Ping of 'OvBbcCb' at: 'http://ovphpt4:383/Hewlett-Packard/OpenView/BBC/ping' successful
```

```
Ping of 'Coda' at: 'http://ovphpt4:383/Hewlett-Packard/OpenView/Coda/' successful
```

複数ポート通信の場合、以下のように表示されます。

```
ovcodautil -n ovphpt4 -ping
```

```
Ping of 'OvBbcCb' at: 'http://ovphpt4:383/Hewlett-Packard/OpenView/BBC/ping' successful
```

```
Ping of 'Coda' at: 'http://ovphpt4:62581/Hewlett-Packard/OpenView/Coda/' successful
```

複数ポート通信の場合は、2つの出力でポート番号が異なります。



coda クライアントの場合、coda.comm 名前空間のすべてのパラメータは、bbc.http 名前空間に定義されているパラメータより優先されます。

## 通信ブローカー名前空間 (bbc.cb)

次の表で、bbc.cb 名前空間の設定パラメータの一覧を示します。

bbc.cb 名前空間のパラメータ	説明	デフォルト値
SERVER_PORT	ポートを設定するために使用します。	383
SERVER_BIND_ADDR	サーバー ポート のバインド アドレスを指定するために使用します。	INADDR_ANY

### SERVER\_PORT

通信ブローカーが使用するデフォルト ポートを設定できます。ポートのデフォルト値は 383 です。ovconfchg ツールを使用して、Performance Agent システムでのポート設定を変更します。フォーマットは次のようになります。

```
ovconfchg -namespace bbc.cb -set SERVER_PORT <ポート番号>
```



通信ブローカー ポート名前空間 (bbc.cb.ports) にすでにポートが定義されている場合は、そのポートがデフォルト ポートとして割り当てられ、SERVER\_PORT の値が無効となります。

### SERVER\_BIND\_ADDR

このオプションでは、サーバー ポートのバインド アドレスを指定します。フォーマットは次のようになります。

```
ovconfchg -namespace bbc.cb -set SERVER_BIND_ADDR <バインドアドレス>
```



## 通信ブローカー ポート名前空間 (bbc.cb.ports)

次の表で、bbc.cb.ports 名前空間の設定パラメータの一覧を示します。

bbc.cb.ports 名前空間 のパラメータ	説明	デフォルト値
PORTS	<p>ネットワーク内で、ホスト上のアプリケーションが通信する可能性があるすべての通信ブローカー用のポートのリストを定義するために使用します。</p> <p>クライアントアプリケーションはこのリストを対象ポートとして使用し、ネットワーク内の通信ブローカーと通信を行います。このポート設定内のエントリとホスト名が一致する場合は、ローカルの通信ブローカーポートが通信用ポートとして設定されます。</p>	ポートの値のデフォルト設定はありません。

## PORTS

この設定パラメータはすべてのノードで同じ値に設定します。特定のホスト上の通信ブローカーのポート番号を変更する場合は、パラメータにホスト名を追加します。

次に例を示します。

```
name.hp.com:8000
```

アスタリクをワイルドカードとして使用すると、ネットワーク全体を指定できます。

```
*.hp.com:8001
```

ホスト名のリスト内のエントリを区切るには、コンマまたはセミコロンを使用します。

次に例を示します。

```
name.hp.com:8000, *.hp.com:8001
```

この例では、末尾が `hp.com` のホスト名の BBC 通信ブローカーがポート 8001 を使用するよう設定しています。ただし、ホスト名が「name」の場合はポート 8000 を使用します。その他の場合はすべてデフォルトのポート 383 を使用します。

また、IP アドレスとアスタリク (\*) を使用してホストを指定することもできます。

次に例を示します。

```
15.0.0.1:8002, 15.*.*.*:8003
```

ポートを設定するには、次のコマンドを実行します。

```
ovconfchg -namespace bbc.cb.ports -set PORTS <ポート番号>
```

## HTTP 名前空間 (bbc.http)

次の表で、bbc.http 名前空間の設定パラメータの一覧を示します。

bbc.http 名前空間の パラメータ	説明	デフォルト値
RESPONSE_TIMEOUT	応答を待機する最大秒数を指定するために使用します。	300
CLIENT_PORT	クライアント要求のバインド ポートを指定します。	0
PROXY	指定したホスト名に使用するプロキシおよびポートを指定します。	プロキシの値のデフォルト設定はありません。

### RESPONSE\_TIMEOUT

このオプションは、応答を待機する最大秒数を指定するために使用します。デフォルト値は 300 です。フォーマットは次のようになります。

```
ovconfchg -namespace bbc.http -set RESPONSE_TIMEOUT < 値 >
```

### CLIENT\_PORT

このオプションでは、クライアント要求のバインド ポートを指定します。デフォルト値はポート 0 です。オペレーティングシステムによって、空いている最初のポートが割り当てられます。ローカルホストへの要求の場合、このパラメータは無視されます。



Windows システムでポートを再利用する際、すぐにはポートが開放されないため、このパラメータには大きめの値を設定します。

## PROXY

指定したホスト名に使用するプロキシおよびポートを指定します。

フォーマットは次のようになります。

**proxy:port+ (a) - (b) ; proxy2:port2+ (a) - (b) ; ... ;**

この例で、変数 *a* および変数 *b* は、プロキシに適用するホスト名をコンマまたはセミコロンで区切った形式のリストです。

a: プロキシを使用します

b: プロキシを使用しません

最初に適合したプロキシが選択されます。

また、ホスト名の代わりに IP アドレスを使用できます。たとえば、`15.*.*.*` も、正しい数のピリオドおよびコロンが指定されているため、有効となります。

# 用語集

## **alarmdef ファイル**

アラームの状態を特定するアラーム定義を含むファイルです。

## **coda デーモン**

収集したデータをアラーム ジェネレータや scopeux ログ ファイルまたは DSI ログ ファイルを含むデータ ソースの分析製品に提供するデーモンです。coda は detasources 設定ファイルにリストされているデータ ソースから、データを読み込みます。

## **datasources ファイル**

/var/opt/OV/conf/perf/ ディレクトリにある設定ファイルです。ファイル内の各エントリは単一のログ ファイルセットから成る scopeux または DSI データ ソースを示しています。「perflbd.rc」、「coda」、および「**データ ソース**」も参照してください。

## **datasources 設定ファイル**

/var/opt/OV/conf/perf/ ディレクトリにある設定ファイルです。ファイル内の各エントリは単一のログ ファイルセットから成る scopeux または DSI データ ソースを示しています。

## **default.txt**

HP ソフトウェア アプリケーションで通信パラメータをカスタマイズするために使用する通信設定ファイルです。

## **DSI**

「**データ ソース統合 (DIS)**」を参照してください。

## **DSI ログ ファイル**

Performance Agent の DSI プログラムによって作成される、自己記述型のデータを含むログ ファイルです。

### **extract**

Performance Agent のプログラムの 1 つで、生のログ ファイルまたは事前に抽出したログ ファイルからデータを抽出 (コピー) し、抽出ログ ファイルに書き込みます。また、分析プログラムで使用するために、データを書き出すことができます。

### **Glance**

Glance (GlancePlus) は、ユーザー端末やワークステーションに現在のパフォーマンス データを直接表示するオンラインの診断ツールです。システムのパフォーマンス障害が発生した場合は、障害の識別と解決の支援ツールとして機能します。

### **logappl**

ユーザーが定義した各アプリケーションのプロセスの測定値を含む生ログ ファイルです。

### **logdev**

ディスクや netif などの個別デバイスのパフォーマンスの測定値を含む生ログ ファイルです。

### **logglob**

システム全体の作業負荷、つまりグローバルな作業負荷の測定値を含む生ログ ファイルです。

### **logindx**

他のログ ファイルに含まれるデータにアクセスする際に必要となる追加情報を含む生ログ ファイルです。

### **logproc**

選択された「対象プロセス」の測定値を含む生ログ ファイルです。プロセスが初めて作成されたとき、プロセスが終了したとき、および CPU 使用量、ディスク使用量、応答時間などに対してユーザーが定義したしきい値を超えたときに、プロセスは「対象プロセス」となります。

## logtran

トランザクションデータの測定値を含む生ログファイルです。

## midaemon

トレースデータを Measurement Interface カウンタデータに変換し、メモリベースの MI パフォーマンス データベースを使用してカウンタを保持する Performance Agent プログラムです。このデータベースは、scopeux のようなコレクタプログラムによりアクセスされます。

## ovbbccb

ovc で制御される通信をベースとした HTTP(S) 用の Operations Manager 通信ブローカーです。「**coda**」、および「**ovc**」も参照してください。

## ovc

Operations Manager の制御プロセスおよび監視プロセスです。スタンドアロン Performance Agent のインストールでは、ovc によって coda および ovbbccb が監視および制御されます。Operations Manager for UNIX 8.x エージェントがインストールされたシステムに Performance Agent をインストールする場合も、ovc によって Operations Manager for UNIX 8.x プロセスが監視および制御されます。「**coda**」、および「**ovbbccb**」も参照してください。

## ovpa スクリプト

scopeux データ コレクタ、アラーム ジェネレータ、ttd、midaemon、および coda などの Performance Agent のプロセスを開始、終了、再開するオプションを含んだ Performance Agent スクリプトです。ovpa の man ページも参照してください。

## parm ファイル

データ収集方法をカスタマイズするために、scopeux により使用されるパラメータを含む Performance Agent のファイルです。

## perflbd.rc

/var/opt/perf/ ディレクトリにある設定ファイルです。このファイルは、datasources ファイルへのシンボリックリンクとして管理されます。「**datasources ファイル**」も参照してください。

## Performance Manager

マルチベンダー分散型ネットワーク システムの統合型パフォーマンス管理を提供します。1 台のワークステーションを使用して、数十から数千のノード サイズに及ぶネットワーク上の環境パフォーマンスをモニタします。

### perfstat

システムにインストールされたパフォーマンス プロセスの全ステータスを表示するプログラムです。

### run ファイル

scopeux プロセスが動作中であることを示すために、scopeux コレクタによって作成されます。run ファイルを削除すると scopeux は終了します。

### scopeux

パフォーマンス データを収集した後、分析または記録用として生ログ ファイルにデータを書き込む (記録する) Performance Agent データ収集プログラムです。「**生ログ ファイル**」も参照してください。

### scopeux ログ ファイル

「**生ログ ファイル**」を参照してください。

### status.scope ファイル

ステータス、データの不整合、およびエラーを記録するために、scopeux コレクタによって作成されます。

### ttd.conf

トランザクション名、パフォーマンス分布範囲、サービス レベルの目標など、各トランザクションおよび追跡される情報を定義するトランザクション設定ファイルです。

### utility

Performance Agent のプログラムの 1 つで、生ログ ファイルと抽出ログ ファイルのオープン、リサイズ (サイズ変更)、スキャン、およびレポート作成を可能にします。このプログラムを使用すると、parm ファイルと alarmdef ファイルの構文が確認でき、過去のログ ファイル データのアラーム情報が取得できます。



## アプリケーション

関連のプロセスまたはプログラム ファイルのユーザー定義グループです。アプリケーションは、パフォーマンス ソフトウェアがパフォーマンス メトリックを収集し、プロセスおよびプログラムのアクティビティを一括して報告できるように定義されます。

## アプリケーション ログ ファイル

「logappl」を参照してください。

## アラーム

パフォーマンスがユーザー指定のアラームの基準に一致した時間、またはその基準を超えた時間を意味します。アラーム情報は、Performance Manager などの分析システムや Operations Manager に送信されます。utility プログラムを使用すると、過去のデータ ログ ファイルでアラームを識別できます。

## アラーム ジェネレータ

アラーム情報の通信を処理します。これは、perfalarm および agdb データベースで構成されます。agdb データベースは、アラームが送信される Performance Agent 分析ノード (存在する場合) のリスト、およびアラーム情報が送信される時間と場所を定義するために必要な各種の on または off フラグで構成されます。

## グローバル

システム全体を意味する修飾子です。

## グローバル ログ ファイル

「logglob」を参照してください。

## システム識別番号

システムを識別する文字列です。デフォルトは `uname -n` で返されるホスト名です。

## 対象プロセス

プロセスが初めて作成されたとき、プロセスが終了したとき、および CPU 使用量、ディスク使用量、応答時間などに対してユーザーが定義したしきい値を超えたときに、プロセスは「対象プロセス」となります。

## 抽出ログ ファイル

extract プログラムによって作成されるログ ファイルです。ユーザーが選択したデータの範囲とデータ型を含みます。抽出ログ ファイルは、ワークステーションの分析ツール、Performance Agent により最適アクセスが可能になるようにフォーマットされています。このファイルのフォーマットは、extract プログラムと utility プログラムへの入力に最適であり、パフォーマンス データのアーカイブにも適しています。

## データ ソース

単一の scopeux または DSI ログ ファイル セット内の 1 つ以上のデータ クラスで構成されます。たとえば、Performance Agent の SCOPE データ ソースは、グローバルなデータから成る scopeux ログ ファイル セットです。「**datasources ファイル**」も参照してください。

## データ ソース統合 (DSI)

Performance Agent がアプリケーション、データベース、ネットワーク、他のオペレーティング システムなどの外部ソースからデータを受信し、記録して、アラームを検出することを可能にする技術です。

## デバイス

システムに接続されている入出力デバイスです。ディスクドライブ、テープドライブ、DVD-ROM ドライブ、プリンタ、ユーザー端末などは共通デバイスです。

## デバイス ログ ファイル

「logdev」を参照してください。

## トランザクション追跡

Performance Agent で使用されている技術で、情報技術 (IT) のリソース マネージャが、ビジネス アプリケーション トランザクションのエンドツーエンドの応答時間を測定できるようにします。

## トランザクション ログ ファイル

「logtran」を参照してください。

## 生ログ ファイル

scopeux が収集するシステム データの要約された測定値を含むファイルです。「logappl」、「logproc」、「logdev」、「logtran」、および「logindx」を参照してください。

## プロセス

プログラム ファイルを実行します。対話型のユーザー (normal、nice または realtime のいずれかの優先順位で実行中のプロセス ) またはオペレーティング システムのプロセスを示します。

## プロセス ログ ファイル

「logproc」を参照してください。

## リアルタイム

イベントが発生する実際の時間です。

## リサイズ

utility プログラムの resize コマンドを使用して、ログ ファイル全体のサイズを変更します。

## ログ ファイル セット

単一のソースから収集されたデータから成るファイルの集まりです。



# 索引

## A

authip ファイル, 31  
    フォーマット, 32  
    例, 32

## C

coda.log ファイル, 34  
coda デーモン, 33, 34

## D

datasources 設定ファイル, 33  
    フォーマット, 33  
DSI データ ソース, 33

## E

extract プログラム, 33

## O

Operations Manager  
    Performance Agent のインストールに使用,  
    14  
ovpa.install スクリプト, 13  
ovpa restart スクリプト, 20  
OVPA\_START\_COMMAND 変数, 21  
OVPA\_START 変数, 21  
ovpa stop スクリプト, 12  
ovpa スクリプト  
    ovpa start, 20

## P

parm ファイル, 35  
Performance Agent  
    インストール, 12  
    起動, 20  
    起動と終了, 20  
    削除, 17  
    終了, 11  
    スクリプト, 20  
    ステータス ファイル, 22  
Performance Agent の起動と終了, 20  
Performance Agent の削除, 17  
Performance Agent の終了, 11  
perfstat コマンド, 11

## S

scopeux  
    起動, 20  
    データ ソース, 33  
scopeux ログ ファイル セット, 33  
SCOPE デフォルト データ ソース, 33

## あ

アラーム ジェネレータ、開始, 20

## い

印刷可能なドキュメント ファイル, 38

インストール  
Performance Agent, 12  
手順, 11  
要件, 9

インストールの手順  
Operations Manager を使用する Performance Agent のインストール, 14

インストール前にプロセスを終了, 11

インストール要件  
ハードウェア, 9

## か

環境変数, 21

## き

起動  
Performance Agent, 20  
scopeux, 20

## く

クライアント認証, 31

## す

スクリプト  
ovpa.install, 13  
ovpa.remove, 17  
ovpa restart alarm, 20  
ovpa stop, 20

ステータスの, 22

ステータス ファイル, 22  
coda.log, 22  
status.mi, 22  
status.perfalarm, 22  
status.scope, 22  
status.ttd, 22

## せ

設定, 33  
データ ソース, 33

## て

ディスク スペース、要件, 10

ディレクトリのサンプル  
README, 22

データ ソース, 33  
DSI, 33  
SCOPE, 33  
scopeux, 33  
SCOPE デフォルト データ ソース, 33  
削除, 34

データ ソースの削除, 34

## と

ドキュメント, 37  
Adobe Acrobat ファイル, 40  
Web に掲載されているドキュメント, 38

## ふ

ファイアウォール  
Performance Agent 通信設定の概要, 24  
ファイアウォールを介した通信, 23  
複数の IP アドレスを持つシステム, 29

ファイアウォール、通信, 23

ファイル  
Adobe Acrobat, 40  
authip, 31  
coda.log, 34  
datasources, 33  
parm, 35  
status.scope, 22



變數, 21

OVPA\_START, 21

OVPA\_START\_COMMAND, 21

環境, 21

