```
Webアプリ、APIアプリ、モバイルアプリ、Function(関数)アプリをホスティングできる
                                            プラットフォームが提供される
                                            最新の言語のランタイムがすぐに使える
                                      PaaS J OSやランタイムのアップデートも自動的に行われる
                                            開発者は、開発したコードをすぐにApp Serviceにデプロイして運用できる
                                            VMでアプリを運用する場合に比べて、手間がかからない 生産性が向上
                                             アプリは、プランの上で動く
                                            プランは、VMSS+LB のようなもの。
                          [5]-[8]アプリとブランの関係 料金はアプリではなく、ブランに対して発生する
                                            [6]プランの「価格レベル」によって、その上で使える機能が変わってくる。
                                            [7][8]スケールアップ、スケールアウトは、プランのレベルで実行する。
                           [9]デモ: プラン作成、アプリ作成、アプリへのアクセス
                                             ランタイムスタック(言語)を選択
                                             OSを選択(Windows/Linux
                                             このアプリを乗せるプランを選択
                           [13]App Service アブリの作成 / アブリ名をsampleとすると、sample.azurewebsites.net のようなドメインが割り当てられる
                                             開発者は、Visual Studio や Visual Studio CodeでWebアプリを開発する
                                             開発したアプリを、App Serviceのアプリにデプロイして稼働させることができる
                                             Dockerコンテナーをデプロイすることもできる。 この場合は任意のランタイムを使用できる
                                       GitHub、Azure DevOpsなどのソースコード管理(バージョン管理)でコードを管理
                                       開発者はコードをリポジトリにプッシュする
                           [14]継続的デプロイ
                                       ブッシュしたコードが自動的にパイプラインでビルド、単体テストされ、App Serviceアプリにデプロイされる
                                       CI/CDを実現可能
                                         アプリの中に複数のデプロイスロットを作ることができる
                                         アプリ名をsample, デプロイスロットの名前を staging とすると、sample-staging azurewebsites net のよ
                                         うなドメイン名が割り当てられる
                                         運用中のアプリ(sample.azurewebsites.net)に影響を与えず、ステージング環境(sample-staging.
                          [15]デプロイスロット
              Azure App Service
                                          azurewebsites.net) にデプロイを行うことができる
                                         アプリが動いているブランによって、使えるスロット数が異なる。 Standardブランの場合、最大5 など
                                          App Serviceに組み込みの認証機能を(必要であれば)使える
                                          EasyAuth (簡易認証) とも呼ばれる この言葉でネット検索するといろいろ有益な情報が得られる
                                          Azure ADアカウントやMicrosoftアカウント等を使った認証をアプリに追加することができる
                                          通常、Webアプリでは、自前の認証機能(ログイン画面、パスワード管理機能、ユーザー登録機能など)を実
                                          装するが、このApp Serviceの認証機能を有効化することで、自前の認証機能を作り込む必要がなくなる。
                          [16]App Serviceの保護 √ 「オン」と「オフ」: デフォルトは「オフ」。「オン」にするとこの認証機能が使える。
                                          「匿名要求を許可する」: 認証が済んでいない(たとえば、Azure ADでサインインしていない)ユーザーが
                                           Webアプリにアクセスすることを許可する。認証が必要になったタイミングで、アプリケーションが認証用の
                                          URLへユーザー(のWebブラウザ)をリダイレクトして認証を開始する。例えばトップページはアクセスでき
                                          るが、その他のページはサインインが必要、といったケースに対応できる。
                                          「Azure Active Directoryでのログイン」など:選択した認証方式で認証が済んでいない場合、Webアプリの
                                          すべてのページにアクセスできない。未サインイン時は、強制的に、認証用のURLへリダイレクトされる。
                                          「App Serviceドメイン」でAzure portalを検索すると、ドメインを購入することができる画面が出てくる
                           [17]カスタムドメイン

Cこでドメインを購入して、そのドメインをWebアブリに割り当てて使用することができる
                                         購入したドメインはAzure DNSのゾーンとして管理できる。
                                         なお、Azure DNSそのものにはドメインの購入機能はない。
                                              App Service独自のバックアップ機能を備えている
                                              App Service にデブロイされたアプリと、アプリが接続するデータベースの内容をバックアップできる
                          [18]App Serviceのバックアップ バックアップはストレージアカウントに保存される。
                                              (モジュール10でAzure Backupというものを紹介するが、Azure BackupでのApp Serviceのバックアップは
今の所できない)
                                           いわゆるAPM(アプリケーションパフォーマンス管理)に分類される、アプリケーション監視用の機能を提供
                           [19]Application Insights (モジュール11でAzure Monitorというモニタリング機能を紹介するが、こちらは主にインフラレベルの監視ま
                                          で。Application Insightsは、アプリのより詳細な動作に関する情報を収集・分析できる)
                                          補足資料参照
                                                               どちらも仮想化の技術。
                                                              大きな違いとしては、VM(のイメージ)にはゲストOSが含まれている。コンテナ(のイメージ)にはゲスト
                                                          比較 OSは含まれていない。
                                                               よって、コンテナ型の仮想化のほうが、イメージのサイズが小さい。ネットワークですばやく送受信することが
                                                               でき、省メモリで動作する。
mod9 サーバーレス)
                                     [23]コンテナーと仮想マシンの比較
                                                                           コンテナが動作するには、DockerエンジンやKubernetesといった実行環境が必要となる。
                                                                           コンテナーは基本的に、そのコンテナーの動作に必要なすべてのものがパッケージングされている。アプリのバ
                                                           コンテナー型仮想化の特徴 イナリー、設定ファイル、ミドルウェア、言語ランタイム等。
                                                                           コンテナーは、一度開発すると、さまざまな環境で同じように動作することが期待できる。環境による差異を受
                                                          オーケストレーター Kubernetesのような、コンテナー型のシステムやアプリケーションの運用を支援するソフトウェア。
                                            Azureで、最も簡単かつ高速にコンテナーを実行するためのサービス。
                                            / コマンド一発ですばやくコンテナーを実行することができる
                                            PaaS型のサービス(インフラはAzureによって管理されており、基盤となるVMなどの管理が不要)
                                            コンテナーにパブリックIPアドレスやDNS名を割り当てることができる(オブション)
IPアドレス/DNS名 Webアブリをコンテナーで実行し、そのIPアドレスやDNS名でアクセスすることができる。
            [22]Azure Container Instances (ACI)
                                                   セキュリティ機能
                                     [24]ACI
                                            分離機能 ACI上のコンテナーは、VMレベルで分離されているのと同じ分離を提供する
                                                    つまり、VMが他のVMに影響を与えることができないのと同じように、コンテナーが他のコンテナーに影響を与
                                                  えることはできない
                                            サイズのカスタマイズ メモリや、CPUコア数を指定してコンテナーに割り当てることができる
                                           コンテナーのデファクトスタンダートはLinuxコンテナー
LinuxおよびWindowsコンテナー Windowsコンテナーにも対応できる
                                                       ACIの最上位のリソース。
                                                       コンテナーグループの中に、1つ〜複数のコンテナーが格納される
                                                        たとえば、あるコンテナーでWebサーバー、別のコンテナーでDBサーバーを稼働させるようなコンテナーグ
                                                       ループを作ることができる
                                     [25]ACIのコンテナーグループ
                                                       つまり、コンテナーグループの単位で、作ったり消したりする
コンテナーグループ単位のライフサイクル
コンテナーグループの動作中に、そごに別のコンテナーを追加する、といったことはできない
                                                       コンテナーグループ単位で、IPアドレスやDNS名などのリソース、ストレージ(Azure Files ファイル共有)が
                                                       共有される。
                                     [26]Docker Dockerの一般的な説明
                                     マネージド型のKubernetesサービス
                                              何らかのタスクを実行して、タスクが完了したらコンテナーはすぐ終了する、など
ACIは、コンテナーを単発ですばやく実行するためのサービス
Webアブリも実行はできるが、たとえばコンテナーの異常停止といった状況に対するサポートはない
                                              AKSは、クラスターを使用して、大規模なコンテナー型のアプリケーションやシステムを長期的に運用させるた
                                     ACIとの違い
             [28]Azure Kubernetes Service (AKS)
                                               「オーケストレーター」であるKubernetesは、コンテナーを使用したアプリケーションやシステムを長期間に
                                              渡って安定的・効率的に実行する機能や、コンテナーを追切なVM(ノード)に配置する機能、コンテナーが落ちた場合に再起動する機能、複数のコンテナーを起動してロードバランサーで負荷分散する機能、コンテナーに
                                              対してディスクなどの記憶域を割り当てる機能などを備えている。
                                     Kubernetesを使用する場合は、「Kubernetesクラスター」を構築する必要がある
                                    Kubernetesそのもののバージョンアップも実施する必要がある
                                        同じスペックのノードの集まり
                              ノードプール ∫ AKSのクラスターには「ブライマリノードプール」が1つある
                                       オプションで、複数のノードブールを利用することもできる
             [29]Kubernetes/AKS用語
                              ポッド Kubernetes上で動作させるアプリケーション
                              マニュフェスト ポッドの仕様を記述したYAMLファイル
                                                Kubernetes全体を管理するプロセスが稼働
                                                へ
AKSでは、コントロールプレーンにはコストは掛からない
                               コントロールプレーン(左) 利用者はkubectlというコマンドを使ってKubernetesクラスターを操作するが、この操作はAPIサーバーが受け
付ける
                                                付ける
                                                etcdはクラスター全体の状態を記録する
                                                スケジューラーは、ノードと通信を行う
             [30]Kubernetesクラスター
                                         -----
利用者のコンテナーを実行する
                               ノード(右)
                                       コンテナーランタイムが含まれる Moby
                                         Kube-proxyは、ネットワーク機能を提供する
                                         ノード内のコンテナーは、Azureディスク(VMと同じもの)やAzure Filesファイル共有にアクセスすることも
                                         できる
                            コンテナーがネットワークのトラフィックを受信するための設定方法が3つある
                           内部:クラスター内部でのコンテナー通信。プライベートのIPアドレスとボートを使用。
                                  Podに、ノード (VM) のIPアドレスとボートを割り当て、直接トラフィックを受け付ける
直接 (ロードバランサーを使わない)
                                  非直接 Azureロードバランサーを経由して、トラフィックを受信し、複数のノードのPodに負荷分散できる
```