

問8 プロビジョニングシステムの設計に関する次の記述を読んで、設問1~4に答えよ。

L社では、新規に仮想サーバのホスティングサービスを開始することになった。このサービスでは、利用者が任意のタイミングで仮想サーバの作成、起動、停止及び削除を行うことができるプロビジョニングの機能を提供する。仮想サーバのホスティングサービスのシステムの全体構成を図1に示す。

仮想サーバのホスティングサービスのシステムは、仮想サーバシステムと、仮想サーバホスティングシステムから構成される。

仮想サーバシステムには市販の製品を用いる。仮想サーバ管理APIは、仮想サーバシステムと他のシステムとの連携動作を実現するために用いるAPIである。仮想サーバホスティングシステムは、これを呼び出すことで、仮想サーバの制御を行う。

仮想サーバホスティングシステムは、管理ポータルサイトとプロビジョニングシステムから成り、これらはL社が新規に開発する。

管理ポータルサイトは、仮想サーバの作成、起動、停止及び削除を指示したり、現在の状態を確認したりする、利用者向けのポータルサイトである。

プロビジョニングシステムは、管理ポータルサイトからのメッセージに応じて仮想サーバ管理APIを呼び出すことで、仮想サーバの作成、起動、停止及び削除を行うシステムである。

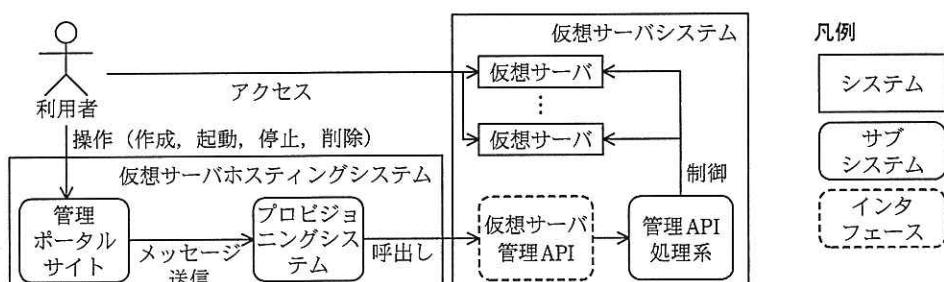


図1 仮想サーバのホスティングサービスのシステムの全体構成

〔仮想サーバ管理APIの仕様〕

仮想サーバ管理APIの仕様上、仮想サーバには、未作成、作成中、起動中、起動、停止中、停止、削除中及び削除済の状態がある。仮想サーバ管理APIの仕様を表1に

示す。

表1 仮想サーバ管理APIの仕様

| API名 | 機能 |
|--------------|---|
| CreateServer | 仮想サーバを作成し、作成したサーバを一意に識別するサーバIDを返す。 |
| StartServer | サーバIDを指定して呼び出すことで、仮想サーバを起動する。仮想サーバの状態が“停止”でなかった場合や、存在しないサーバIDが指定された場合はエラーになる。 |
| StopServer | サーバIDを指定して呼び出すことで、仮想サーバを停止する。仮想サーバの状態が“起動”でなかった場合や、存在しないサーバIDが指定された場合はエラーになる。 |
| DeleteServer | サーバIDを指定して呼び出すことで、仮想サーバを削除する。仮想サーバの状態が“停止”でなかった場合や、存在しないサーバIDが指定された場合はエラーになる。 |

仮想サーバ管理APIは、呼び出されたAPIの処理が完了した時点で応答を返す。例えばCreateServerを実行すると、サーバの作成が完了した時点で、呼出し側にサーバIDを返す。

仮想サーバ管理APIは、複数のスレッドから同時に呼び出すことで並列に動作させることができるが、同一のサーバIDに対するAPIを複数のスレッドから同時に呼び出すと、APIがエラーになることがある。同一のサーバIDに対するAPI呼出しの排他制御は、APIを呼び出すアプリケーションで行う必要がある。

[プロビジョニングシステムの設計]

仮想サーバホスティングシステムの開発に当たって、プロビジョニングシステムの設計を行った。

まず、プロビジョニングシステムの管理下に置く仮想サーバについて、プロビジョニングシステム上での状態遷移の仕様を定義した。仮想サーバの状態遷移は、仮想サーバごとに独立に制御する。一つの仮想サーバの状態遷移の仕様を図2に示す。

仮想サーバの状態は、管理ポータルサイトからのメッセージの受信や、処理完了の検出などのイベントが発生したときに遷移する。

なお、プロビジョニングシステム上での仮想サーバの状態の定義は、仮想サーバ管理APIの仕様上の定義と必ずしも一致している必要はないが、ここではAPIを利用しやすくするために、同じ定義を用いることにした。

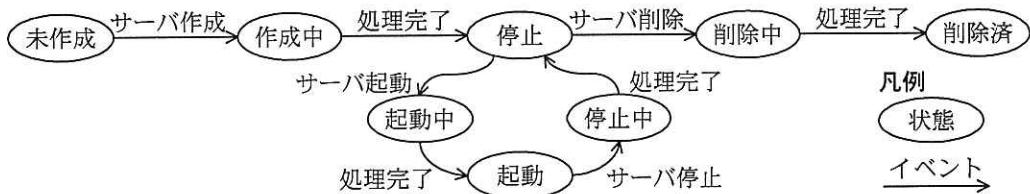


図 2 仮想サーバの状態遷移の仕様

次に、図 2 の状態遷移の仕様を実現するための、プロビジョニングシステムの構成を検討し、図 3 にまとめた。

プロビジョニングシステムの管理下に置く仮想サーバの状態は、仮想サーバのサーバ ID などの情報とあわせてデータベースに格納しておく。イベント発生時の状態遷移は、メッセージキューとメッセージハンドラを用いて実現する。

管理ポータルサイトから送信されたメッセージは、メッセージキューに格納される。メッセージキューに格納されたメッセージは、ディスパッチャが順次取得する。ディスパッチャは、取得したメッセージの内容と、仮想サーバの現在の状態に応じて、適切な種類のメッセージハンドラを生成する。生成したメッセージハンドラは、それぞれ独立した別々のスレッドで動作させる。メッセージキューに複数のメッセージが存在した場合は、メッセージごとに別々のスレッドを生成し、それぞれのメッセージハンドラを並列に動作させる。

メッセージハンドラは、データベースを更新し、仮想サーバ管理 API を呼び出す。メッセージハンドラの定義を表 2 に、ディスパッチャによるメッセージハンドラの選択ルールを表 3 に示す。

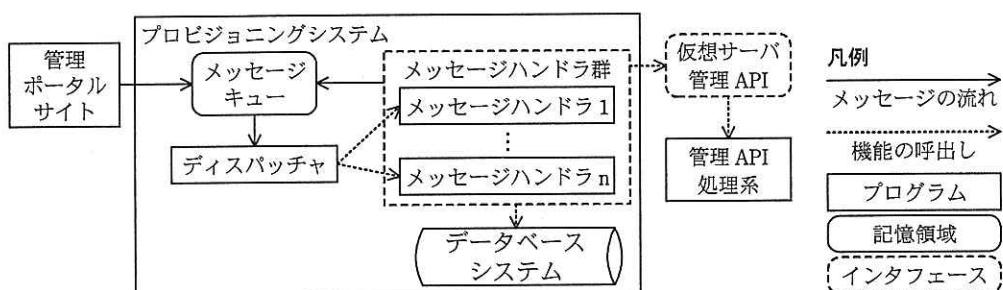


図 3 プロビジョニングシステムの構成

表2 メッセージハンドラの定義

| メッセージ ハンドラ名 | 仮想サーバの状態変更内容 | 呼び出す API | 処理完了メッセージ の送信 |
|----------------|---|--------------|------------------|
| Creating | “作成中”に変更する。 | CreateServer | する |
| Created | “ a ”に変更する。 | なし | ① |
| Starting | “起動中”に変更する。 | StartServer | ② |
| Ready | “起動”に変更する。 | なし | ③ |
| Stopping | “停止中”に変更する。 | StopServer | ④ |
| Stop | “停止”に変更する。 | なし | ⑤ |
| Deleting | “削除中”に変更する。 | DeleteServer | ⑥ |
| Deleted | “削除済”に変更する。 | なし | ⑦ |
| Err | なし | なし | しない |

注記 ①～⑦には“する”, “しない”的ないずれかが入る。

表3 メッセージハンドラの選択ルール

| メッセージ 種類 | 仮想サーバの状態 | | | | | | | |
|-------------|----------|---------|-------|----------|------|---|---------|-----|
| | 未作成 | 作成中 | 起動中 | 起動 | 停止中 | 停止 | 削除中 | 削除済 |
| サーバ作成 | Creating | — | — | — | — | — | — | — |
| サーバ起動 | Err | Err | Err | Err | Err | b | Err | Err |
| サーバ停止 | Err | Err | Err | Stopping | Err | Err | Err | Err |
| サーバ削除 | Err | Err | Err | Err | Err | c | Err | Err |
| 処理完了 | Err | Created | Ready | Err | Stop | Err | Deleted | Err |

注記 “—”は、起こり得ない状況であり、対応するメッセージハンドラがないことを示す。サーバ作成時にはサーバ ID を指定しないので、仮想サーバの現在の状態は必ず“未作成”となる。

例えば、ある仮想サーバの現在の状態が“起動”的なときにサーバ停止のメッセージを受け取ると、ディスパッチャは表3のルールに従って、Stopping のメッセージハンドラを生成する。Stopping のメッセージハンドラでは、表2の定義に従って、データベース中の仮想サーバの状態を“停止中”に変更した後、仮想サーバ管理 API の StopServer を呼び出す。メッセージハンドラは、メッセージハンドラの処理完了のイベントを発生させる必要がある場合に限り、呼び出した API の処理終了後に処理完了メッセージを生成し、メッセージキューに格納する。

[仮想サーバ削除時の自動停止機能の仕様変更]

仮想サーバの状態が“起動”的なときに、サーバ削除のメッセージを受信した場合は、

メッセージハンドラが StopServer の API を呼び出した後、自動的に状態遷移して、続いて DeleteServer の API が呼び出されるようにすることで、サーバを削除できるよう仕様を変更することにした。このとき、(i)サーバ停止とサーバ削除の二つのメッセージをメッセージキューに格納するだけだと、意図した結果にならないことがある。

そこで、“停止・削除中”の状態を新設することにした。

変更後の状態遷移では、仮想サーバの状態が“起動”的にサーバ削除のメッセージを受け取ると、仮想サーバは“停止・削除中”的な状態に遷移し、StopServer の API が呼び出される。StopServer が完了すると、自動的に“削除中”的な状態に遷移し、DeleteServer の API が呼び出される。

仕様を変更するために、表 2 と表 3 について、(1)～(3)の修正を加えた。

(1) 表 2 に、StopDeleting の行を追加する。追加する内容を表 4 に示す。

(2) 表 3 に、“停止・削除中”的列を追加する。追加する内容を表 5 に示す。

(3) 表 3 中の仮想サーバの状態が “d” で、メッセージの種類が e の箇所のメッセージハンドラ名を、f に変更する。

表4 表2に追加する内容

| メッセージハンドラ名 | 仮想サーバの状態変更内容 | 呼び出すAPI | 処理完了メッセージの送信 |
|--------------|----------------|------------|--------------|
| StopDeleting | “停止・削除中”に変更する。 | StopServer | する |

表5 表3に追加する内容

| メッセージの種類 | 仮想サーバの状態 |
|----------|----------|
| | 停止・削除中 |
| サーバ作成 | — |
| サーバ起動 | Err |
| サーバ停止 | Err |
| サーバ削除 | Err |
| 処理完了 | Deleting |

設問 1 表 2 及び表 3 中の a ~ c に入る適切な字句を答えよ。

設問 2 表 2 中の①～⑦について、処理完了メッセージを生成し、メッセージキューに格納するメッセージハンドラはどれか、①～⑦の番号で全て答えよ。

設問 3 本文中の下線(i)について、意図した結果にならない理由を、図 2 を参考に仮想サーバの状態と送信されるメッセージについて言及し、40字以内で述べよ。

設問 4 [仮想サーバ削除時の自動停止機能の仕様変更]について、(1), (2)に答えよ。

(1) 状態遷移図に“停止・削除中”的状態を追加し、状態遷移図を完成させよ。

(2) 本文中の d ~ f に入る適切な字句を答えよ。