

Symantec™ ApplicationHA 汎 用エージェント設定ガイド

Oracle VM Server for SPARC 上の
Solaris

6.0

Symantec™ ApplicationHA 汎用エージェント設定ガイド

このマニュアルで説明するソフトウェアは、使用許諾契約に基づいて提供され、その内容に同意する場合にのみ使用することができます。

Agent version: 6.0.000.000

マニュアルバージョン: 6.0.000.000.0

法的通知と登録商標

Copyright © 2012 Symantec Corporation. All rights reserved.

Symantec、Symantec ロゴ、Veritas、Veritas Storage Foundation、CommandCentral、NetBackup、Enterprise Vault、LiveUpdate は、Symantec Corporation または同社の米国およびその他の国における関連会社の商標または登録商標です。その他の会社名、製品名は各社の登録商標または商標です。

本書に記載の製品は、ライセンスに基づいて配布され、使用、コピー、配布、逆コンパイル、リバースエンジニアリングはそのライセンスによって制限されます。本書のいかなる部分も、Symantec Corporation とそのライセンサーの書面による事前の許可なく、いかなる形式、方法であっても複製することはできません。

本書は「現状有姿のまま」提供され、商品性、特定目的への適合性、不侵害の黙示的な保証を含む、すべての明示的または黙示的な条件、表明、保証は、この免責が法的に無効であるとみなされない限り、免責されるものとします。Symantec Corporation は、本書の供給、性能、使用に関する付随的または間接的損害に対して責任を負わないものとします。本書に記載の情報は、予告なく変更される場合があります。

ライセンス対象ソフトウェアと関連書類は、FAR 12.212 の規定によって商業用コンピュータソフトウェアとみなされ、場合に応じて、FAR 52.227-19「Commercial Computer Software - Restricted Rights」、DFARS 227.7202「Rights in Commercial Computer Software or Commercial Computer Software Documentation」、その後継規制の規定により制限された権利の対象となります。米国政府によるライセンス対象ソフトウェアと関連書類の使用、修正、複製のリリース、実演、表示または開示は、本使用許諾契約の条項に従ってのみ行われるものとします。

弊社製品に関して、当資料で明示的に禁止、あるいは否定されていない利用形態およびシステム構成などについて、これを包括的かつ暗黙的に保証するものではありません。また、弊社製品が稼動するシステムの整合性や処理性能に関しても、これを暗黙的に保証するものではありません。

これらの保証がない状態で、弊社製品の導入、稼動、展開した結果として直接的、あるいは間接的に発生した損害等についてこれが補償されることはありません。製品の導入、稼動、展開にあたっては、お客様の利用目的に合致することを事前に十分に検証および確認いただく前提で、計画および準備をお願いします。

目次

第 1 章	Symantec ApplicationHA 用の汎用アプリケーションエージェント	5
	汎用エージェントについて	5
	エージェント機能	6
	状態の定義	7
第 2 章	汎用 Application エージェントの設定	9
	ApplicationHA でのアプリケーション監視の設定について	9
	カスタムアプリケーション のアプリケーション監視を設定する前に	10
	[ApplicationHA] タブへのアクセス	11
	カスタムアプリケーションのアプリケーション監視の設定	11
付録 A	リソースタイプの定義	17
	リソースタイプの定義	17
	属性	18
付録 B	設定例	21
	カスタムプロセスの設定例	21
付録 C	カスタムアプリケーションのサンプルスクリプト	25
	カスタムアプリケーションを開始、停止、監視するためのサンプルスクリプト	25
付録 D	カスタム監視プログラム	27
	複数のプロセスを監視するためのカスタム管理プログラムの作成	27
	PID ファイルを使って複数のプロセスを監視するためのカスタム監視プログラムの作成	28

Symantec ApplicationHA 用の汎用アプリケーションエージェント

この章では以下の項目について説明しています。

- [汎用エージェントについて](#)

汎用エージェントについて

汎用エージェントはカスタムアプリケーションをオンラインまたはオフラインにし、サービスの状態を監視します。様々なプログラムのオンライン、オフライン、監視ルーチンに対して異なる実行可能ファイルを指定するには、このエージェントを使います。実行ファイルはゲストドメインでローカルに存在する必要があります。ApplicationHA によってデフォルトでサポートされていないアプリケーションに対して高可用性を提供するためにこのエージェントを使うことができます。

サポート対象のアプリケーションのリストについては、『Symantec ApplicationHA インストールガイド』を参照してください。

アプリケーションはルートのデフォルトコンテキストで実行されます。

アプリケーションは次の方法で管理できます。

- 監視プログラムを使用する
- プロセスのリストを指定する
- プロセス ID ファイルのリストを指定する
- 上記の方法を任意に組み合わせる

メモ: ApplicationHA Custom Application ウィザードは監視プログラムのみを設定します。プロセスまたはプロセスID ファイルのリストを指定する方法については、p.28の「PIDファイルを使って複数のプロセスを監視するためのカスタム監視プログラムの作成」を参照してください。

エージェント機能

Online **StartProgram** 属性の値で指定したスクリプトまたはコマンドを実行します。指定したユーザーコンテキストにおいて、指定したパラメータでコマンドを実行します。

リソースをオンラインにするため、エージェントの機能は次のコマンドを実行します。

```
su - user -c command_to_online_resource
```

Offline **StopProgram** 属性の値で指定したスクリプトまたはコマンドを実行します。指定したユーザーコンテキストにおいて、指定したパラメータでコマンドを実行します。

リソースをオフラインにするため、エージェントの機能は次のコマンドを実行します。

```
su - user -c command_to_offline_resource
```

Monitor **MonitorProgram** 属性が指定されている場合、エージェントはユーザーが指定した環境でユーザー定義の **MonitorProgram** を実行します。 **PidFiles** 属性が指定されている場合は、ルーチンがリストに表示された各ファイル内にあるプロセスIDが実行中であることを確認します。 **MonitorProcesses** 属性が指定されている場合は、ユーザーの指定した環境で、ルーチンがリストに示された各プロセスが実行中かどうかを確認します。

これらの属性 (**MonitorProgram**、**PidFiles**、または **MonitorProcesses**) の任意の組み合わせを使って、アプリケーションを監視します。

PidFiles または **MonitorProcesses** で指定されているプロセスが実行していないと判断された場合、**monitor** は **OFFLINE** を返します。プロセスが異常終了した場合は、**monitor** は **OFFLINE** を返します。

リソースを監視するため、エージェントの機能は次のコマンドを実行します。

```
su - user -c command_to_monitor_resource
```

Clean PidFiles または MonitorProcesses で指定した処理を終了します。User 属性で指定されたユーザー名を使って実行されているプロセス(MonitorProcesses 属性で指定されている)のみが強制終了されます。アプリケーションを強制的に停止するようにプログラムを定義した場合、エージェントはプログラムを実行しません。

リソースを強制的に停止するため、エージェントの機能は次のコマンドを実行します。

```
su - user -c command_to_clean_resource
```

状態の定義

ONLINE PidFiles および MonitorProcesses 属性で指定されているすべてのプロセスが実行中であること、および MonitorProgram が ONLINE を返すことを示します。

OFFLINE PidFiles 属性または MonitorProcesses で指定されているプロセスが少なくとも 1 つが実行していないこと、または MonitorProgram が OFFLINE を返すことを示します。

UNKNOWN アプリケーションが判別不能な状態であるか、設定が無効であることを示します。

汎用 Application エージェントの設定

この章では以下の項目について説明しています。

- [ApplicationHA でのアプリケーション監視の設定について](#)
- [カスタムアプリケーションのアプリケーション監視を設定する前に](#)
- [\[ApplicationHA\]タブへのアクセス](#)
- [カスタムアプリケーションのアプリケーション監視の設定](#)

ApplicationHA でのアプリケーション監視の設定について

この章では、仮想化環境において ApplicationHA でアプリケーション監視を設定する手順について説明します。

続行する前に次の点を考慮してください。

- **Application Monitoring Configuration Wizard** を使って、ゲストドメインにアプリケーション監視を設定します。
- **Application Monitoring Configuration Wizard** は Veritas Operations Manager (VOM) コンソールの [ApplicationHA] タブの [Configure Application Monitoring] をクリックすると起動されます。
- このリリースでは、ウィザードで監視を設定できるアプリケーションはゲストドメインごとに 1 つだけです。
ウィザードを使って別のアプリケーションを設定するには、最初に既存のアプリケーションの監視を設定解除する必要があります。

- ウィザードを使ってアプリケーションの監視を設定した後、Veritas Cluster Server (VCS) コマンドを使って同じゲストドメインに存在する他のアプリケーションの監視を設定できます。
詳しくは次の TechNote を参照してください。
<http://www.symantec.com/docs/TECH159846>
- カスタムアプリケーションを監視するように設定した後、アプリケーションの別のインスタンスを作成した場合、これらの新しいコンポーネントは既存の設定の一部として監視されません。
このような場合は、最初に既存の設定を設定解除してから、ウィザードを使ってアプリケーションを再設定してください。それから、すべてのインスタンスを監視対象として選択できます。

カスタムアプリケーションのアプリケーション監視を設定する前に

ゲストドメインで カスタムアプリケーション のアプリケーション監視を設定する前に次のタスクを完了してください。

- Veritas Operations Manager (VOM) 管理サーバーをインストールし、VOM 管理サーバーに VOM add-on for ApplicationHA をインストールします。VOM の使用について詳しくは、『Symantec ApplicationHA ユーザーズガイド』を参照してください。[ApplicationHA] タブへのアクセスについて詳しくは、p.11 の「[ApplicationHA] タブへのアクセス」を参照してください。
- 監視する必要があるゲストドメインに ApplicationHA ゲストコンポーネントをインストールします。
- ApplicationHA を割り当てます - アプリケーション監視を設定するゲストドメインでログオンしたユーザーにアプリケーション監視 (Admin) 権限を設定します。
- ゲストドメインで監視するアプリケーションとそれに関連するコンポーネントをインストールします。
- ファイアウォールを設定している場合は、ファイアウォールの設定で ApplicationHA のインストール、ウィザード、サービスによって使われるポートへのアクセスが許可されていることを確認します。
使われるポートとサービスのリストについては、『Symantec ApplicationHA インストールガイド』を参照してください。

[ApplicationHA]タブへのアクセス

Oracle VM Server for SPARC 環境で動作しているゲストドメイン上のアプリケーションを管理するためには、Veritas Operations Manager (VOM) コンソールの [ApplicationHA] タブにアクセスする必要があります。

[ApplicationHA] タブでは、次のような管理作業を実行できます。

- アプリケーションを起動する
- アプリケーションを停止する
- アプリケーション監視を設定する
- アプリケーション監視の設定を解除する
- アプリケーションのハートビートを有効にする
- アプリケーションのハートビートを無効にする
- メンテナンスモードを開始する
- メンテナンスモードを終了する

[ApplicationHA] タブにアクセスするには

- 1 Veritas Operations Manager コンソールで、[Manage]、[Servers]、[Hosts] の順に選択します。
- 2 左ペインの [License] リストボックスで、[ApplicationHA] チェックボックスを選択します。
- 3 右ペインで、管理操作を実行するゲストドメインをクリックします。
- 4 [ApplicationHA] タブをクリックします。

カスタムアプリケーションのアプリケーション監視の設定

ゲストドメインでカスタムアプリケーションの監視を設定するには、次の手順を実行してください。

カスタムアプリケーションのアプリケーション監視を設定するには

- 1 Veritas Operations Manager コンソールの [ApplicationHA] タブで、[Configure Application Monitoring] をクリックします。
これにより Application Monitoring Configuration Wizard が起動されます。
- 2 [Welcome] 画面の情報を確認し、[Next] をクリックします。
ウィザードはシステムのすべてのサポート対象アプリケーションをリストします。

- 3 [Custom Application]を選択し、[Next]をクリックします。
[Program Selection]画面が表示されます。
- 4 監視するアプリケーションコンポーネントを指定するには、[Add Component]をクリックします。
[Application Component Parameters]ダイアログボックスが表示されます。
- 5 アプリケーションの管理に使うプログラムスクリプトの名前とユーザー名を指定します。

メモ: ウィザードは指定されたコンポーネントの[Display Name]を自動的にポピュレートします。ただし、このフィールドの情報は編集できます。また、汎用エージェントを設定するゲストドメイン(ゲスト)で有効なクレデンシャルと適切な権限を持つユーザーを指定してください。そうでないと、アプリケーション監視は失敗することがあります。

- 6 [OK]をクリックします。
指定されたコンポーネントが[Program Selection]画面に表示されます。
- 7 監視するそのほかのコンポーネントを指定するには、手順 4 から手順 6 を繰り返します。そうでない場合は、[Next]をクリックします。
[Define Start Stop Order]画面が表示されます。画面にはこれまでに選択したコンポーネントのリストが表示されます。
- 8 [Parent Component]リストで、コンポーネントをクリックします。
手順 4 から手順 7 で指定したそのほかのコンポーネントが[Components]リストに表示されます。

Display Name	Start Program	Stop Program	Monitor Program	Force-Stop Program	User
cups_Program	/etc/init.d/cups start	/etc/init.d/cups stop	/etc/init.d/cups status	/etc/init.d/cups stop	root
crond_Program	/etc/init.d/crond start	/etc/init.d/crond stop	/etc/init.d/crond status	/etc/init.d/crond stop	root

9 指定された親コンポーネントとの依存関係を設定するには、[Components]リストでコンポーネントをクリックします。すべての親コンポーネントに対してこの手順を繰り返します。

10 [Configure]をクリックします。

ウィザードはアプリケーション監視の設定タスクを実行します。[ApplicationHA Configuration]画面に各タスクの状態が表示されます。

すべてのタスクが完了したら、[Next]をクリックします。

メモ: 設定タスクが失敗した場合は、[View Logs]をクリックして障害の詳細を調べます。

その後、ウィザードを再び実行してアプリケーションを設定する必要があります。

11 [Finish]をクリックしてウィザードを終了します。

これで、アプリケーション監視の設定は完了です。p.21の「[カスタムプロセスの設定例](#)」を参照してください。

- 12 Veritas Operations Manager コンソールでゲストドメイン上の設定されたアプリケーションの状態を表示するには、該当するゲストドメインに移動し、[ApplicationHA] タブをクリックします。

ApplicationHA ビューが表示されます。

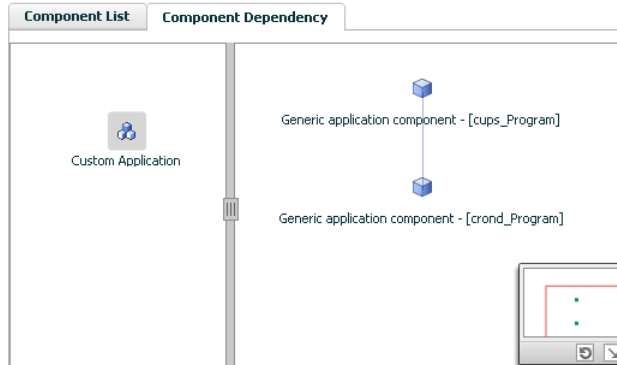
The screenshot shows the ApplicationHA console interface. At the top, it displays 'Applications: Custom Application' and 'Status: Online' with a refresh rate of 60 seconds. Below the status are several control buttons: 'Configure Application Monitoring', 'Unconfigure Application Monitoring', 'Enable Application Heartbeat', 'Disable Application Heartbeat', 'Start Application', 'Stop Application', 'Enter Maintenance Mode', and 'Exit Maintenance Mode'. The main area is divided into two tabs: 'Component List' (selected) and 'Component Dependency'. The 'Component List' tab shows two entries, both with green checkmarks and the status 'RUNNING': 'GenericApplication Instance [cups_Program] is RUNNING' and 'GenericApplication Instance [cron_d_Program] is RUNNING'. At the bottom, it shows 'ApplicationHA (Version 6.0.00000.419) | Symantec.' and a 'View log' link.

デフォルトでは、[Component List]タブが表示されます。このタブには、設定されたアプリケーションの各コンポーネントと各コンポーネントの状態の説明が表示されます。

Veritas Operations Manager を使ったアプリケーションの表示と管理について詳しくは、『Symantec ApplicationHA ユーザーズガイド』を参照してください。

- 13 監視されるアプリケーションのコンポーネントの依存関係を表示するには、**[Component Dependency]** タブをクリックします。

コンポーネントの依存関係のグラフが表示されます。



グラフは、選択したコンポーネントグループ（アプリケーションまたは相互に関連するコンポーネントのグループ）と、設定されたアプリケーションのそのコンポーネントの間の依存関係を示します。左ペインにはコンポーネントグループや設定されたアプリケーションが表示されます。右ペインには選択したコンポーネントグループまたはアプリケーションのコンポーネントが表示されます。

設定されたアプリケーションのコンポーネントの依存関係の表示について詳しくは、『Symantec ApplicationHA ユーザーズガイド』を参照してください。

リソースタイプの定義

この付録では以下の項目について説明しています。

- [リソースタイプの定義](#)
- [属性](#)

リソースタイプの定義

```
type Application (  
  static keylist SupportedActions = { "program.vfd",  
    "user.vfd", "cksum.vfd", getcksum, propcv }  
  static keylist RegList = { MonitorProcesses, User }  
  static str IMFRegList[] = { MonitorProcesses, User, PidFiles,  
    MonitorProgram }  
  static str ArgList[] = { User, StartProgram, StopProgram, CleanProgram,  
    MonitorProgram, PidFiles, State, IState, User,  
    MonitorProcesses, EnvFile, UseSUDash }  
  static int ContainerOpts{} = { RunInContainer=1, PassCInfo=0 }  
  str User = root  
  str StartProgram  
  str StopProgram  
  str CleanProgram  
  str MonitorProgra  
  str PidFiles[]  
  str MonitorProcesses[]  
  str EnvFile  
  boolean UseSUDash = 0  
)
```

属性

表 A-1 必須属性

必須属性	説明
StartProgram	<p>アプリケーションを開始する実行ファイル。絶対パスでこの実行ファイルを指定します。必要なコマンドライン引数を指定する場合は、実行ファイル名の後にスペースで区切って指定します。</p> <p>メモ: この文字列では、開始と終了の ({}) 波カッコの記号を使わないでください。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>例: <code>"/usr/sbin/sample_app start"</code></p>
StopProgram	<p>アプリケーションを停止する実行ファイル。絶対パスでこの実行ファイルを指定します。必要なコマンドライン引数を指定する場合は、実行ファイル名の後にスペースで区切って指定します。</p> <p>メモ: この文字列では、開始と終了の ({}) 波カッコの記号を使わないでください。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>例: <code>"/usr/sbin/sample_app stop"</code></p>
次のいずれか(1つ以上)の属性	p.18 の 表 A-2 を参照してください。
<ul style="list-style-type: none"> ■ MonitorProcesses ■ MonitorProgram ■ PidFiles 	

表 A-2 省略可能な属性

省略可能な属性	説明
CleanProgram	<p>アプリケーションを強制終了するための実行ファイル。絶対パスでこの実行ファイルを指定します。必要なコマンドライン引数を指定する場合は、実行ファイル名の後にスペースで区切って指定します。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>例: <code>"/usr/sbin/sample_app force-stop"</code></p>

省略可能な属性	説明
MonitorProcesses	<p>監視とクリーンの対象とするプロセスのリスト。各プロセス名は、実行ファイルの名前です。パスを指定して実行ファイルを起動する場合は、実行ファイルの名前を絶対パスで指定する必要があります。</p> <p>プロセス名は <code>ps -ef</code> コマンドでプロセスに表示される名前を指定します。</p> <p>データ形式と値の種類: 文字列 - ベクトル</p> <p>例: "nmbd"</p>
MonitorProgram	<p>アプリケーションを監視する実行ファイル。絶対パスでこの実行ファイルを指定します。必要なコマンドライン引数を指定する場合は、実行ファイル名の後にスペースで区切って指定します。</p> <p>MonitorProgram は次の値を返すことができます。OFFLINE 値は 100 です。ONLINE 値は信頼度に応じて 101 から 110 です。110 は信頼度が 100% であることを示します。その他の値の場合は UNKNOWN です。</p> <p>メモ: この文字列では、開始と終了の ({}) 波カッコの記号を使わないでください。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>httpd のような <code>init</code> プロセスは、特別な監視スクリプトを必要としません。<code>ApplicationHA</code> は <code>init</code> スクリプトの <code>status</code> オプションを監視に使います。</p>
PidFiles	<p>監視とクリーンの対象とするプロセスのプロセスID (PID) が含まれているPIDファイルのリスト。これらのファイルは、アプリケーションによって生成されます。各PIDファイルには、監視下にあるPIDが1つ含まれます。属性値には、絶対パスで各PIDファイルを指定します。</p> <p>プロセスIDは、プロセスの再起動時に変更される可能性があります。PIDファイルの更新に時間がかかると、エージェントの Monitor 機能から不正な結果が返される場合があります。誤った結果になった場合は、リソース定義の ToleranceLimit の値を大きくします。</p> <p>データ形式と値の種類: 文字列 - ベクトル</p>
User	<p>StartProgram、StopProgram、MonitorProgram、CleanProgram を実行するためのユーザー名。MonitorProcesses で指定されたプロセスは、この属性値で指定されたユーザーコンテキストで実行される必要があります。監視時に、プロセスが指定されたユーザーコンテキストで実行されていることを確認します。</p> <p>データ形式と値の種類: 文字列 - スカラー</p> <p>デフォルト: root</p>

他の属性について詳しくは、『Veritas Cluster Server 6.0 付属エージェントリファレンスガイド』を参照してください。

設定例

この付録では以下の項目について説明しています。

- [カスタムプロセスの設定例](#)

カスタムプロセスの設定例

このセクションでは、Symantec ApplicationHAを使って高可用性用のカスタムプロセスを設定する手順を説明します。

次のことを想定します。

- サンプルカスタムアプリケーションコンポーネント **HttpServer**
は、`/usr/local/apache2/bin/httpd start`、`/usr/local/apache2/bin/httpd stop`、`/usr/local/apache2/bin/httpd stop`、`/usr/local/apache2/bin/monitor_httpd`の各スクリプトの適切なオプションを使って、起動、停止、強制的に停止、監視できるものと想定します。monitor_httpd スクリプトはカスタムアプリケーションの **MonitorProgram** 属性と準拠するように作成されています。
- サンプルカスタムアプリケーションコンポーネント **MyComponent1** は、`startMyComponent1`、`stopMyComponent1`、`forcestopMyComponent1`、`monitorMyComponent1` スクリプトを使ってそれぞれ起動、停止、強制的に停止、監視できます。monitorMyComponent1 スクリプトはカスタムアプリケーションの **MonitorProgram** 属性と準拠するように作成されています。

詳しくは、p.18の「[属性](#)」を参照してください。

init プロセスのアプリケーション監視を設定するには

- 1 Veritas Operations Manager コンソールの [ApplicationHA] タブで、[Configure Application Monitoring] をクリックします。これにより Application Monitoring Configuration Wizard が起動されます。
- 2 [Welcome] 画面の情報を確認し、[Next] をクリックします。
ウィザードはシステムのすべてのサポート対象アプリケーションをリストします。
- 3 [Custom Application] を選択し、[Next] をクリックします。
[Program Selection] 画面が表示されます。
- 4 監視する httpd アプリケーションの詳細を指定するには、[Add Component] をクリックします。
[Application Component Parameters] ダイアログボックスが表示されます。

メモ: httpd のような init プロセスは、特別な監視スクリプトを必要としません。ApplicationHA は init スクリプトの status オプションを監視に使用します。ただし、このようなプロセスを監視するために独自のプログラムスクリプトを使うこともできます。

- 5 それぞれのフィールドに次の値を入力し、[OK] をクリックします。

Program to start the application	<code>/usr/local/apache2/bin/httpd -f /usr/local/apache2/conf/httpd.conf -k start</code>
Program to stop the application	<code>/usr/local/apache2/bin/httpd -f /usr/local/apache2/conf/httpd.conf -k stop</code>
Program to monitor the application	<code>/usr/local/apache2/bin/monitor_httpd</code>

- 6 監視するアプリケーションコンポーネントを指定するには、[Add Component] をクリックします。
[Application Component Parameters] ダイアログボックスが表示されます。

- 7 それぞれのフィールドに次の値を入力します。

Program to start the application	<code>/home/user1/myapplication/bin/startMyComponent1</code>
Program to stop the application	<code>/home/user1/myapplication/bin/stopMyComponent1</code>
Program to monitor the application	<code>/home/user1/myapplication/bin/monitorMyComponent1</code>
Program to forcibly stop the application	<code>/home/user1/myapplication/bin/forcestopMyComponent1</code>
User name to use when administering the application	<code>username</code> デフォルトユーザ名は「root」です

- 8 [OK]をクリックします。

指定されたコンポーネントが[Program Selection]画面に表示されます。

- 9 監視するそのほかのコンポーネントを指定するには、手順 6 から手順 8 を繰り返します。

httpd プログラムと MyApplication コンポーネントの関係を定義するには、[Next] をクリックします。[Define Start Stop Order]画面が表示されます。画面にはこれまでに選択したコンポーネントのリストが表示されます。

- 10 httpd プログラムを最初にオンラインにし、次に MyApplication プログラムをオンラインにするには、[Parent Component]リストで、[startMyComponent1_Program] をクリックします。

- 11 [Component]リストで、httpd_Program をクリックしてチェックを入れ、[Configure] をクリックします。

ウィザードはアプリケーション監視の設定タスクを実行します。[ApplicationHA Configuration]画面に各タスクの状態が表示されます。すべてのタスクが完了したら、[Next]をクリックします。

- 12 [Finish]をクリックしてウィザードを終了します。これで、アプリケーション監視の設定は完了です。

カスタムアプリケーションのサンプルスクリプト

この付録では以下の項目について説明しています。

- [カスタムアプリケーションを開始、停止、監視するためのサンプルスクリプト](#)

カスタムアプリケーションを開始、停止、監視するためのサンプルスクリプト

汎用エージェントでカスタムアプリケーションをオンラインにしたり、カスタムアプリケーションをオフラインにしたり、カスタムアプリケーションの状態を監視するための独自のスクリプトを作成できます。また、次のサンプルスクリプトを変更し、カスタムアプリケーションの開始、停止、監視に使うこともできます。

- カスタムアプリケーションを開始するサンプルスクリプト:

```
#!/bin/sh
touch /tmp/sampleapp # add any steps, if required
exit 0
```

カスタムアプリケーションの要件に合うように、サンプル開始スクリプトを修正できます。開始スクリプトを `startsampleapp` という名前で保存した場合、カスタムアプリケーションをオンラインにするために、エージェント機能は次のコマンドを実行します。

```
su - user -c startsampleapp
```

- カスタムアプリケーションを停止するサンプルスクリプト:

```
#!/bin/sh
rm -f /tmp/sampleapp # add any steps, if required
exit 0
```

カスタムアプリケーションの要件に合うように、サンプル停止スクリプトを修正できます。停止スクリプトを `stopsampleapp` という名前で保存した場合、カスタムアプリケーションを停止するために、エージェント機能は次のコマンドを実行します。

```
su - user -c stopsampleapp
```

メモ: 開始と停止のスクリプトの戻りコードの値は、0 である必要があります。他の戻りコードはサポートされません。

■ カスタムアプリケーションを監視するサンプルスクリプト:

```
#!/bin/sh
APPLICATION_IS_ONLINE=110
APPLICATION_IS_OFFLINE=100
if [ -f /tmp/sampleapp ] ; then # add any steps, if required
exit $APPLICATION_IS_ONLINE
else
exit $APPLICATION_IS_OFFLINE
fi
```

監視スクリプトを `monitorsampleapp` という名前で保存した場合、カスタムアプリケーションを監視するために、エージェント機能は次のコマンドを実行します。

```
su - user -c monitorsampleapp
```

カスタム監視プログラム

この付録では以下の項目について説明しています。

- 複数のプロセスを監視するためのカスタム管理プログラムの作成
- PID ファイルを使って複数のプロセスを監視するためのカスタム監視プログラムの作成

複数のプロセスを監視するためのカスタム管理プログラムの作成

Application Monitoring Configuration Wizard の[Custom Application]オプションでは、複数のプロセスを監視できません。ApplicationHA を使って複数のプロセスを監視するために使うことができる監視プログラムを作成するには次の手順を実行します。

プロセスパス名を使ってカスタム監視プログラムを作成するには

- 1 次のコマンドを実行して各プロセスが実行中かどうかを確認します。

```
ps -ef | grep ProcessName
```

- 2 すべてのプロセスが実行中の場合は、110 をリターンコードとして監視プログラムを終了します。

いずれかのプロセスが実行していない場合は、100 をリターンコードとして監視プログラムを終了します。

- 3 このプロセスをシェルスクリプトに保存し、実行権限をスクリプトに割り当てます。
- 4 Application Monitoring Configuration Wizard の[Application Component Parameters]ダイアログボックスで、[Monitor Program]フィールドにスクリプトの絶対パスを入力し、監視するプロセスのリストを続けて入力します。

PID ファイルを使って複数のプロセスを監視するための カスタム監視プログラムの作成

Application Monitoring Configuration Wizard の [Custom Application] オプションでは、PID ファイルを使って複数のプロセスを監視できません。ApplicationHA で PID ファイルを使って複数のプロセスを監視するために使うことができる監視プログラムを作成するには次の手順を実行します。

PID ファイルを使ってカスタム監視プログラムを作成するには

- 1 次のコマンドを実行して各 PID ファイルで指定されたプロセスが実行中であるかどうかを確認します。

```
ps -ef | grep ProcessID
```

ここで、*ProcessID* は PID ファイルの内容です。

- 2 すべてのプロセスが実行中の場合は、110 をリターンコードとして監視プログラムを終了します。

いずれかのプロセスが実行していない場合は、100 をリターンコードとして監視プログラムを終了します。

- 3 このプロセスをシェルスクリプトに保存し、実行権限をスクリプトに割り当てます。
- 4 Application Monitoring Configuration Wizard の [Application Component Parameters] ダイアログボックスで、[Monitor Program] フィールドにスクリプトの絶対パスを入力し、続けて PID ファイルのリストをスペースで区切って入力します。