Módulo 3 – Soluções de Dados, Big Data e Machine Learning

Bootcamp: Arquiteto Cloud Computing

Gustavo Aguilar

2020

Soluções de Dados, Big Data e Machine Learning

Bootcamp: Arquiteto Cloud Computing

Gustavo Aguilar

Sumário

Capítulo 1. Introdução à Plataforma de Dados do Azure

1.1. Modalidades de serviços

Em termos de modalidade de serviços, o Azure, assim como a maioria dos providers, oferecem três modalidades de serviços:

- **Infraestrutura como serviço (IaaS):** infraestrutura de computação instantânea, provisionada e gerenciada pela Internet. Com IaaS é possível aumentar e diminuir a infra rapidamente de acordo com a demanda, permitindo que se pague apenas pelo que usar. Ele ajuda a evitar as despesas e a complexidade de comprar e gerenciar seus próprios servidores físicos e outra infraestrutura de datacenter. Cada recurso é oferecido como um componente de serviço separado e só é preciso alugar um em particular pelo tempo que for necessário. Um provedor de serviços de computação em nuvem, como o Azure, gerencia a infraestrutura enquanto o cliente compra, instala, configura e gerencia seu próprio software — sistemas operacionais, middleware e aplicativos.

- **Plataforma como serviço (PaaS):** possui recursos que permitem ao cliente fornecer desde aplicativos simples baseados em nuvem, a sofisticados aplicativos empresariais habilitados para a nuvem. Os recursos são adquiridos por meio de um provedor de serviços de nuvem em uma base pré-paga e os acessa por uma conexão com a Internet segura. Assim como IaaS, o PaaS inclui infraestrutura — servidores, armazenamento e rede, além de middleware, ferramentas de desenvolvimento, serviços de BI (Business Intelligence), sistemas de gerenciamento de banco de dados e muito mais, com a diferença de que esses servidores são administradores pelo provider. PaaS permite ao cliente evitar os gastos e a complexidade de comprar e descontinuar software, infraestrutura e middleware, ou ferramentas de desenvolvimento e outros recursos. O cliente gerencia os aplicativos e serviços que ele fornece, e o provedor de serviços de nuvem normalmente gerencia todo o resto.

- **Software como Serviço (SaaS):** permite aos usuários se conectar e usar aplicativos baseados em nuvem pela Internet. Exemplos comuns são e-mail,

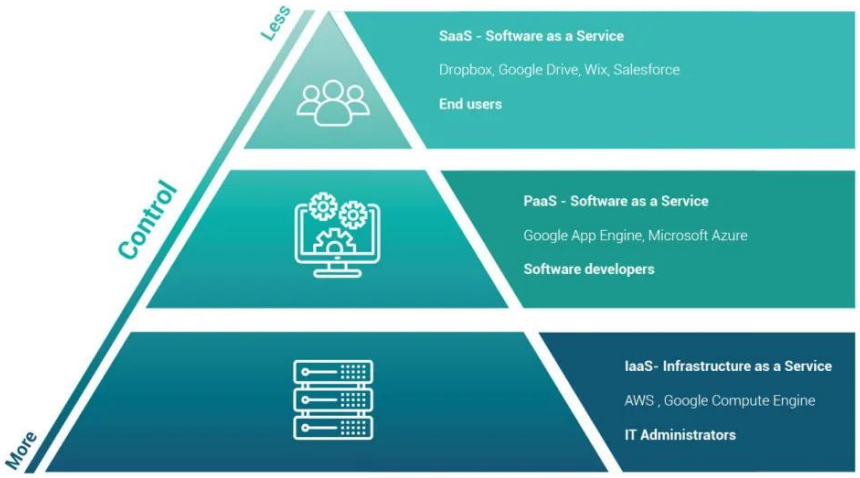
calendário e ferramentas do Office (como Microsoft Office 365). O SaaS oferece uma solução de software completa, onde pode-se comprar em um regime pré-pago de um provedor de serviço de nuvem. Pode-se alugar o uso de aplicativo para sua organização e seus usuários se conectarem a ele pela Internet, normalmente por um navegador da Web. Toda a infraestrutura subjacente, middleware, software de aplicativo e dados de aplicativo, ficam no datacenter do provedor de serviços. O provedor de serviço gerencia o hardware e software e, com o contrato de serviço apropriado, dá garantia de disponibilidade e segurança do aplicativo e de seus dados. O SaaS permite que uma organização entre em funcionamento rapidamente, com um aplicativo pelo custo inicial mínimo.

Figura 1 – Modalidades de serviços.

Fonte: Gustavo Aguilar (2019).

Importante notar que a escolha de uma modalidade em detrimento de outra, implica em ter mais controle (administração) ou menos controle da infraestrutura, bem como aumento ou redução do custo.

Figura 2 – Modalidades de Serviços x Controle da Infra.



1.2. Tipos de dados

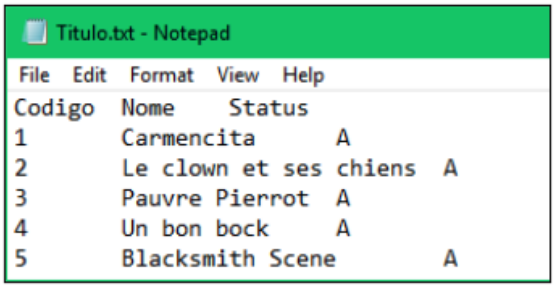
Quando Codd publicou a teoria do Modelo de Dados Relacional, em 1970, ele era então um pesquisador no laboratório da IBM, em San José. Entretanto, com o intuito de preservar o faturamento gerado pelos produtos “pré-relacionais”, como por exemplo o SGBD hierárquico IMS, a IBM optou inicialmente por não implementar as ideias relacionais de Codd.

Dados estruturados:

Quando se fala em dados estruturados, por trás está implicitamente envolvida a definição do modelo de dados, seja ele gráfico ou não. Essa definição contém quais campos de dados serão provisionados e como esses dados serão armazenados, ou seja, o tipo de dados dele (numérico, moeda, alfabético, data, binário etc.) e quaisquer restrições na entrada de dados (quantidade de caracteres, precisão, obrigatoriedade etc.). Com isso, pode-se perceber que dados estruturados são aqueles organizados e representados com uma estrutura rígida, a qual foi previamente planejada para armazená-los (MONTEIRO, 2019). Nesse planejamento, um dos pontos principais é definir acerca de cada campo de informação que o conjunto de dados irá conter.

Para armazenamento de dados estruturados existem diversos recursos tecnológicos, como arquivos textos tabulados (*flat files*) e planilhas eletrônicas, dentre os mais antigos, ou como as tabelas, nos famosos e muito utilizados bancos de dados relacionais.

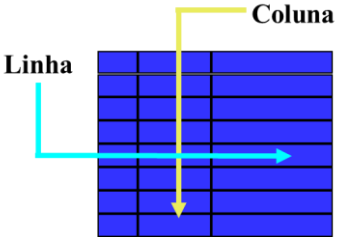
Figura 3 – Dados estruturados em flat files e em planilha eletrônica, respectivamente.



Codigo	Nome	Status
1	Carmencita	A
2	Le clown et ses chiens	A
3	Pauvre Pierrot	A
4	Un bon bock	A
5	Blacksmith Scene	A

Tanto para o armazenamento em planilhas eletrônicas, quanto para o armazenamento em bancos de dados relacionais, o formato utilizado é o tabular, onde os dados são dispostos em linhas e colunas, onde um determinado registro é visto como uma tupla (linha formada por uma lista ordenada de colunas).

Figura 4 – Formato tabular.



O formato de dados estruturados, principalmente o tabular, é o mais utilizado mundialmente, apesar de se estimar que apenas entre 5 a 10% do volume total de informações produzida pela humanidade seja estruturada.

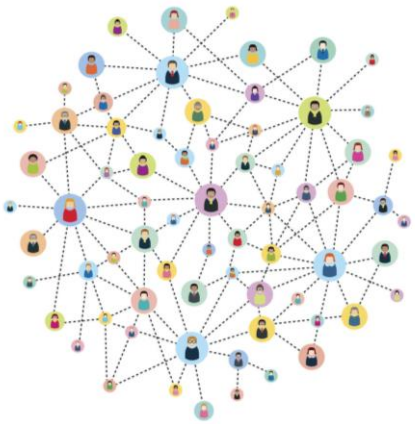
Dados não estruturados:

Os dados não estruturados possuem uma estrutura interna, mas não são estruturados por meio de modelos ou esquemas de dados predefinidos, como explica TAYLOR (2018). Podem ser gerados por humanos ou por máquinas, e um dos tipos mais comuns de dados não estruturados é o texto.

O texto não estruturado é gerado e coletado em uma ampla variedade de formulários, incluindo documentos do Word, mensagens de e-mail, apresentações em PowerPoint, respostas a pesquisas, transcrições de gravações de atendimentos de call center e postagens em blogs e sites de mídia social.

Outros tipos de dados não estruturados incluem arquivos de imagens, áudio e vídeo. Dados gerados por equipamentos é uma outra categoria de dados não estruturados, que cresce exponencialmente ao longo do mundo. Nela, encontra-se, por exemplo, arquivos de log de sites, servidores, redes e aplicativos (principalmente os móveis), os quais produzem uma grande quantidade de dados de atividade e desempenho. Em outra categoria, podemos encontrar os dados não estruturados gerados pelos sensores de equipamentos, satélites e outros dispositivos conectados à Internet das Coisas (IoT).

Figura 5 – Exemplo de informações não estruturadas representando os relacionamentos em uma rede social.



Fonte: iStock/aelitta (2019).

Com o advento da Internet e redes sociais, o volume de dados não estruturados se tornou o maior volume de dados produzido pela humanidade, sendo que até 2025, o IDG projeta que haverá 163 zettabytes de dados no mundo, dos quais estima-se que 80% serão dados não estruturados.

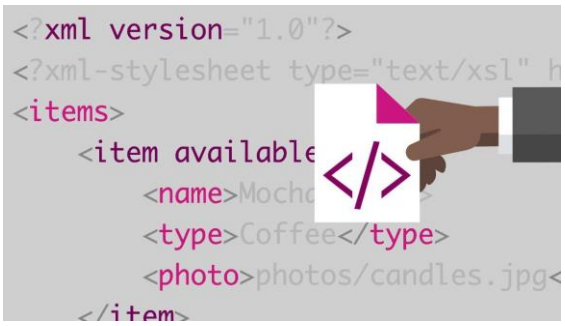
Dados semiestruturados:

Entre os dados estruturados e os não estruturados, surge o tipo de dado semiestruturado, que não contém toda a rigidez requerida na definição dos tipos de dados estruturados, mas que por outro lado procuram manter certa uniformidade no armazenamento das informações. Para isso, os dados semiestruturados mantêm tags e marcações internas que identificam elementos de dados separados, o que permite o agrupamento e formação de hierarquias de informações (TAYLOR, 2018).

Falando de armazenamento, tanto documentos quanto bancos de dados podem ser semiestruturados. Em termos de documentos, os mais utilizados são:

- **Formato XML** (Extensible Markup Language), com uma estrutura orientada a tags altamente flexível e muito usada para transporte de dados na Web.

Figura 6 – Exemplo de formato XML.



Fonte: Joe Marini (2019).

- **Formato JSON** (JavaScript Object Notation), um padrão aberto cuja estrutura consiste em armazenar dados em pares campo-valor (*filed: values*).

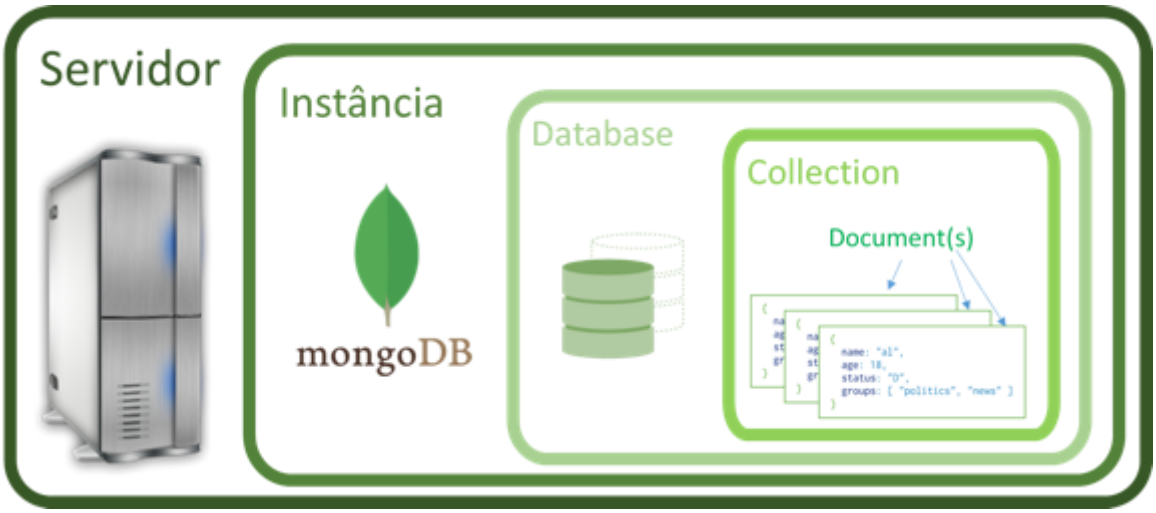
Figura 7 – Exemplo de formato JSON.

```
{
  name: "sue",                          <-- field: value
  age: 26,                              <-- field: value
  status: "A",                          <-- field: value
  groups: [ "news", "sports" ]          <-- field: value
}
```

Fonte: MongoDB, Inc. (2019).

Em termos de bancos de dados, os **bancos de dados NOSQL** (*Not Only SQL*) tem sido os mais utilizados para armazenamento de dados semiestruturados. Esse tipo difere bastante dos bancos de dados relacionais, porque não separa a organização (esquema) dos dados. Isso torna o NOSQL a melhor opção para armazenar informações que não se encaixam facilmente no formato de registro e tabela, como dados textuais com comprimentos variados ou linhas de dados com variância de informações (colunas). Alguns sistemas gerenciadores de bancos de dados NOSQL, como o MongoDB e o CouchDB, também incorporam documentos semiestruturados, armazenando-os nativamente no formato JSON.

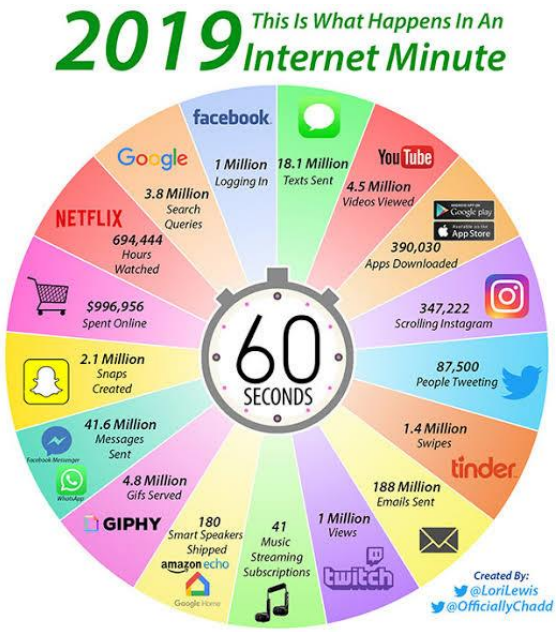
Figura 8 – Macroarquitetura de armazenamento de dados em JSON no MongoDB.



Da mesma forma que os dados estruturados, os dados semiestruturados correspondem ao menor volume de dados produzidos pela humanidade, algo estimado entre 5 e 10%.

Em plena “Era da Informação”, o uso massivo de Internet banda larga, a computação ubíqua e a Internet das Coisas (IOT), juntamente com o uso intensivo de redes sociais para divulgação de informações e conteúdo, vemos um crescimento exponencial do volume de dados, principalmente não estruturados (fotos, vídeos, posts, likes, snaps etc.).

Figura 9 – O que acontece em 1 minuto de Internet.



Fonte: Lori Lewis (2019).

1.3. Perfis de profissionais de dados

Nesse contexto de boom de dados e soluções para trabalhar com estes dados, fica nítido o papel fundamental do Profissional de Dados, seja ele um arquiteto de dados, engenheiro de dados, analista de dados etc.

Figura 10 – Processo macro de soluções de dados.

Projetar a Solução | Implantar a Infraestrutura para a Solução | Implantar o Pipeline de Dados | Análise de Dados Ciência de Dados Aprendizado de Máquina Etc.

Fonte: Gustavo Aguilar (2020).

Dentro do processo macro de planejamento, implantação e gerenciamento de soluções de dados mostrado acima, pode-se perceber a necessidade da participação dos seguintes perfis de profissionais de dados:

- **Arquiteto de Soluções (Solutions Architect):**
  - Visão holística de toda a solução.
  - Integração das soluções.
  - Recursos e capacidade para a solução.
  - Estratégias de backup e monitoramento.
  - Disponibilidade e escalabilidade da solução.
  - Plano de Continuidade de Negócio (PCN).
  - Custo da solução.
  - Etc.
- **Arquiteto de Dados (Data Architect):**
  - Estruturas de armazenamento dos dados.
  - Modelos de dados.

  - Administração de dados corporativos.
  - Backup de dados de negócio.

- **Administrador de Banco de Dados (DBA):**
  - Instalação / aprovisionamento:
    - SGBDs.
    - Plataformas de armazenamento de dados.
    - Bancos de dados.
    - Repositórios / quotas.
  - Aspectos operacionais.
  - Tunning e troubleshoting.
  - Disponibilidade dos SGBDs / plataformas.
  - Segurança de acesso.

- **Engenheiro de Dados (Data Engineer):**
  - Projeto do Pipeline (Fluxo) de Dados:
    - Extração de dados.
    - Transformação de dados.
    - Ingestão (carga) de dados.
  - Implementação e gerenciamento do fluxo de dados estruturados e não estruturados de diversas origens (fontes de dados).

- **Analista de Dados (Data Analyst):**
  - Projetar e construir modelos de dados analíticos.

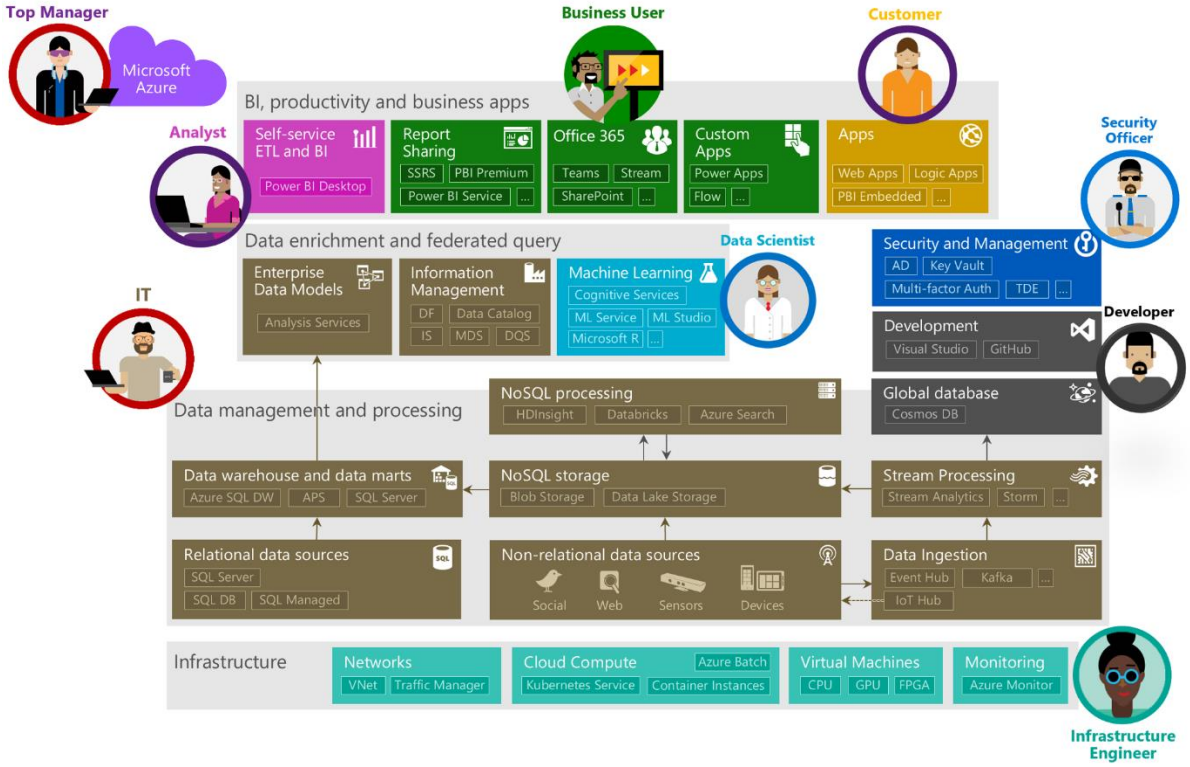
- Transformar dados em informações analíticas com valor comercial.
- Planejamento e gerenciamento de dashboards.

- **Cientista de Dados (Data Scientist):**
  - Análise avançada para ajudar a gerar valor a partir dos dados.
  - Análise descritiva: Análise Exploratória de Dados (EDA).
  - Análises preditivas: com Machine Learning.
  - Suporte a decisões orientadas a dados (Data-Driven Decision).
- **Engenheiro de Inteligência Artificial:**
  - Arquitetar e implementar soluções de IA:
    - Serviços cognitivos.
    - Machine Learning.
    - Mineração de conhecimento.
  - Suporte a soluções de processamento de linguagem natural, reconhecimento de voz e imagem.
  - Bots e agentes virtuais.

1.4. Overview da plataforma de dados do Azure

Os serviços da plataforma de dados do Microsoft Azure fornecem os recursos, habilidades, experiência e engajamento necessários para projetar, implementar e adotar uma solução orientada a dados de forma ágil e flexível.

Figura 11 – Overview da plataforma de dados do Azure.



Fonte: Microsoft (2019).

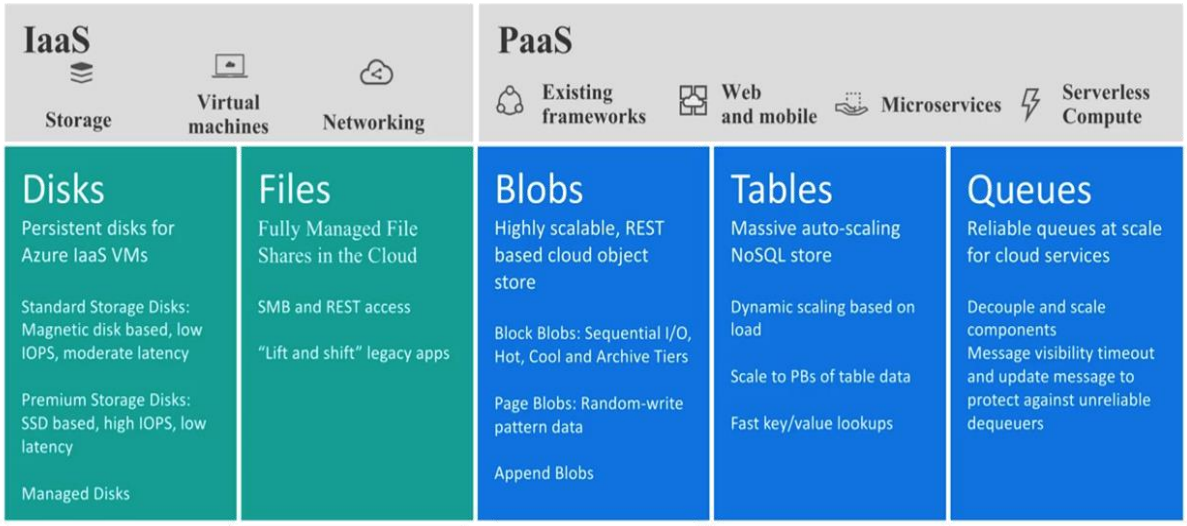
Azure Storage:

As contas de armazenamento do Azure são o tipo de armazenamento básico usado no Microsoft Azure. O armazenamento do Azure oferece um armazenamento de objetos massivamente escalável para objetos de dados e serviços de sistema de arquivos para a nuvem. Ele também pode fornecer um armazenamento de mensagens ou atuar como um armazenamento NoSQL.

Além disso, o Azure Storage é usado como base para armazenamento ao provisionar uma tecnologia de plataforma de dados, como o Azure Data Lake Storage e o HDInsight. No entanto, também pode ser provisionado para uso autônomo, como um Azure Blob Store, que é provisionado como armazenamento padrão, na forma de armazenamento em disco magnético ou armazenamento premium na forma de unidades de estado sólido (SSD).

Cada serviço é acessado através de uma conta de armazenamento (storage account) e pode ser usado com máquinas virtuais, via SMB ou API.

Figura 12 – Azure Storage.

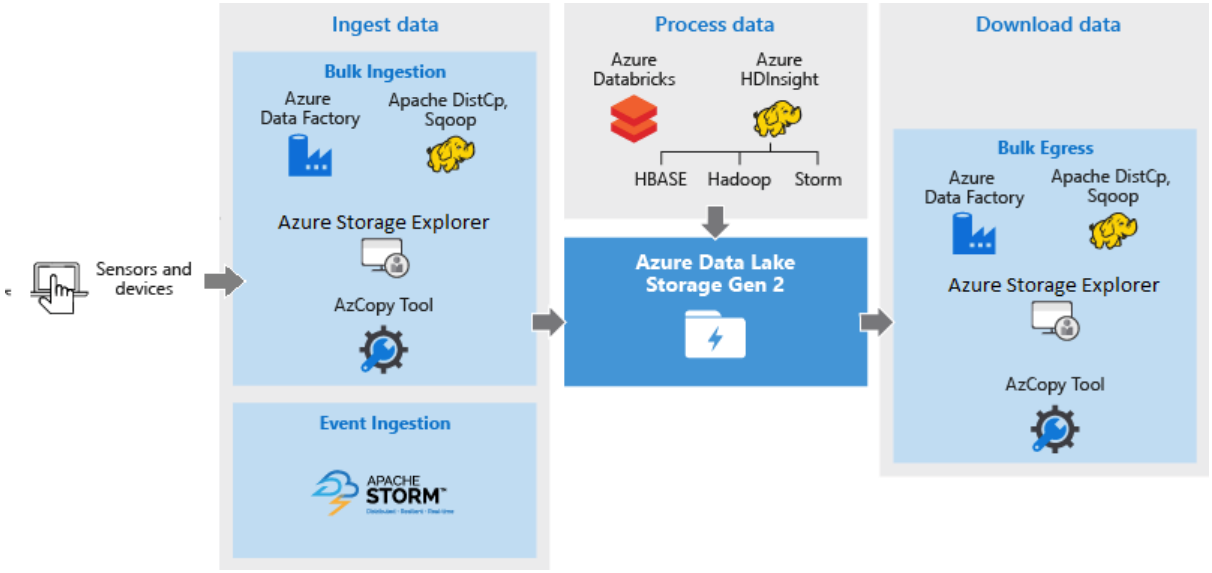


Azure Data Lake Storage:

Repositório de dados compatível com Hadoop (HDFS), que pode armazenar qualquer tamanho ou tipo de dados. O Azure Data Lake está disponível em duas ofertas: Geração 1 (Gen1) ou Geração 2 (Gen2), sendo que o Gen2 combina os serviços de armazenamento do Gen1 com os benefícios do Azure Blob Storage e tem o desempenho ajustado para o processamento de soluções de análise de big data.

Essa geração 2 inclui novos recursos, como um sistema de arquivos hierárquico, e os desenvolvedores podem acessar os dados por meio da API Blob ou da API de arquivos do Azure Data Lake (ADLS). Um benefício adicional para a geração 2 é que ele pode atuar em uma camada de armazenamento para uma ampla variedade de plataformas de computação, incluindo Azure Databricks, Hadoop ou Azure HDInsight, sem a necessidade de fazer a ingestão dos dados para estes sistemas.

Figura 13 – Exemplo de utilização do Azure Data Lake Storage Gen 2.



Fonte: Microsoft (2019).

Azure Databricks:

O Databricks é uma versão do popular mecanismo de análise e processamento de dados Apache Spark1, de código aberto. O Azure Databricks é a versão totalmente gerenciada do Databricks e é uma oferta premium no Azure, que oferece uma plataforma de Big Data e Aprendizado de Máquina baseada em nuvem segura.

Azure Cosmos DB:

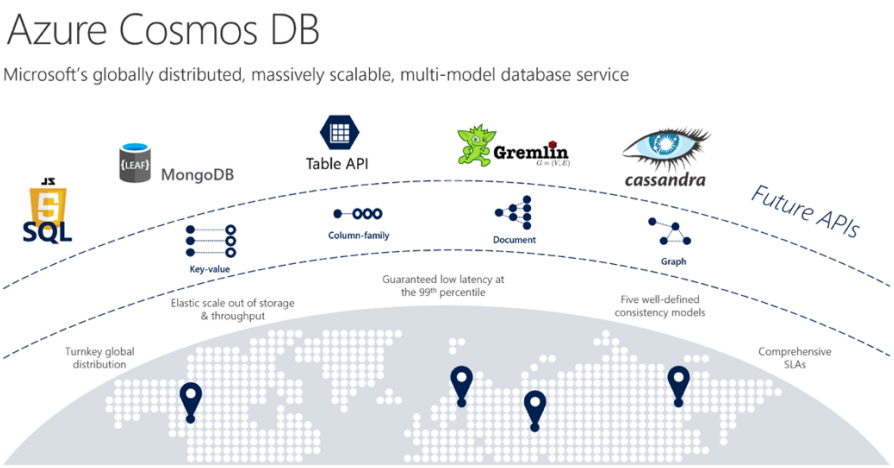
O Azure Cosmos DB é um banco de dados de vários modelos distribuído globalmente. Ele pode ser implantado usando vários modelos de API, incluindo:

- API SQL.
- API do Mongo DB.
- API do Cassandra DB.
- API do Gremlin DB.

- API da tabela.

Essa arquitetura de vários modelos permite ao Engenheiro de Banco de Dados alavancar os recursos inerentes a cada modelo, como MongoDB para dados semiestruturados, Cassandra para bancos de dados colunares ou Gremlin para bancos de dados de grafo. Usando o Gremlin, o Engenheiro de Dados pode criar entidades gráficas e executar operações de consulta de grafo para realizar travessias entre vértices e arestas, obtendo tempo de resposta em segundos para cenários complexos como o Processamento de Linguagem Natural (PNL) ou associações de redes sociais. Além disso, os aplicativos criados no SQL, MongoDB ou Cassandra, continuarão a operar sem alterações no aplicativo, apesar do servidor de banco de dados ser movido do SQL, MongoDB ou Cassandra para o Azure Cosmos DB.

Figura 14 – CosmosDB.



Fonte: Microsoft (2019).

Azure SQL Database:

O Banco de Dados SQL do Azure é um serviço de banco de dados relacional gerenciado no Azure, que suporta estruturas como dados relacionais e formatos não estruturados, como dados espaciais e XML.

Ele fornece OLTP (Online Transaction Processing), que pode ser dimensionado sob demanda enquanto emprega os recursos abrangentes de segurança e disponibilidade dos serviços de banco de dados no Azure.

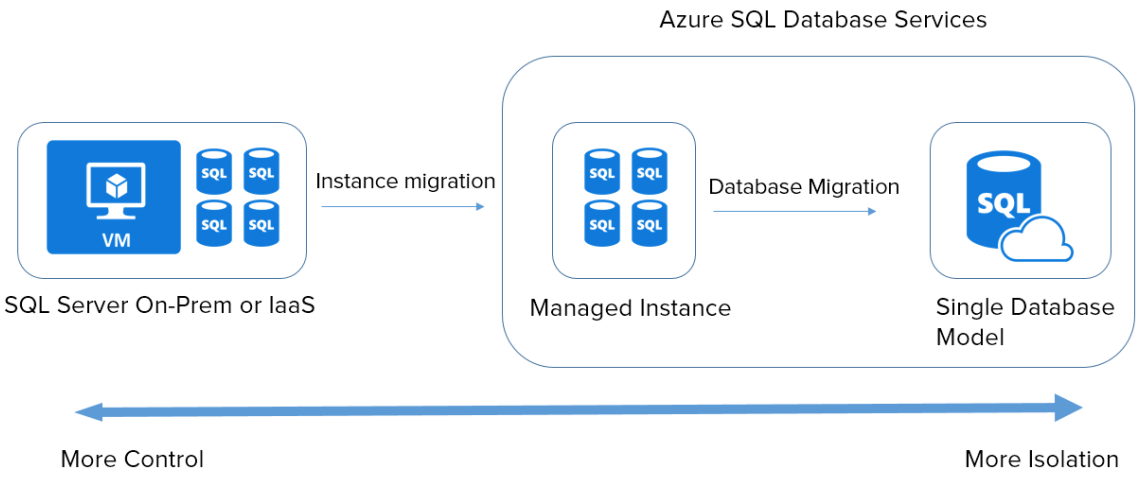
Azure SQL Managed Instance:

Embora muitas organizações migrem inicialmente para o Azure usando ofertas de IaaS, a oferta de serviço de plataforma como serviço (PaaS) permite benefícios adicionais. Um dos principais benefícios é que não se precisa mais instalar ou atualizar o SQL Server, pois já são atividades contempladas pelo serviço. Além disso, a verificação de consistência e os backups também fazem parte do serviço gerenciado, e há ferramentas adicionais de segurança e desempenho incluídas nas ofertas de PaaS.

Em termos de adoção inicial, a instância gerenciada do SQL do Azure permite caminhos de migração fáceis para aplicativos existentes, permitindo restaurações de backups locais.

Outro ponto positivo é que, ao contrário do Banco de Dados SQL do Azure, projetado com base em estruturas únicas de banco de dados, a instância gerenciada fornece uma instância inteira do SQL Server, permitindo até 100 bancos de dados, além de fornecer acesso aos bancos de dados do sistema. A instância gerenciada fornece outros recursos que não estão disponíveis no Banco de Dados SQL do Azure, incluindo consultas entre bancos de dados, CLR (Common Language Runtime) e o uso do SQL Agent.

Figura 15 – Azure SQL Managed Instance.



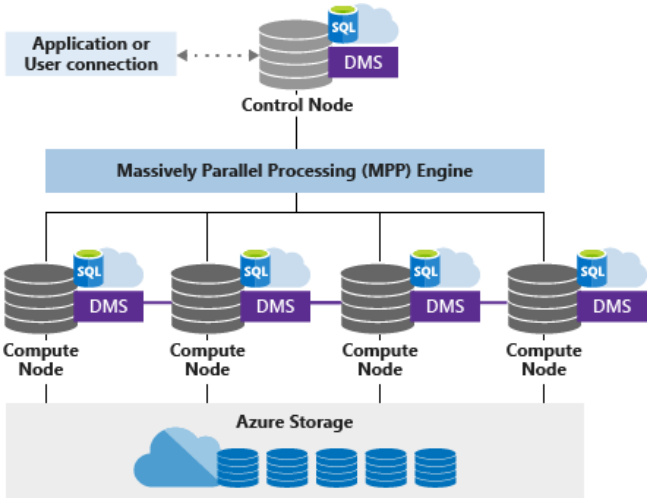
Fonte: Microsoft (2019).

Azure Synapse Analytics:

O Azure Synapse Analytics (formalmente conhecido como SQL DW), fornece um ambiente unificado combinando o enterprise data warehouse do SQL, os recursos de análise de Big Data do Spark e as tecnologias de integração de dados para facilitar a movimentação de dados entre os dois, e de fontes de dados externas.

Usando o Azure Synapse Analytics, pode-se ingerir, preparar, gerenciar e fornecer dados para necessidades imediatas de BI e aprendizado de máquina. Do ponto de vista do data warehouse, o Azure Synapse Analytics usa o MPP (Massive Parallel Processing) para executar consultas rapidamente entre petabytes de dados.

Figura 16 – Azure Synapse Analytics.



Fonte: Microsoft (2019).

1.5. Outros serviços da plataforma de dados do Azure

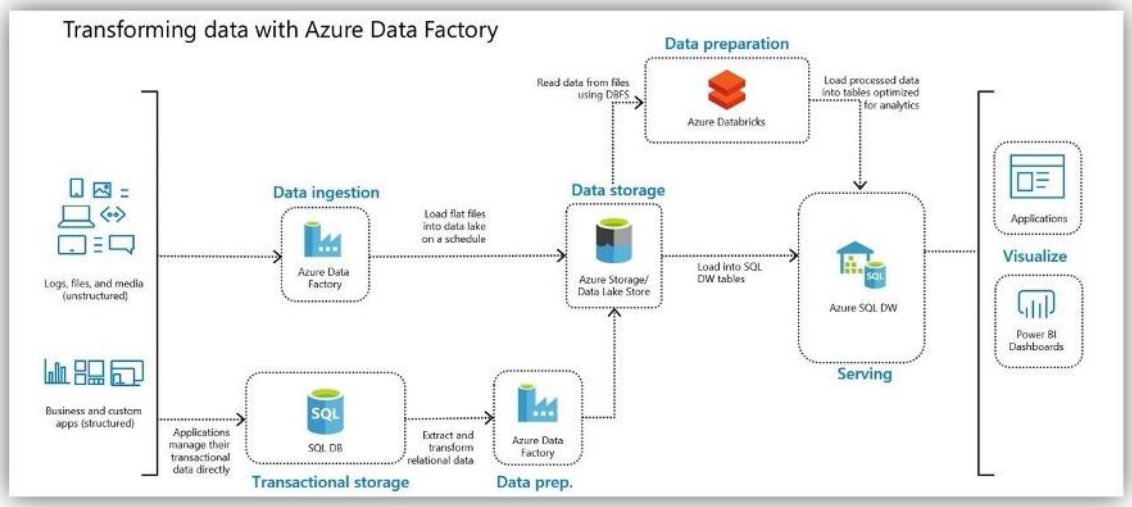
Azure Data Factory:

É o ETL baseado em nuvem e o serviço de integração de dados, que permite criar fluxos de trabalho orientados a dados para orquestrar a movimentação de dados e transformar dados.

Usando o Azure Data Factory, pode-se criar e agendar fluxos de trabalho controlados por dados (chamados pipelines), que podem ingerir dados de diferentes armazenamentos de dados. Pode-se, visualmente, criar processos ETL complexos, que transformam dados com fluxos de dados ou usando serviços de computação como o Azure HDInsight Hadoop, Azure Databricks e o Banco de Dados SQL do Azure.

Além disso, pode-se publicar os dados transformados em recursos de armazenamentos de dados, como o Azure SQL Data Warehouse, para uso em aplicativos de business intelligence (BI).

Figura 17 – Exemplo de fluxo de transformação de dados com Data Factory.



Fonte: Microsoft (2019).

Azure HDInsight:

O Azure HDInsight fornece tecnologias para ingestão, processamento e análise de big data para dar suporte ao processamento em lote, data warehousing, IoT e Data Science. O Azure HDInsight é uma solução em nuvem de baixo custo, que contém várias tecnologias, incluindo Apache Hadoop, Apache Spark, Apache Kafka, Apache HBase, consulta interativa e Apache Storm.

O Apache Hadoop inclui Apache Hive, Apache HBase, Spark e Kafka. O Hadoop armazena dados usando um sistema de arquivos (HDFS), enquanto o Spark armazena dados na memória, tornando o Spark aproximadamente 100 vezes mais rápido.

Figura 18 – Processamento de fluxo de dados no HDInsight.

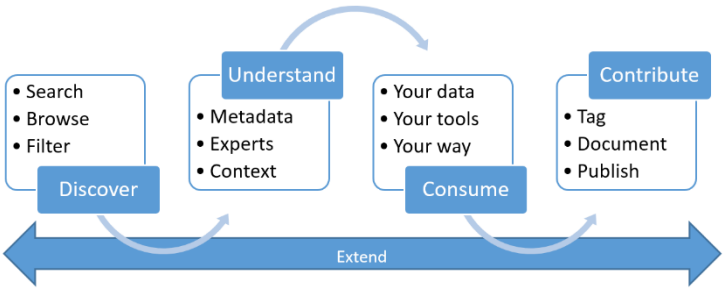
Fonte: Microsoft (2019).

Azure Data Catalog:

Com o catálogo de dados, qualquer usuário (analista, cientista de dados ou desenvolvedor) pode descobrir, entender e consumir fontes de dados. O catálogo de dados inclui um modelo de crowdsourcing de metadados e anotações. É um local único e central, para que todos os usuários de uma organização contribuam com seu conhecimento e construam uma comunidade e uma cultura de fontes de dados pertencentes a uma organização.

É um serviço de nuvem totalmente gerenciado. Os usuários podem descobrir as fontes de dados de que precisam e entender as fontes de dados que encontram, e usar o catálogo de dados para ajudar as organizações a documentar as informações sobre suas fontes de dados.

Figura 19 – Azure Data Catalog.



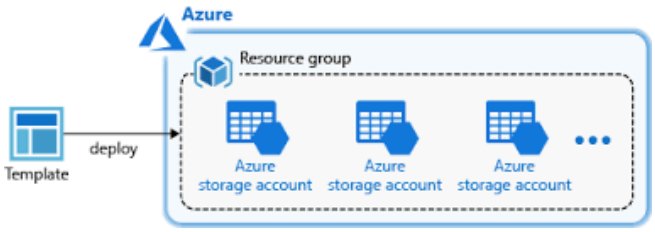
Fonte: Microsoft (2019).

Capítulo 2. Armazenamento de dados

2.1. Storage Account

Uma conta de armazenamento (storage account) é um contêiner que agrupa um conjunto de serviços de Armazenamento do Azure e está incluída em um grupo de recursos (Resoucer Manager).

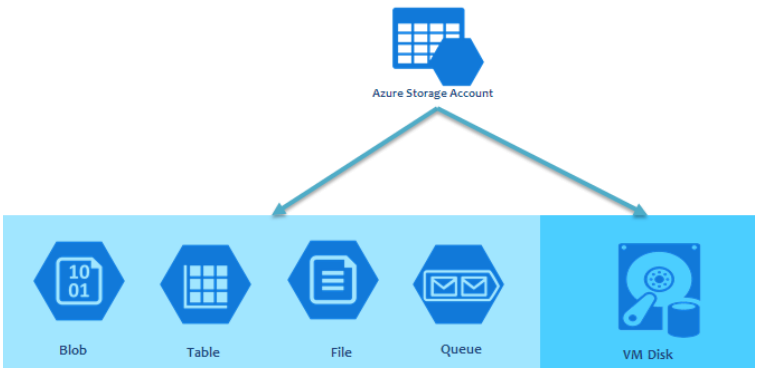
Figura 20 – Storage Account.



Fonte: Microsoft (2019).

Somente serviços de armazenamento de dados podem ser incluídos em uma conta de armazenamento (Blobs do Azure, Arquivos do Azure, Filas do Azure e Tabelas do Azure).

Figura 21 – Serviços de armazenamento de dados e Storage Account.



Fonte: Microsoft (2019).

Uma storage account define uma política que se aplica a todos os serviços de armazenamento na conta. Por exemplo, pode-se especificar que todos os serviços

contidos serão armazenados no datacenter do oeste dos EUA, acessíveis apenas por https e cobrados na assinatura do departamento de vendas.

O tipo de conta de armazenamento é um conjunto de políticas que determinam quais serviços de dados você pode incluir na conta e os preços desses serviços. Existem três tipos de contas de armazenamento:

- **StorageV2 (uso geral v2):** a oferta atual que suporta todos os tipos de armazenamento e todos os recursos mais recentes.
- **Storage (finalidade geral v1)**: um tipo herdado que suporta todos os tipos de armazenamento, mas pode não suportar todos os recursos.
- **Blob Storage**: um tipo herdado que permite apenas blobs de blocos e anexos.

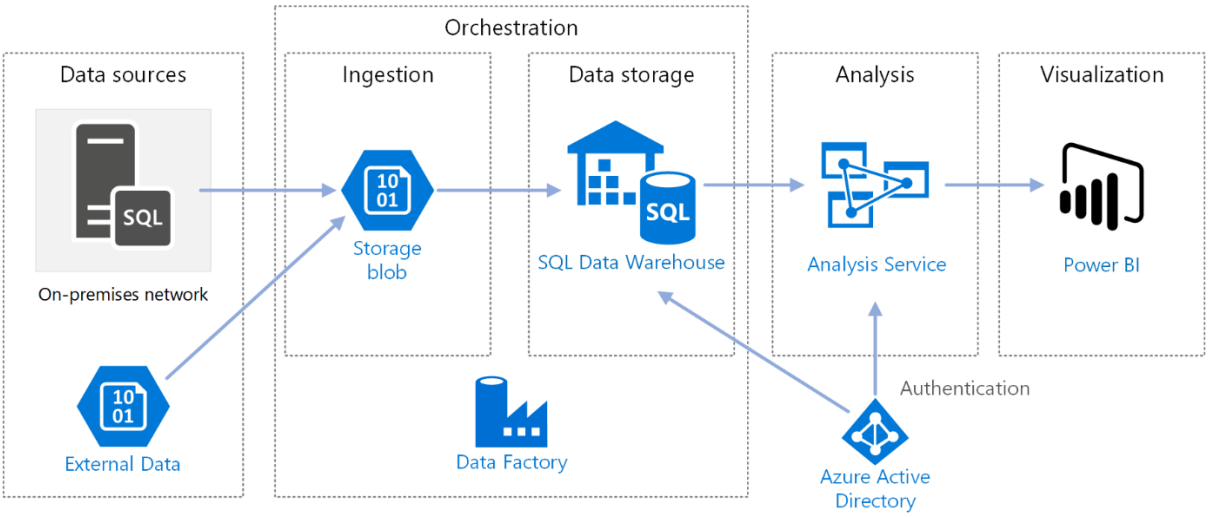
Storage accounts podem ser criadas através dos recursos:

  - Portal do Azure.
  - Azure CLI.
  - Command-line interface.
  - Azure PowerShell.
  - Management client libraries (para incorporar a criação dentro de um app).

2.2. Ingestão de dados

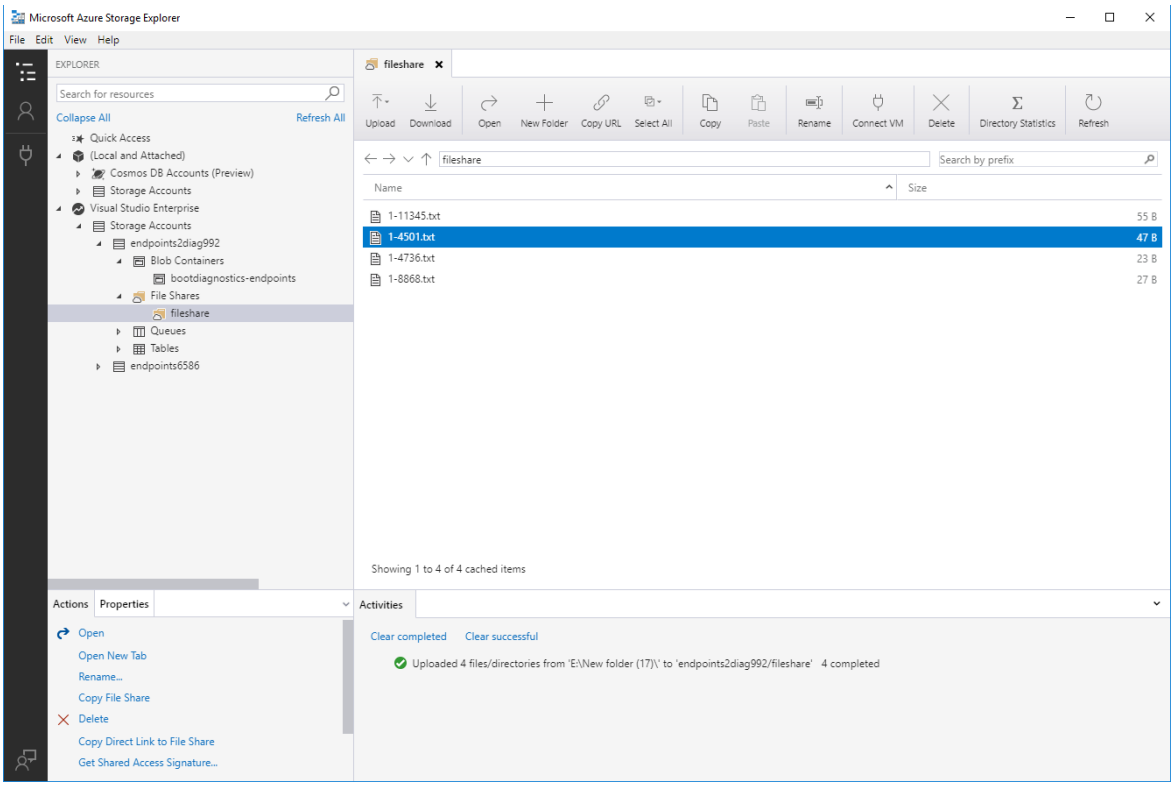
Para ingerir dados, um Engenheiro de Dados pode usar o Azure Data Factory, o Azure Storage Explorer ou a AzCopy Tool, PowerShell ou Visual Studio.

Figura 22 – Azure Data Factory.



Fonte: Microsoft (2019).

Figura 23 – Azure Storage Explorer.



Fonte: Microsoft (2019).

Para importar tamanhos de arquivo acima de 2 GB usando o recurso upload de arquivos, os engenheiros de dados devem usar o PowerShell ou o Visual Studio. A ferramenta AzCopy suporta um tamanho máximo de arquivo de 1 TB e será automaticamente dividida em vários arquivos se o arquivo de dados exceder 200 GB.

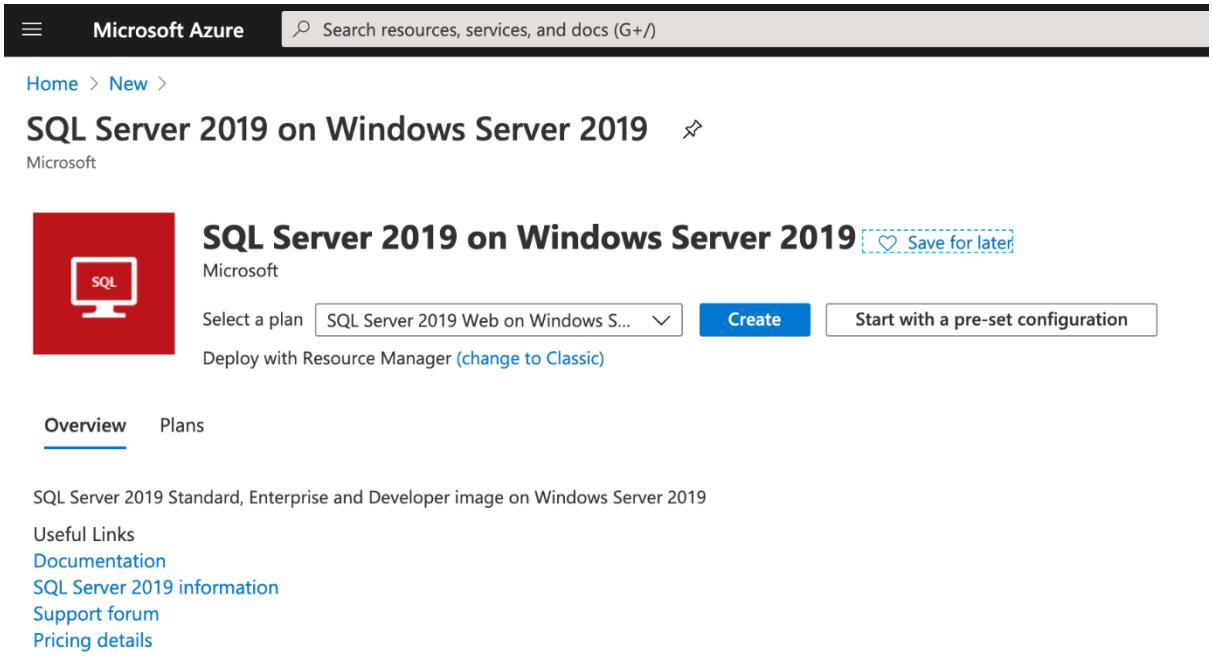
Capítulo 3. Armazenamento de Dados Relacionais no Azure

3.1. Bancos de dados relacionais em IaaS

Usando-se da modalidade de serviço infraestrutura como serviço (IaaS), é possível aprovisionar servidores virtuais e instalar manualmente os sistemas gerenciadores de bancos de dados (SGBD) nos mesmos.

Outra possibilidade é usar um template ARM (Azure Resource Manager), como o mostrado abaixo, que já vem com o sistema operacional instalado e o SGBD.

Figura 24 – Template ARM.



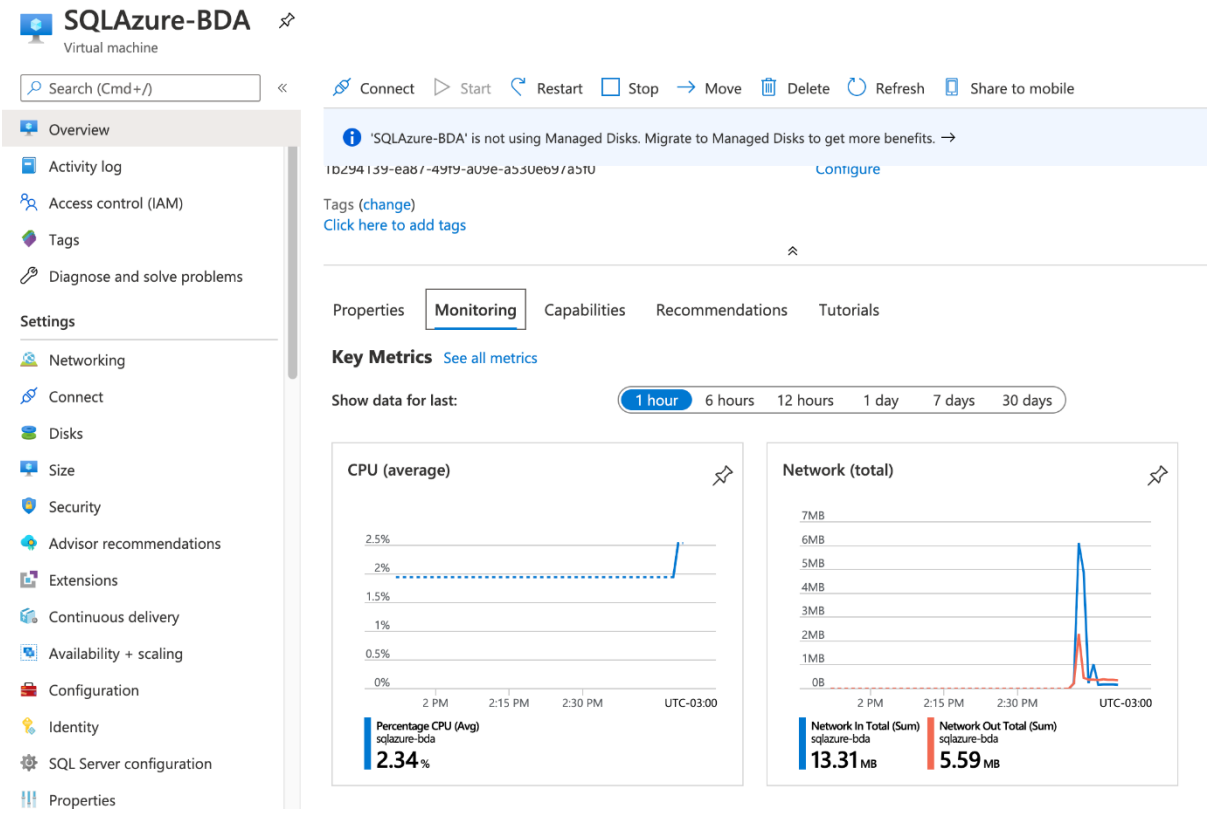
Fonte: Gustavo Aguilar (2020).

Quando se aprovisiona uma VM do Marketplace com SQL Server, parte do processo instala o SQL Server IaaS Agent Extension. Extensões são códigos executados na pós-implantação da VM, normalmente para executar configurações pós-implantação, como a instalação de recursos antivírus ou a instalação de um recurso do Windows. A extensão do agente IaaS do SQL Server fornece três recursos principais, que podem reduzir sua sobrecarga administrativa.

- Backup automatizado do SQL Server.
- Patches automatizados do SQL Server.
- Integração do Azure Key Vault (“cofre de senhas”).

Além desses recursos, a extensão permite exibir informações sobre a configuração, performance e utilização de armazenamento do SQL Server.

Figura 25 – Dasboard de VM com SQL Server IaaS Agent Extension.



Fonte: Gustavo (2020).

3.2. Azure SQL Database Managed Instance

Instância de banco de dados SQL Server gerenciada (SQL Database Managed Instance) é fornecida em duas camadas de serviço: **Propósito Geral (General Purpose)** e **Negócios Críticos (Business Critical)**. Ambas as camadas de

serviço suportam o mesmo conjunto de recursos, e as principais diferenças entre as duas estão relacionadas ao desempenho e disponibilidade. As únicas diferenças de recursos entre as duas camadas são que a Business Critical suporta OLTP In-Memory e leitura nas réplicas secundárias. Ela também inclui mais memória por núcleo (vCore) e usa armazenamento atachado direto (ao contrário de SAN), o que oferece menor latência de armazenamento.

É possível aproveitar as licenças de SQL Server já adquiridas ao se migrar para SQL Managed Instacen. Para cada núcleo do Enterprise Edition com Active Software Assurance, você é elegível para um vCore do Azure SQL Database ou Managed Instance Business Critical e oito vCores de General Purpose. Para cada núcleo da Standard Edition com Software Assurance que você possui, você é elegível para um vCore da General Purpose. Isso pode reduzir o custo total da licença em até 40%.

O banco de dados SQL do Azure e a instância gerenciada, têm arquiteturas semelhantes de alta disponibilidade, que garantem 99,99% por cento de tempo de atividade. As atualizações do Windows e do SQL Server são tratadas pela infraestrutura de back-end e de responsabilidade do Azure. A solução de alta disponibilidade é automática e integrada à plataforma e foi projetada para que os dados comprometidos nunca sejam perdidos por falhas e os bancos de dados não tenham um ponto único de falha.

O backup gerenciado fornece um serviço de backup totalmente gerenciado que realiza backups completos, diferenciais e de log regularmente. É possível também realizar manualmente backups “copy only” dos bancos de dados no Azure.

3.3. Azure SQL Database

O banco de dados SQL do Azure é um mecanismo de banco de dados PaaS (plataforma como serviço) totalmente gerenciado, que lida com a maioria das funções de gerenciamento de banco de dados, como atualização, aplicação de patches, backups e monitoramento sem envolvimento do usuário. O banco de dados SQL do

Azure está sempre em execução na versão estável mais recente do SQL Server e de patches do sistema operacional, com 99,99% de disponibilidade.

Possui dois modelos de implantação:

- **Single Database:**
  - Banco de dados isolado.
  - Totalmente gerenciado pelo Azure.
- **Elastic Pool:**
  - Coleção de single databases.
  - Com um conjunto compartilhado de recursos (CPU / RAM).

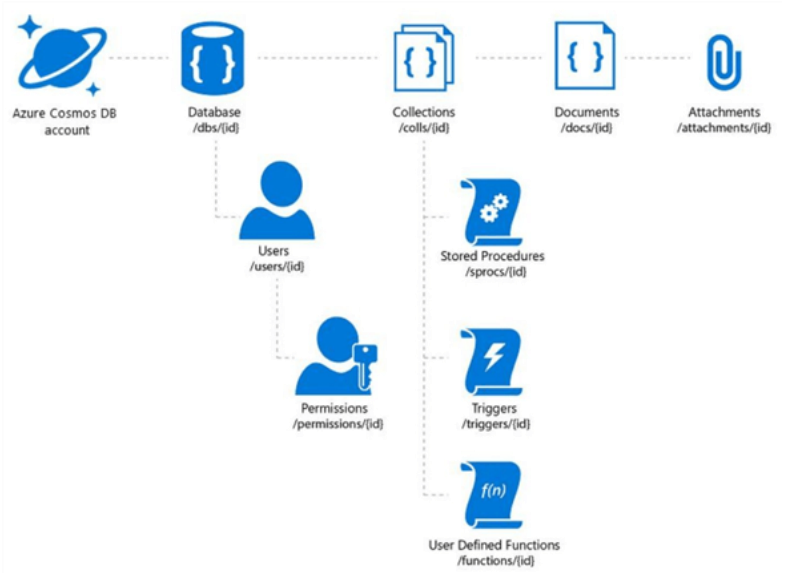
Modelos de contratação:

- **Modelo de compra baseado em vCore:** permite escolher o número de vCores, a quantidade de memória e a quantidade e velocidade do storage.
  - **Modelo de compra baseado em DTU**: oferece uma combinação de recursos de computação (CPU), memória RAM e I/O.
  - **Modelo serverless (sem servidor):** dimensiona automaticamente a configuração necessária de recursos, com base na demanda da carga de trabalho e cobra pela quantidade de computação usada por segundo. Neste modelo, os bancos de dados também são pausados automaticamente durante os períodos inativos. Dessa forma, apenas o armazenamento é cobrado e os bancos de dados são ativados automaticamente quando a atividade retorna.

3.4. Azure CosmosDB

Antes de se criar um banco de dados CosmosDB, é preciso criar uma conta do CosmosDB. Uma conta do Azure Cosmos DB é um recurso do Azure que atua como uma entidade organizacional para seus bancos de dados. Ele conecta seu uso à sua assinatura do Azure para fins de cobrança. Cada conta do Azure Cosmos DB está associada a um dos vários modelos de dados aos quais o Azure Cosmos DB oferece suporte e podem ser criadas quantas contas precisar.

Figura 26 – Conta Azure CosmosDB.



Fonte: Microsoft (2019).

O Azure CosmosDB mede a taxa de transferência usando uma métrica chamada unidade de solicitação (Requisition Unit - RU). O uso da unidade de solicitação é medido por segundo, portanto, a unidade de medida para o CosmosDB é unidades de solicitação por segundo (RU/s). Deve-se reservar o número de RU/s que deseja que o Azure CosmosDB provisione com antecedência para que ele possa lidar com a carga estimada, mas pode-se aumentar ou diminuir a RU/s a qualquer momento.

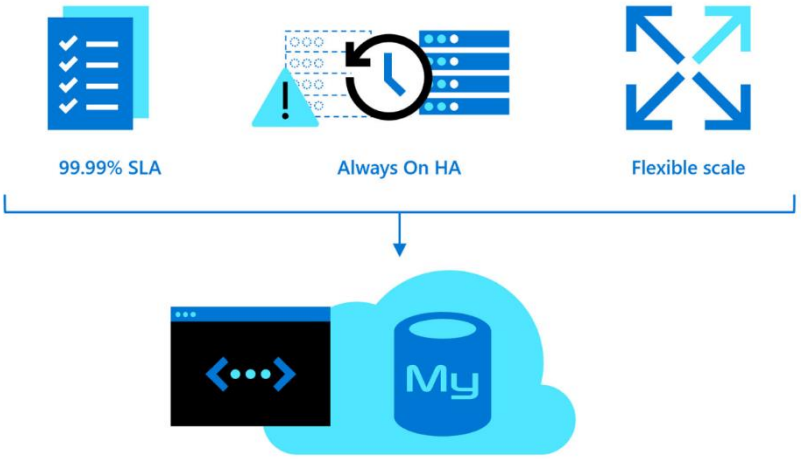
3.5. Bancos de dados Open Source no Azure

Com o mesmo viés e implicações da modalidade de PaaS oferecida com o Azure SQL Database, é possível criar os seguintes bancos de dados open source no Azure:

- Azure Database for MySQL.
- Azure Database for MariaDB.
- Azure Database for PostgreSQL.

Para o MySQL e MariaDB, há o recurso de replicação (master / slave), mas também existe a opção de criação de um database único (single database).

Figura 27 – Azure Database para MySQL.



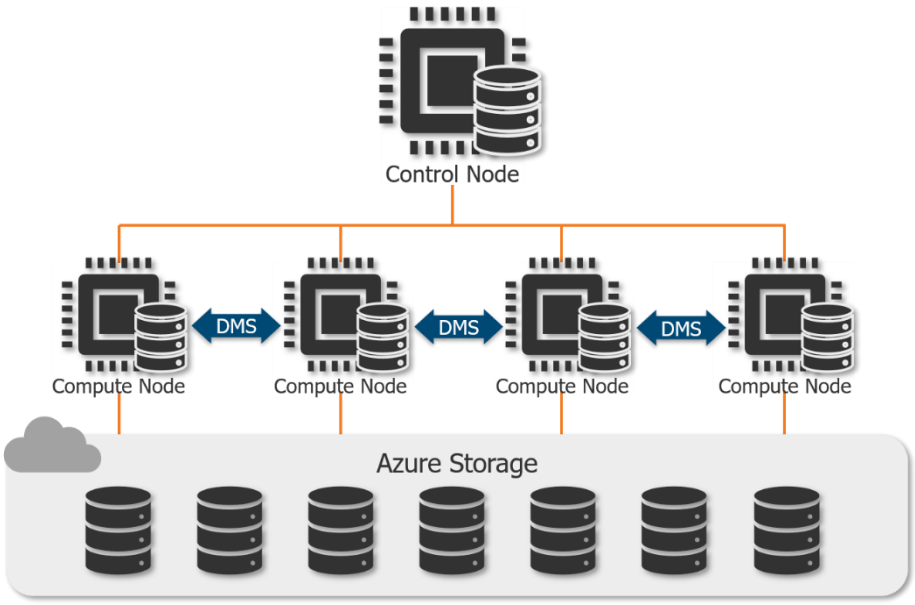
Fonte: Microsoft (2019).

3.6. Azure Synapse Analytics

O Azure Synapse Analytics, também conhecido formalmente como SQL DW, é um serviço de análise que reúne data warehouse e análise de Big Data, usando processamento massivo paralelo (MPP).

Com o Azure Synapse Analytics CPU, memória e IO são agrupados em unidades de escala de computação chamadas SQL Pools. O tamanho de cada pool SQL é determinado por Data Warehousing Units (DWU).

Figura 28 – SQL Pool do Azure Synapse Analytics.



Fonte: Microsoft (2019).

Para controlar as cargas de trabalho (workloads), em termos de uso de recursos dos SQL Pools, pode criar grupos de carga de trabalho (workload groups). Os grupos de carga de trabalho permitem definir os recursos para isolar e reservar recursos para seu uso, trazendo alguns benefícios para a solução, como:

- Reservar recursos para um workload.
- Limitar a quantidade de recursos que um grupo pode consumir.
- Garantir o uso de recursos compartilhados com base no nível de importância do workload.

Capítulo 4. Armazenamento de Dados Não Relacionais no Azure

4.1. Bancos de dados não relacionais no Azure

NOSQL é um termo que identifica qualquer tipo de banco de dados que não seja relacional, no sentido de armazenar os dados de forma tabular. Como esses bancos de dados não têm esquemas rígidos (ou possuem esquemas flexíveis), há menos sobrecarga ao tentar manter a integridade dos dados, como chaves estrangeiras, tipos de dados e campos opcionais. Em vez disso, cabe aos desenvolvedores de aplicativos e cientistas de dados manter a integridade dos dados.

Existem quatro grandes famílias de bancos de dados NOSQL:

- Chave-valor (key-value).
- Colunar.
- Bancos de dados de documentos.
- Bancos de dados de grafos.

Família chave-valor: “Os dicionários”:

Bancos de dados que usam o conceito de chave para acessar uma unidade de dados (o valor). No sentido mais simples, eles podem ser consultados em um dicionário distribuído. Muitas vezes eles oferecem grande confiabilidade e desempenho em detrimento da consistência ou escalabilidade. Esses bancos de dados não precisam se preocupar em juntar dados ou relacionamentos, ou manter a integridade dos dados quando novos dados são adicionados. Alguns casos de uso excelentes para esses bancos de dados são:

- Cache distribuído e dados de usuário / sessão.
- Aplicativos pesados de gravação, como bate-papos, carrinhos de compras e registros.

Esses são bancos de dados que armazenam dados em colunas em vez de linhas. Em um sentido simplista, eles são como um banco de dados SQL que **não** permite que você:

- Consulta por qualquer coisa, exceto a chave primária.
- Crie índices para encontrar dados mais rapidamente.
- Junte os dados.

Por que tantas limitações? Porque esses bancos de dados geralmente atendem ao mundo do Big Data. Eles são construídos para absorver dados em uma taxa monstruosa e fornecer consultas razoavelmente rápidas, devido ao seu tamanho. Esses bancos de dados costumam ser tão grandes que se propagam em várias máquinas, de modo que o dimensionamento é apenas uma questão de adicionar outro nó de máquina.

Ao proibir junções e consultas no conteúdo dos dados, cada máquina precisa apenas se preocupar com as chaves que está armazenando. Portanto, uma consulta em um desses bancos de dados está essencialmente sendo direcionada apenas para o nó em que reside o ponto de dados. Essas limitações evitam a execução de cada consulta em todos os nós do cluster.

Família de bancos de dados de documentos:

Esses bancos de dados armazenam dados na forma de documentos como JSON, BSON ou XML. Eles não têm esquemas flexíveis ou inexistentes e permitem a criação de índices nos dados. Alguns escalam facilmente e outros são ACID. Cada um desses bancos de dados traz algo diferente quando se trata de recursos. No entanto, esses bancos de dados são os primeiros candidatos a entrar em um banco de dados relacional tradicional. Na maior parte, eles atendem às mesmas necessidades dos bancos de dados relacionais tradicionais. Em vez de usar SQL para manipular os dados, esses bancos de dados usam APIs e SDKs.

Algumas das propostas de valor mais comuns deles, são:

- Código aberto.
- Desempenho mais rápido.
- Agilidade mais rápida do desenvolvedor (devido a APIs / SDKs e sem esquema).
- Eventualmente consistente.
- Menos necessidade de otimização, ou seja, menos necessidade de um DBA.
- Menor custo em hospedagem + taxas de licenciamento.

Família dos bancos de dados de grafo:

Difícil de acreditar, mas esses são mais relacionais do que os bancos de dados relacionais tabulares tradicionais que usam SQL. Em vez de armazenar os relacionamentos com os dados, eles colocam ênfase nos próprios relacionamentos e depois nos dados.

Redes sociais e mecanismos de recomendação são ótimos casos de uso para bancos de dados de grafo. Esses bancos de dados brilham em situações em que muitas junções causariam confusão em um banco de dados SQL.

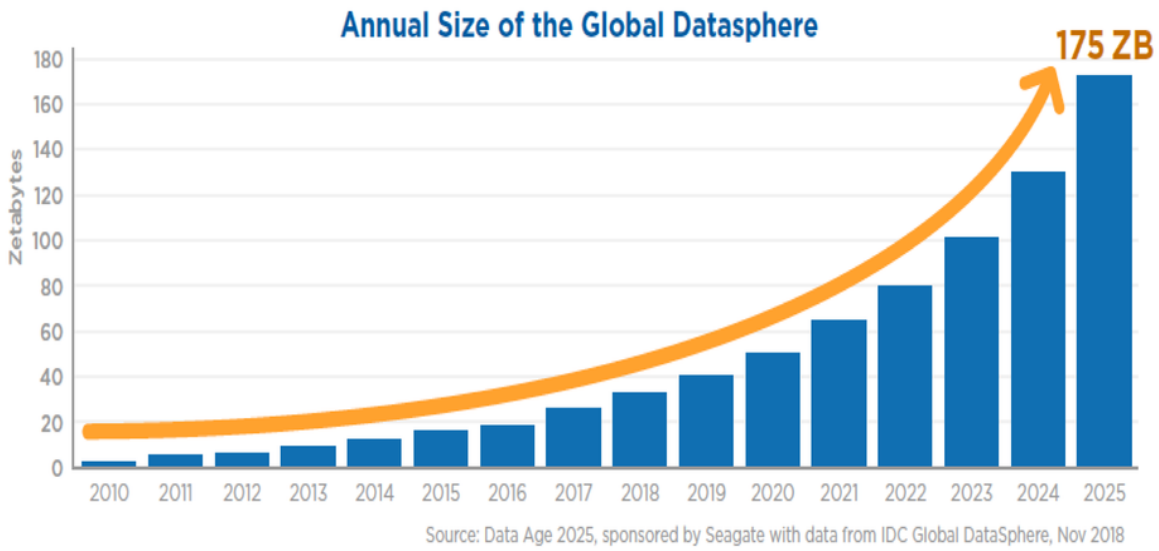
O Azure tem opções de PaaS para NOSQL, como o CosmosDB e suas APIs (modelos de dados) para Cassandra, MongoDB e Gremlin. Outra opção é usar IaaS e hospedar o banco de dados em uma máquina virtual do Azure, seja instalando-o manualmente ou usando um template ARM existente no Market Place do Azure.

Capítulo 5. Soluções de Big Data

5.1. Introdução ao Big Data

Em plena era da Internet das Coisas (IOT), o crescimento do armazenamento e processamento de dados não estruturados e a utilização de soluções de Big Data são cada vez mais uma vertente exponencial e sem volta.

Figura 29 - Crescimento anual do volume de dados produzido pela humanidade.



Fonte: Seagate (2018).

O Gartner define Big Data como “ativos de informações de grande volume, alta velocidade e/ou alta variedade que exigem formas inovadoras e de baixo custo de processamento de informações, que permitem uma visão aprimorada, tomada de decisão e automação de processos”:

> Termo adotado pelo mercado para descrever problemas no gerenciamento e processamento de informações extremas, as quais excedem a capacidade das tecnologias de informações tradicionais ao longo de uma ou várias dimensões.

> Big Data está focado principalmente em questões de volume de conjunto de dados extremamente grandes gerados a partir de práticas tecnológicas, tais como mídia social, tecnologias operacionais, acessos à Internet e fontes de informações distribuídas. Big Data é essencialmente uma prática que apresenta novas oportunidades de negócios. (GARTNER, 2015)

Tendo como gatilho a intensa utilização de redes sociais on-line, de dispositivos móveis para conexão à Internet, transações e conteúdos digitais, e também o crescente uso da computação em nuvem, Big Data, em resumo, pode ser visto como um imenso “conjunto de dados cujo crescimento é exponencial e cuja dimensão está além das habilidades das ferramentas típicas de capturar, gerenciar e analisar dados” (McKinsey Global Institute, 2011):

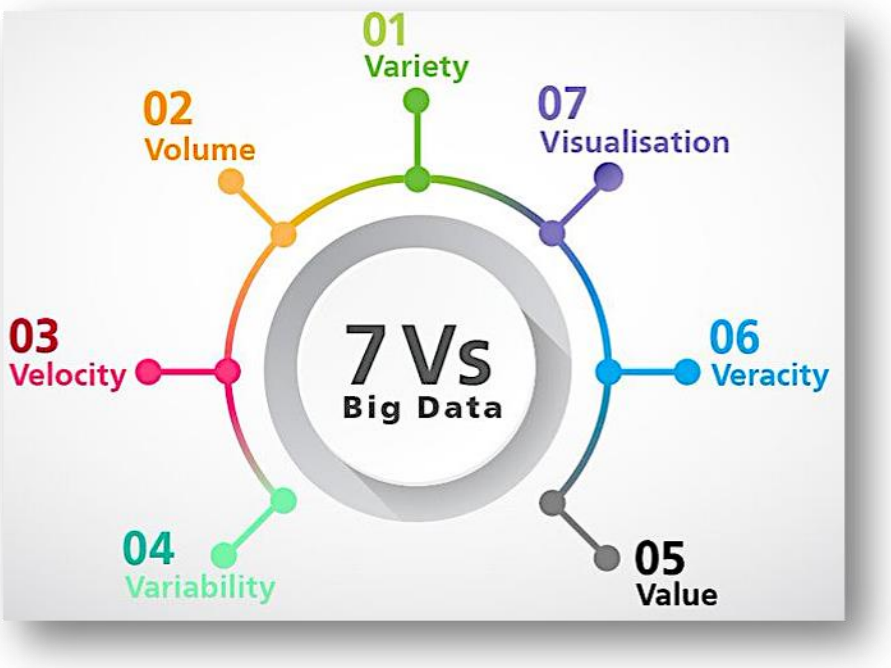
> A intensa utilização de redes sociais online, de dispositivos móveis para conexão à Internet, transações e conteúdos digitais, e também o crescente uso de computação em nuvem tem gerado quantidades incalculáveis de dados. O termo Big Data refere-se à este conjunto de dados cujo crescimento é exponencial e cuja dimensão está além da habilidade das ferramentas típicas de capturar, gerenciar e analisar dados. (McKinsey Global Institute, 2011)

Em sumo, Big Data é um campo ou área que trata de maneiras de analisar, extrair sistematicamente informações ou, de outra forma, lidar com conjuntos de dados que são muito grandes ou complexos para serem tratados por softwares de aplicativos de processamento de dados tradicionais. Big Data é mais que um produto de software ou hardware. É um conjunto de tecnologias, processos e práticas que permitem às empresas analisarem dados que antes não tinham acesso e tomar decisões, ou mesmo gerenciar atividades de forma muito mais eficiente.

Fundamentos de Big Data

De forma macro, os principais fundamentos de Big Data podem ser resumidos nos “7 Vs”:

Figura 30 - Os 7Vs do Big Data.



- **Volume**: a quantidade de dados gerados, que costumava ser medida em Gigabytes agora é medida em Zettabytes (ZB), caminhando para Yottabytes (YB).
- **Velocidade**: velocidade em que os dados são gerados e se tornam acessíveis.
- **Variedade**: de formatos, de tipos (estruturado, não estruturado e semiestruturado) e da natureza (numérica, data, caractere, etc.).
- **Variabilidade**: mesma combinação de dados cujo significado muda constantemente.
- **Veracidade**: garantir que os dados sejam precisos, verdadeiros, fidedignos.
- **Visualização**: formas mais aderentes para visualizar grandes quantidades de dados complexos.
- **Valor**: transformar dados em valor, vantagem competitiva ou otimização operacional.

5.2. Introdução ao HDInsight

Recurso do Azure para processamento e análise de Big Data, que possibilita a criação rápida de clusters de Big Data sob demanda, com escalabilidade horizontal e/ou vertical. Como principais pontos fortes, podemos destacar os de ser uma solução em nuvem de baixo custo para projetos de Big Data e possuir os principais softwares livres da Apache para projetos de Big Data.

Figura 31 - Softwares livres da Apache no HDInsight.

Fonte: Microsoft (2020).

Recurso do Azure para processamento e análise de Big Data, que possibilita a criação rápida de clusters de Big Data sob demanda, com escalabilidade horizontal e/ou vertical. Como principais pontos fortes, podemos destacar os de ser uma solução em nuvem de baixo custo para projetos de Big Data e possuir os principais softwares livres da Apache para projetos de Big Data.

Apache Hadoop

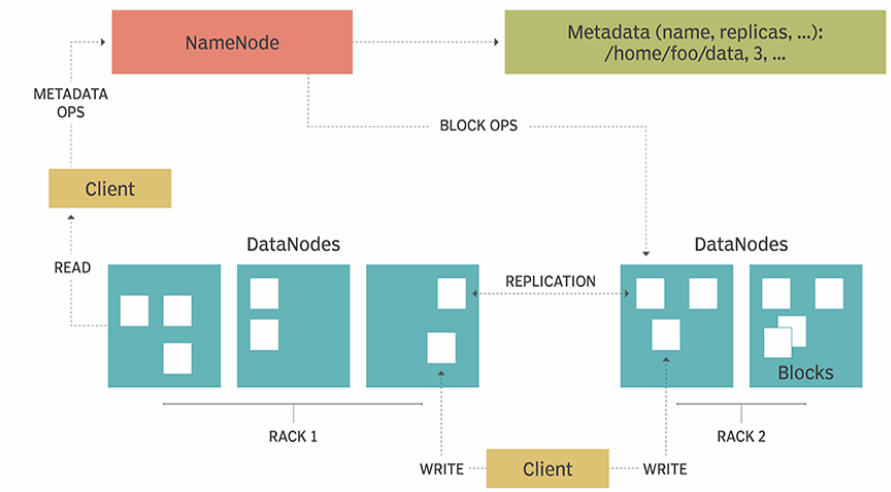
Estrutura que usa HDFS, gerenciamento de recursos YARN e um modelo de programação MapReduce simples para processar e analisar, paralelamente, dados em lote.

O HDFS é um projeto da Apache Software Foundation e um subprojeto do projeto Apache Hadoop (HANSON, 2013). Se tratando da abreviatura para Hadoop Distributed File System, é um sistema de arquivos distribuído, criado com o propósito de armazenar e gerenciar grandes quantidades e volumes de dados.

Em termos de topologia, o HDFS foi planejado para funcionar em cluster, possuindo basicamente dois tipos de servidores: um chamado mais comumente de namenode, que é o nó mestre que gerencia o sistema de arquivos distribuído e

contém todos os metadados do ambiente, e um número indefinido de datanodes, que são os nós escravos que armazenam e recuperam os blocos de dados.

Figura 32 - Arquitetura do HDFS.

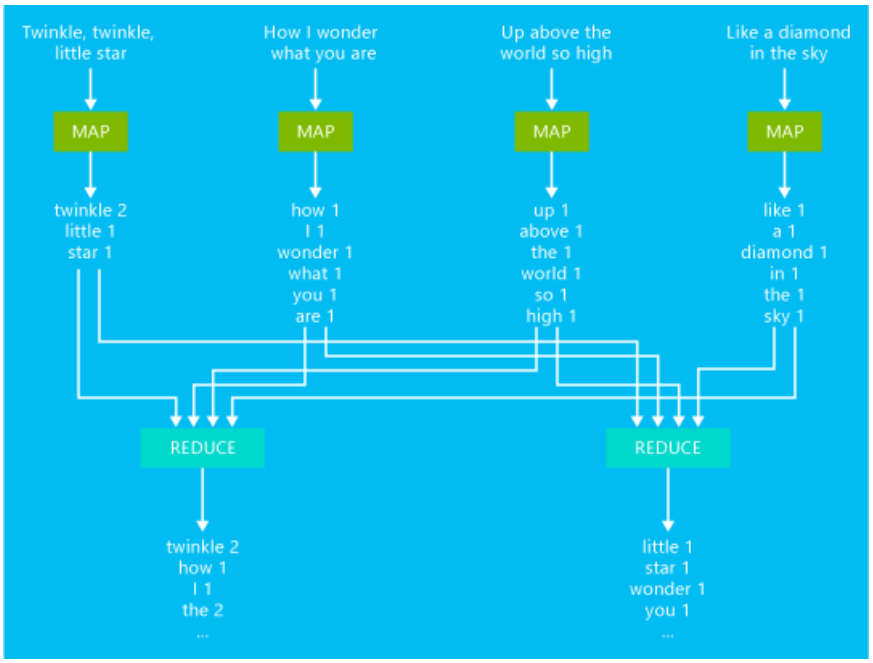


Fonte: Norton Works (2019).

Na grande totalidade dos ambientes de Big Data, é utilizado o HDFS para armazenamento dos dados com volume massivo, ou seja, grandes volumes de dados.

O Hadoop utiliza um modelo de programação MapReduce simples para processar e analisar, paralelamente, dados em lote.

Figura 33 - MapReduce no Apache Hadoop.



Fonte: Microsoft (2019).

Apache Spark

O Apache Spark é uma estrutura de processamento paralelo que suporta o processamento em memória para aumentar o desempenho de aplicativos analíticos de Big Data (MICROSOFT, 2019). Dessa forma, um trabalho (job) do Spark pode carregar e armazenar dados em cache (na memória) e consultá-los repetidamente, ao invés de acessar dados armazenados em disco no HDFS.

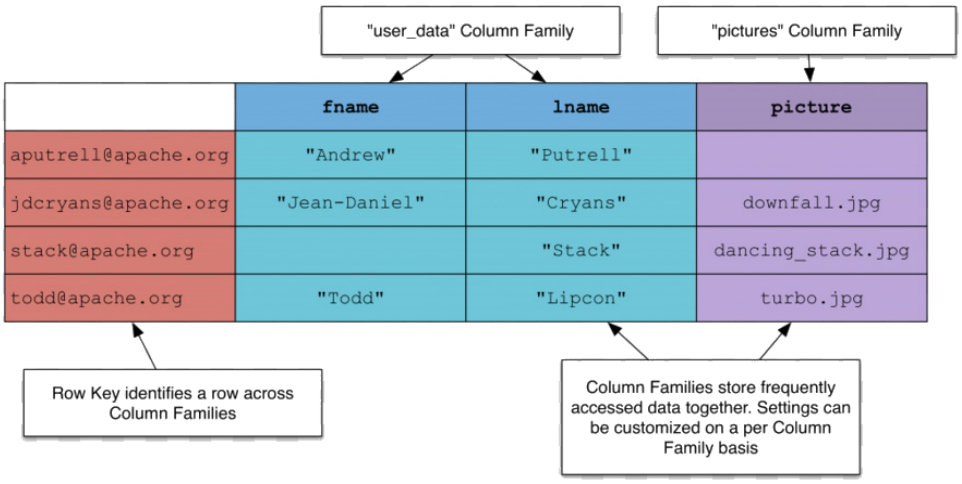
Figura 34 - Spark x MapReduce no HDFS.

Fonte: Microsoft (2019).

Apache HBASE

Banco de dados NOSQL baseado em Hadoop que fornece acesso aleatório e forte coerência para grandes quantidades de dados sem esquema. Construído com base no Google BigTable, os dados são armazenados nas linhas e colunas de uma tabela, e os dados em uma linha são agrupados por família de colunas.

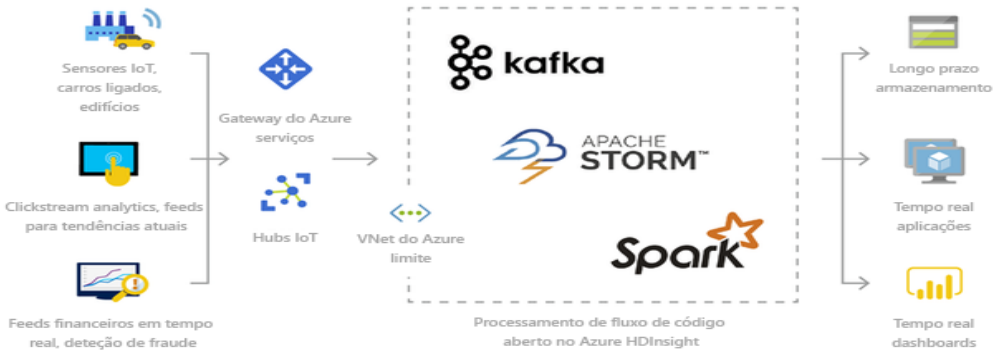
Figura 35 - Famílias de colunas no HBASE.



Apache Kafka

Plataforma de código fonte aberto usada para criar aplicativos e pipelines de streaming de dados. Também fornece funcionalidade de fila de mensagens, o que permite publicar e consumir pipelines de dados.

Figura 36 - Fluxo de processamento com Kafka.

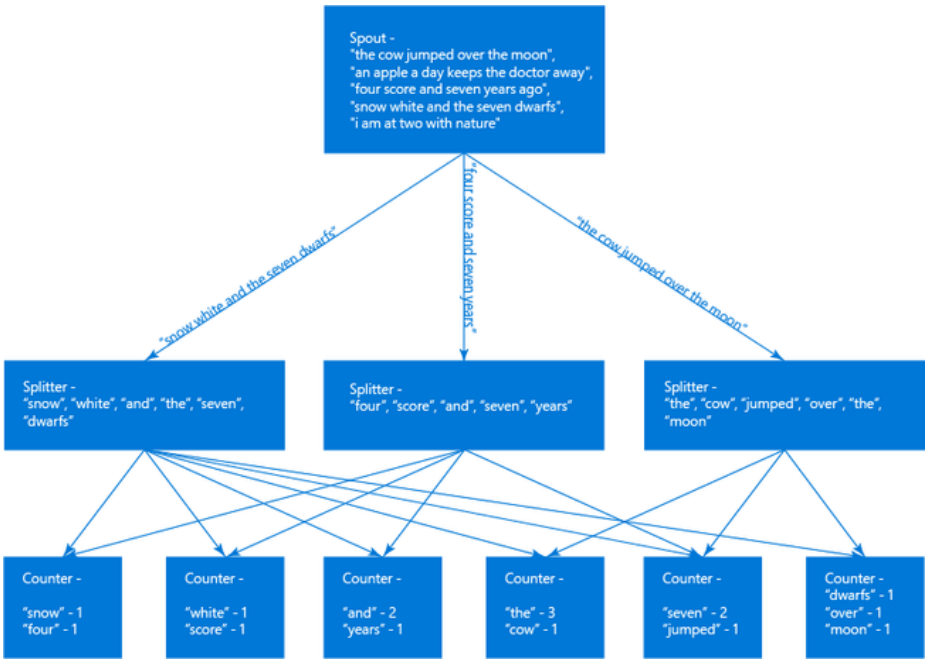


Fonte: Microsoft (2019).

Apache Storm

Sistema de computação distribuído e em tempo real para processar rapidamente grandes fluxos de dados. Processa topologias ao invés de trabalhos MapReduce.

Figura 37 - Processamento de topologia com o Storm.



Fonte: Microsoft (2019).

Apache Hive LLAP

O Hive oferece uma interface semelhante à SQL para consulta de dados em diferentes bancos de dados e sistemas de arquivos integrados ao Hadoop. Comandos tradicionais de SQL são implementados na API Java (HiveQL) para serem executados em dados distribuídos.

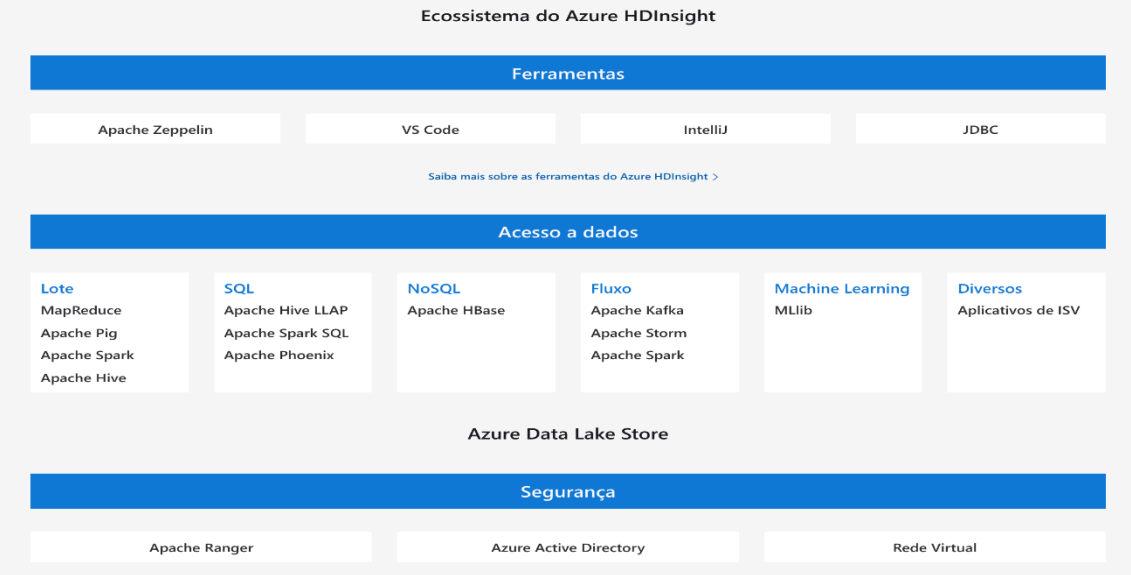
O Hive LLAP é um recurso para cache de dados em memória para consultas do Hive, o que acelera muito o processamento das instruções SQL.

Apache Machine Learning (Ml) Services

Cluster para soluções de aprendizagem de máquina. Fornece aos cientistas de dados, estatísticos e programadores de R/Python o acesso sob demanda a métodos escalonáveis e distribuídos de análise no HDInsigh. Possui um conjunto de modelos e algoritmos de machine learning que podem ser adaptados, conforme as necessidades dos projetos.

Ecossistema Do Azure HDInsight

Figura 38 - Ecossistema do Azure HDInsight.

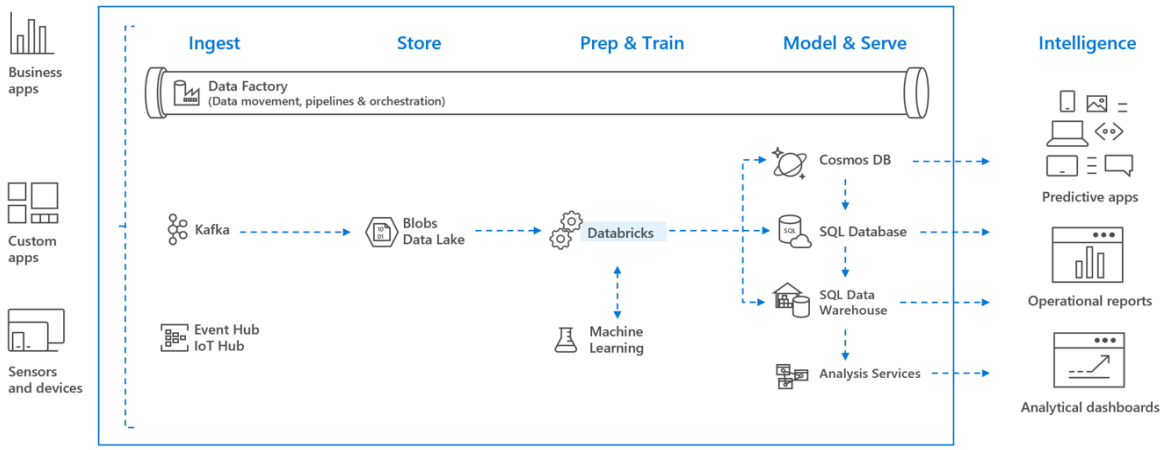


Fonte: Microsoft (2019).

5.3. Introdução ao Azure Databricks

O Azure Databricks é uma plataforma de análise baseada no Apache Spark e otimizada para a plataforma de serviços de nuvem do Microsoft Azure. Permite uma configuração rápida e automatizada de clusters do Spark, fornecendo fluxos de trabalho simplificados, otimizados e performáticos.

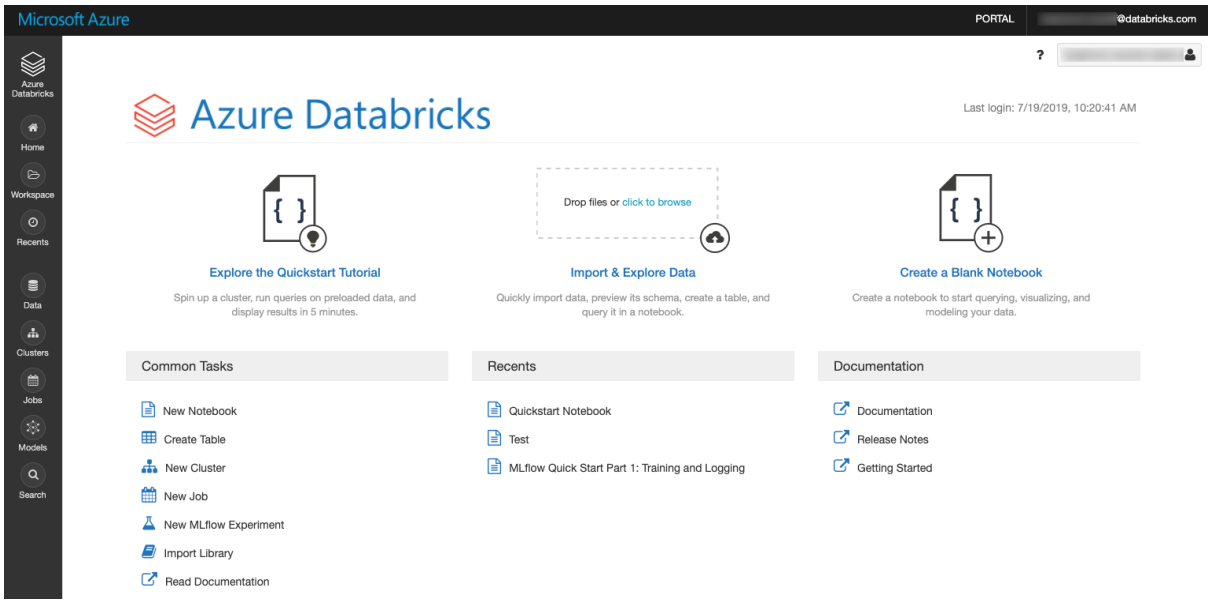
Figura 39 - Pipeline utilizando o Databricks.



Fonte: Microsoft (2019).

Possui um workspace interativo que permite a colaboração entre os cientistas de dados, os engenheiros de dados e os analistas de dados/negócios.

Figura 40 - Workspace do Databricks.



Fonte: Microsoft (2019).

Capítulo 6. Soluções para Pipeline de Dados

6.1. Introdução ao Azure Data Factory

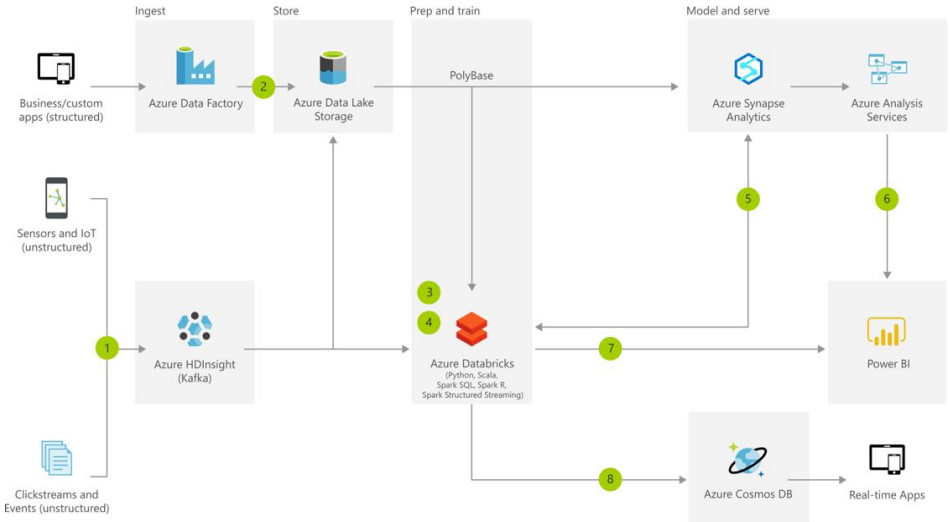
Azure Data Factory é o serviço de integração de dados e ETL (Extração, Transformação e Carga), baseado em nuvem, que permite criar fluxos de trabalho orientados a dados para orquestrar a movimentação de dados e transformá-los em escala. Usando o Azure Data Factory, é possível criar e agendar fluxos de trabalho baseados em dados (chamados de pipelines) que podem ingerir dados de diversas fontes de armazenamentos.

Figura 41 - Pipeline simples com o Data Factory.

Fonte: Microsoft (2019).

Com o Data Factory, é possível também criar processos ETL complexos que transformam dados com fluxos de dados, ou usando serviços de computação, como Azure HDInsight Hadoop, Azure Databricks e Azure Synapse Analytics.

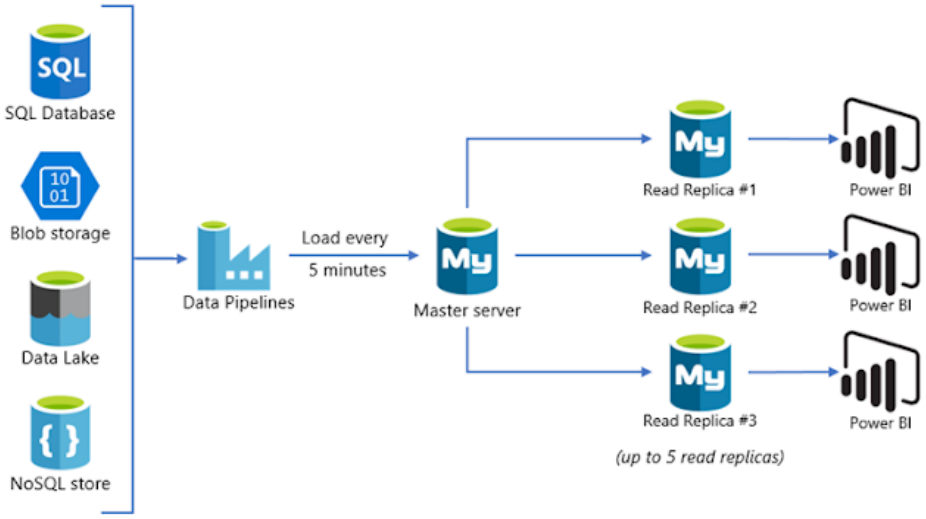
Figura 42 - Pipeline do Data Factory com HDInsight, Databricks e SQL DW.



Fonte: Microsoft (2019).

Além dos pipelines executados manualmente, é possível também criar e agendar fluxos de trabalho orientados a dados para orquestrar a movimentação de dados e transformá-los.

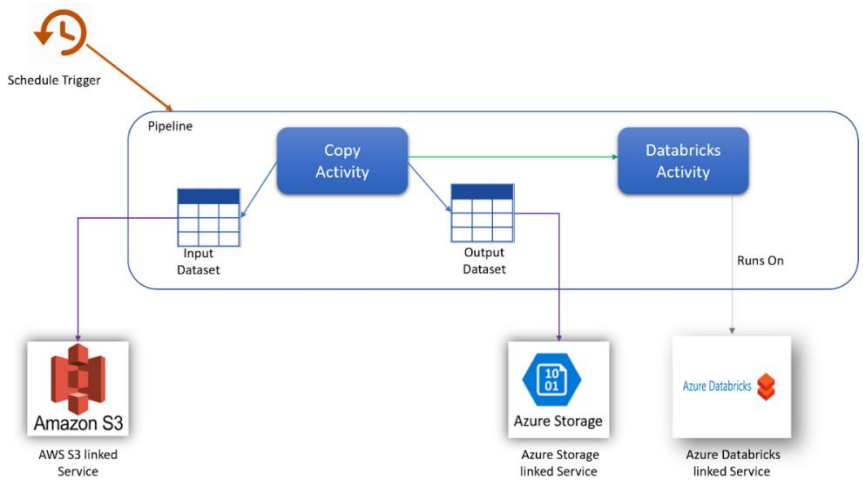
Figura 43 - Pipeline Agendado (Triggered Pipeline).



Fonte: Microsoft (2019).

Importante frisar que as fontes de dados do Data Factory podem ser externas ao Azure, como no exemplo abaixo, onde os dados são extraídos de um bucket S3 na AWS.

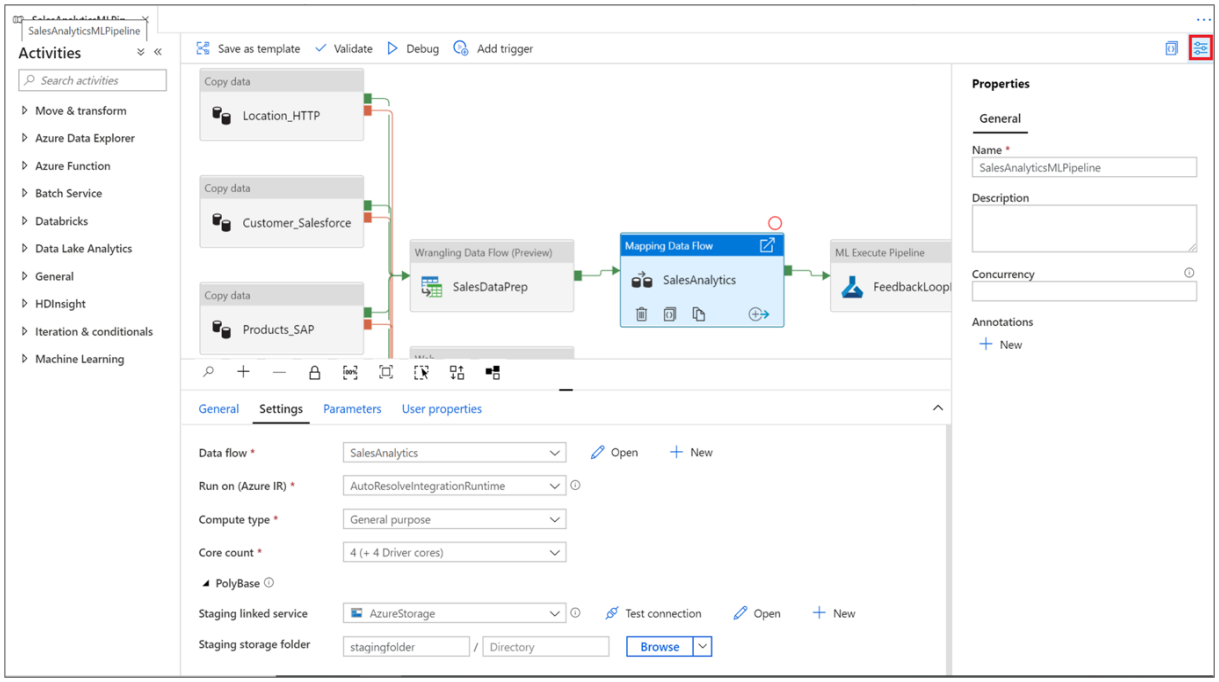
Figura 44 - Fonte de dados externa ao Azure Data Factory.



Fonte: Microsoft (2019).

O ponto auge do Azure Data Factory é permitir a criação de pipelines de forma gráfica, além da forma via linha de comando (código).

Figura 45 - Interface do Data Factory para criação de pipelines graficamente.



Fonte: Microsoft (2019).

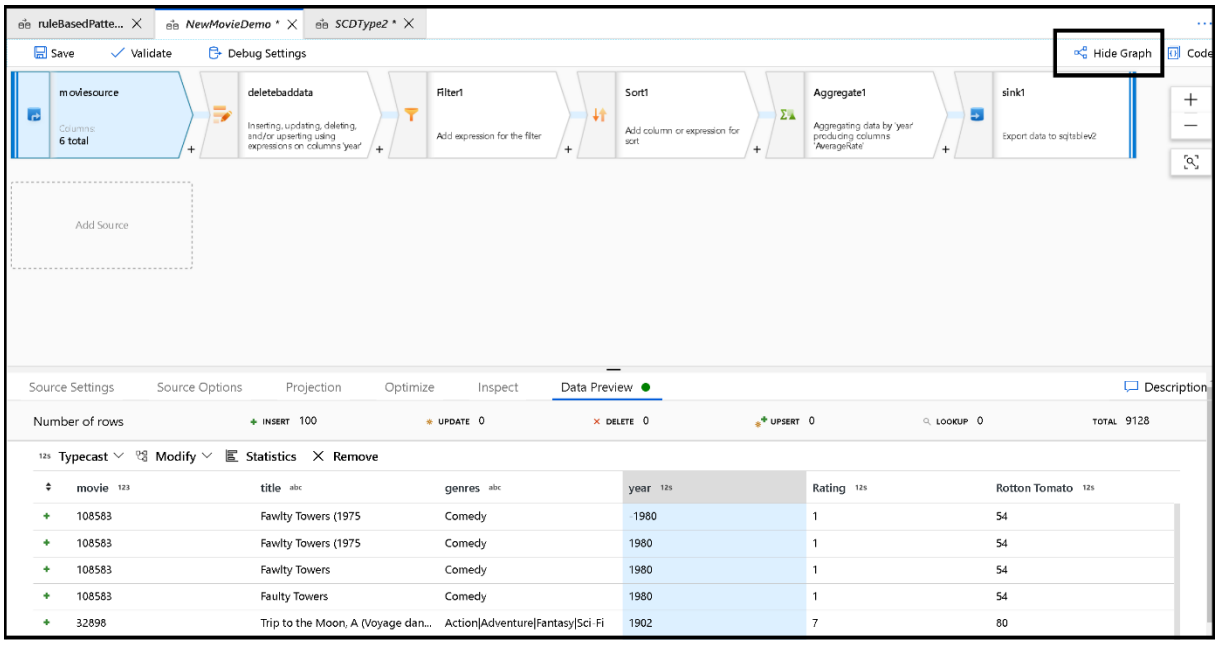
Componentes do Azure Data Factory

- **Atividade:**
  - Representa uma etapa de processamento em um pipeline.
    - Atividade para copiar dados de um repositório de dados para outro;
    - Atividade que executa uma consulta de Hive em um cluster do Azure HDInsight para transformar ou analisar dados;
    - Etc.
  - O Data Factory dá suporte a três tipos de atividades:

- Atividades de movimentação de dados;
- Atividades de transformação de dados;
- Atividades de controle.

- **Pipeline:**
  - Agrupamento lógico de atividades que realiza uma unidade de trabalho. Juntas, as atividades em um pipeline executam uma tarefa.
  - Exemplo: pipeline contém um grupo de atividades que ingere dados provenientes de um blob do Azure e, em seguida, executa uma consulta Hive em um cluster HDInsight para particionar os dados.
  - Pipeline permite gerenciar atividades como um conjunto, em vez de gerenciar cada uma individualmente.
  - Atividades podem operar de modo sequencial ou de forma independente, em paralelo.
- **Mapeamento de Fluxo de Dados (Mapping Data Flow):**
  - São transformações de dados visualmente projetadas no Azure Data Factory.
  - Permitem que os engenheiros de dados desenvolvam lógicas de transformação de dados sem escrever código.
  - O Data Factory executará a lógica em um cluster Spark, autogerenciado pelo Azure, que será ativado e desativado quando necessário.

Figura 46 - Azure Data Factory Mapping Data Flow.



Fonte: Microsoft (2019).

- **Conjunto de Dados (Dataset):**

  – Representam as estruturas de dados nos repositórios de dados, que simplesmente apontam para ou fazem referência aos dados que se deseja usar em atividades, seja como entrada ou saída.

- **Serviço Vinculado (Linked Service):**

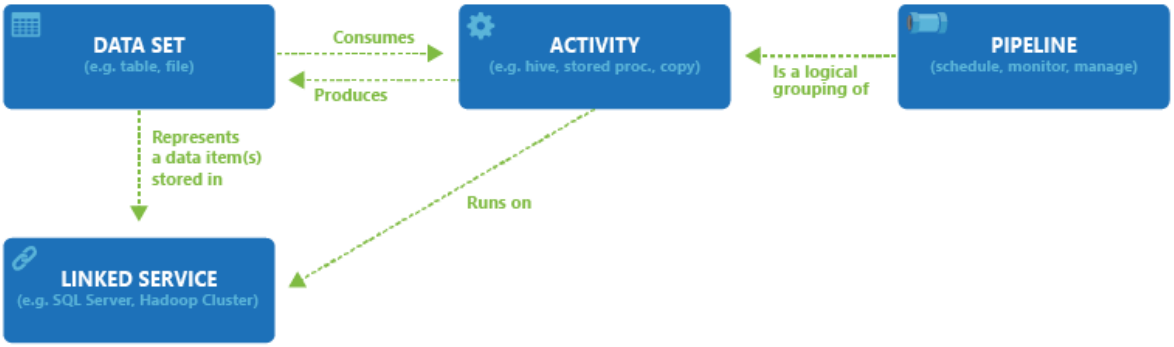
  – Define as informações de conexão necessárias para que o Data Factory se conecte aos recursos externos. Duas finalidades:

    - Para representar um armazenamento de dados: ex. banco SQL/Oracle;

    - Para representar um recurso de computação que pode hospedar a execução de uma atividade: ex. um cluster Hadoop do HDInsight, onde a atividade HDInsightHive é executada.

Resumindo, um serviço vinculado define a conexão à fonte de dados e um conjunto de dados representa a estrutura dos dados. Por exemplo, um serviço vinculado de armazenamento do azure especifica a string de conexão para conectar-se à conta de Armazenamento do Azure (Storage Account), e um conjunto de dados de blob do Azure especifica o contêiner de blob e a pasta que contém os dados.

Figura 47 - Componentes do Azure Data Factory.



Fonte: Microsoft (2019).

Capítulo 7. Soluções de Machine Learning

7.1. Overview do Azure Machine Learning

Aprendizado de máquina (Machine Learning - ML) é uma técnica da ciência de dados que permite que os computadores usem os dados existentes para prever tendências, resultados e comportamentos futuros.

Usando ML, os computadores têm a capacidade de aprender de acordo com as respostas esperadas por meio das associações de diferentes dados, os quais podem ser imagens, áudio, números, etc.

Figura 48 - Tipos de Aprendizagem de Máquina.

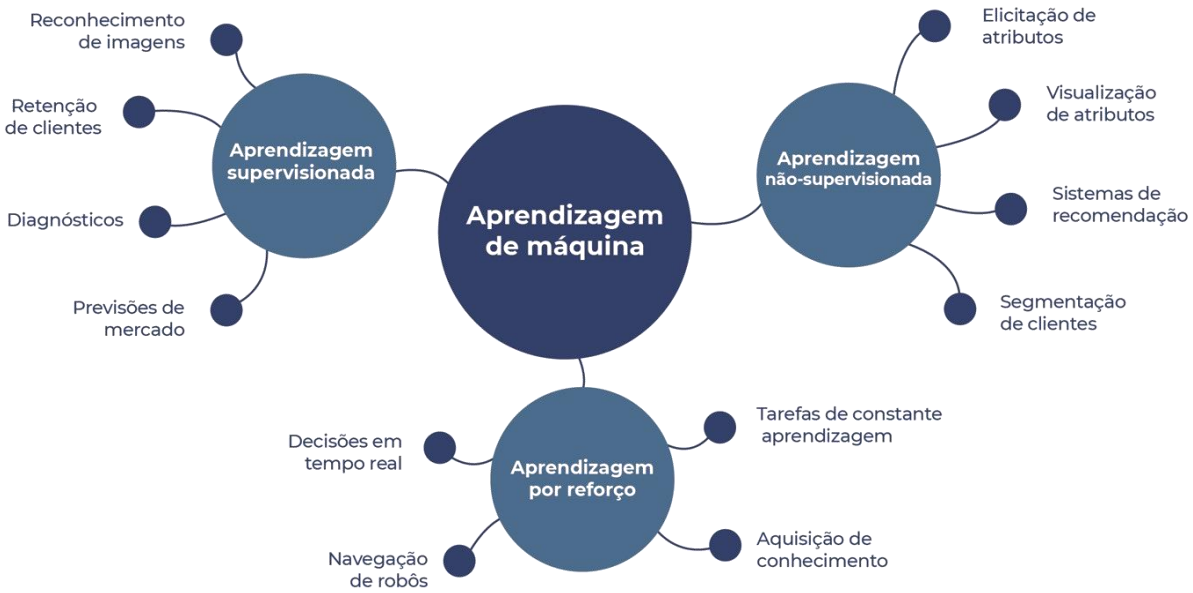
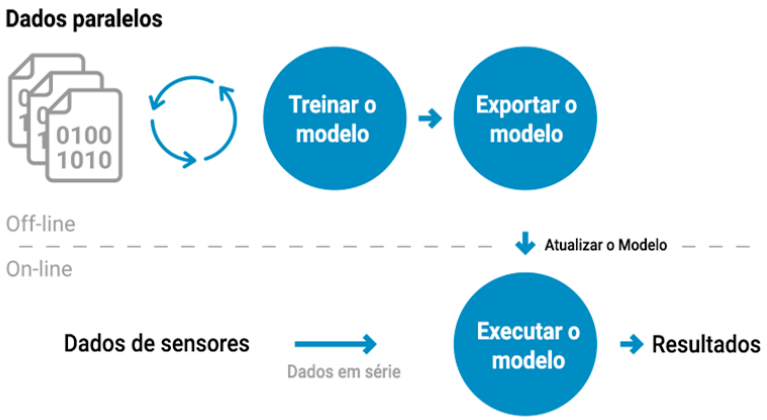


Figura 49 - Exemplo de processo macro de aprendizagem de máquina.



Azure Machine Learning

Ambiente baseado em nuvem que pode ser usado para treinar, implantar, automatizar, gerenciar e rastrear modelos de ML. Pode ser usado para qualquer tipo de aprendizado de máquina, desde ML clássico até aprendizado profundo, aprendizado supervisionado e não supervisionado.

Figura 50 - Machine Learning no Azure Market Place.

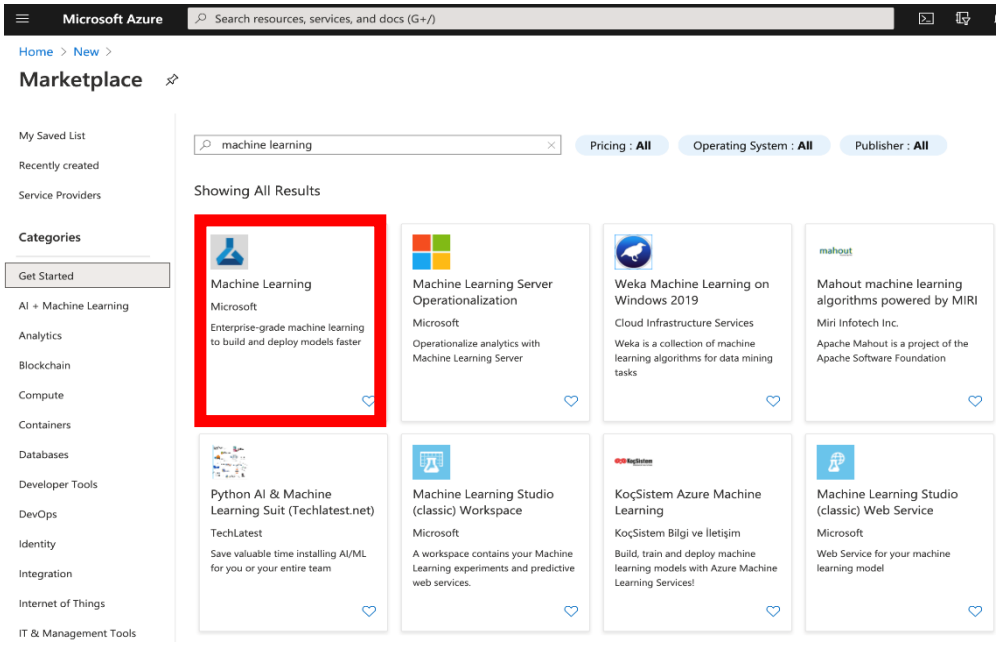


Figura 51 - Azure Machine Learning Workspace.

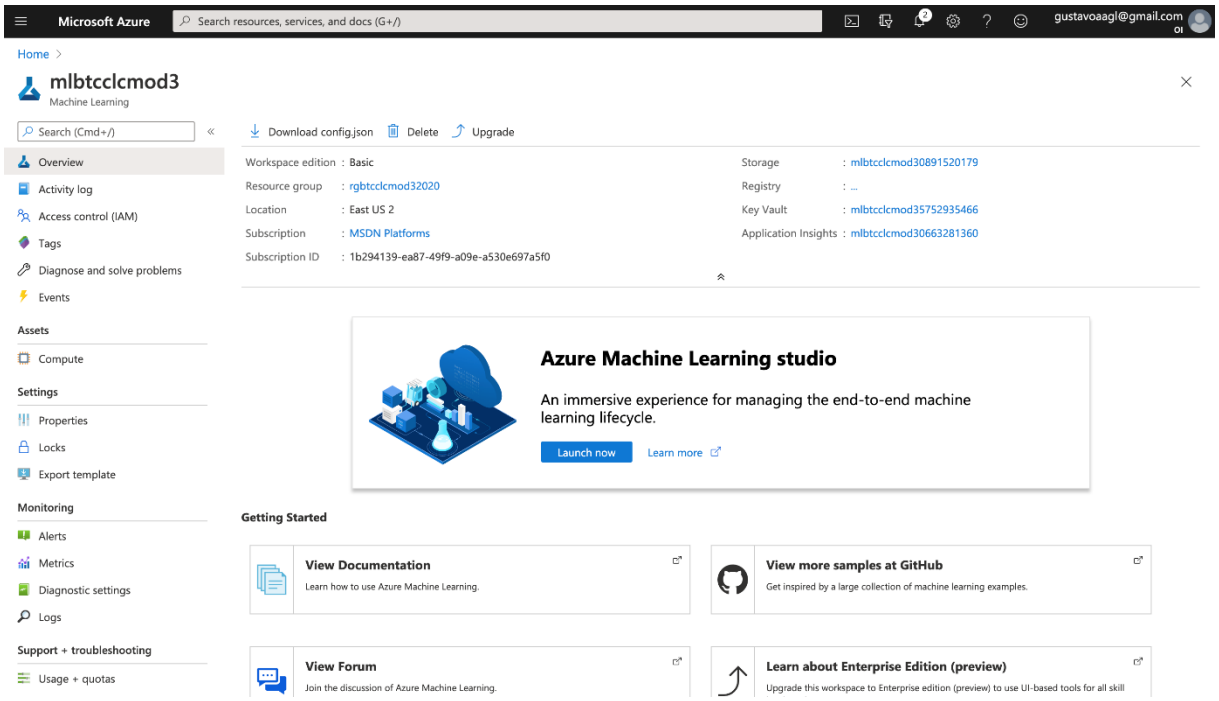


Figura 52 - Azure Machine Learning Studio.

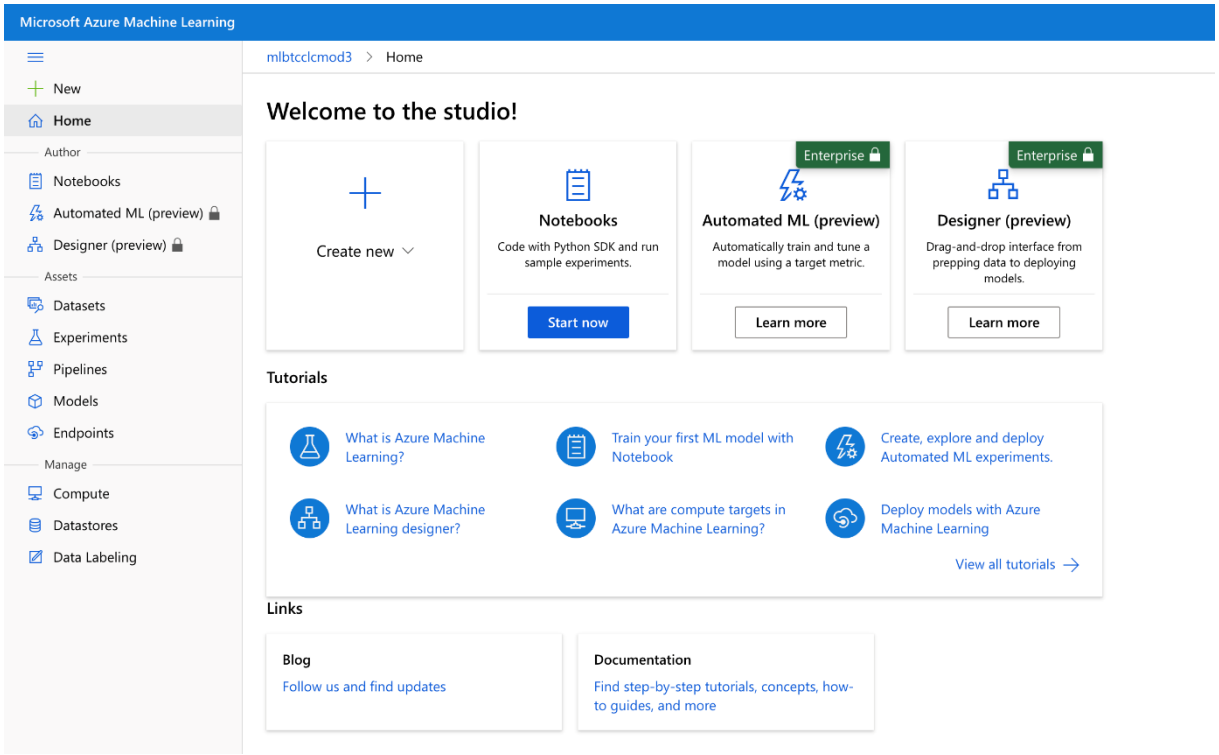
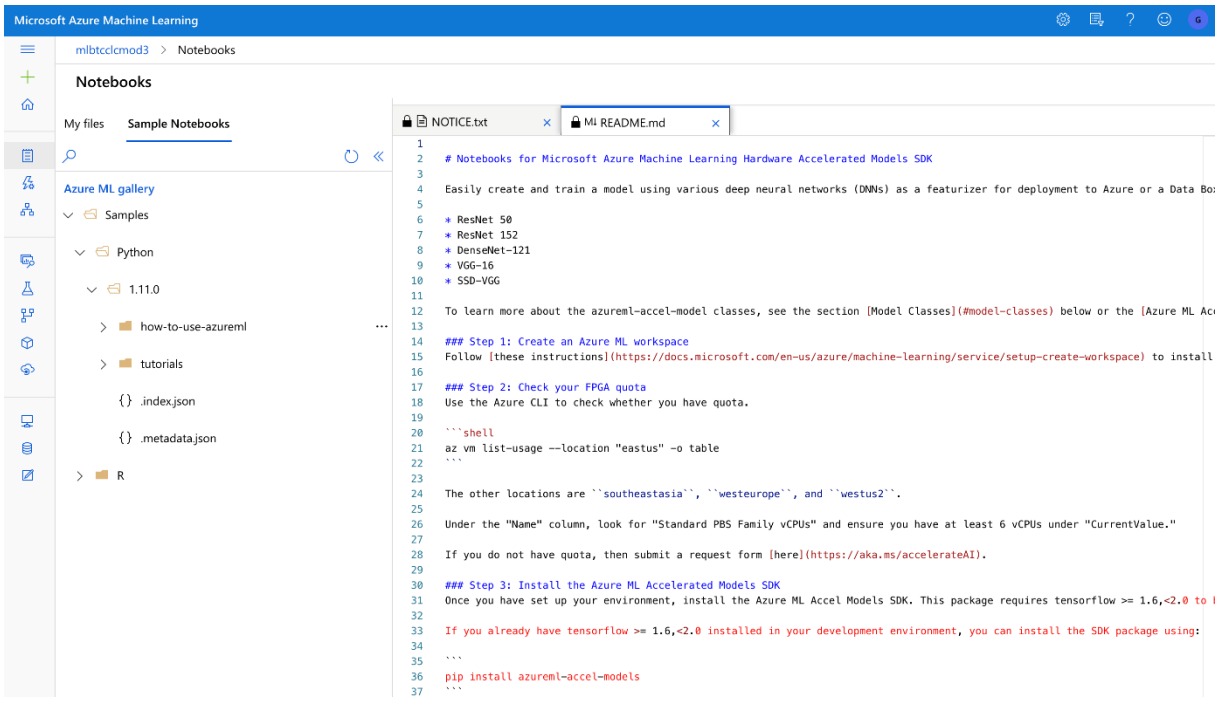


Figura 53 - Azure Machine Learning Notebooks.



Azure Machine Learning Designer

Interface gráfica para preparar dados, treinar, testar, implantar, gerenciar e rastrear modelos de aprendizado de máquina sem escrever nenhum código.

Figura 54 - Azure Machine Learning Designer.

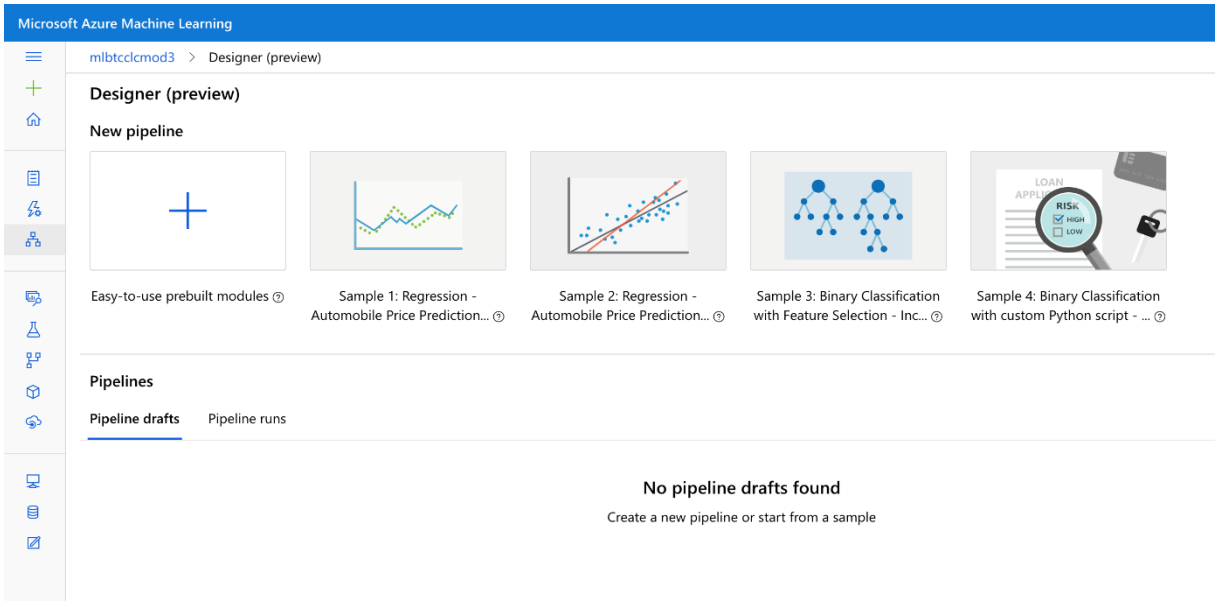
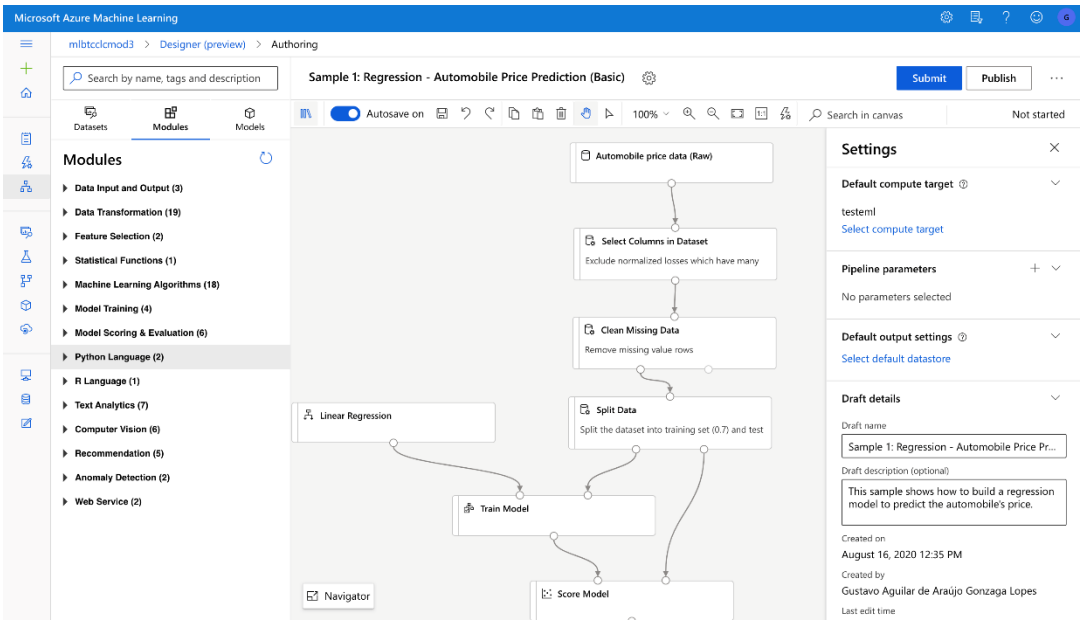


Figura 55 - Pipeline de ML no Azure Machine Learning Designer.



Azure Machine Learning Automatizado

Interface gráfica para construção drag & drop (interface com componentes prontos) de pipelines de machine learning. Realiza a iteração, de forma rápida, entre várias combinações de algoritmos e parâmetros, para ajudar a encontrar o melhor modelo com base em uma métrica selecionada. Disponível somente na assinatura Enterprise, assim como o Azure Machine Learning Designer.

Figura 56 - Azure Machine Learning Automatizado.

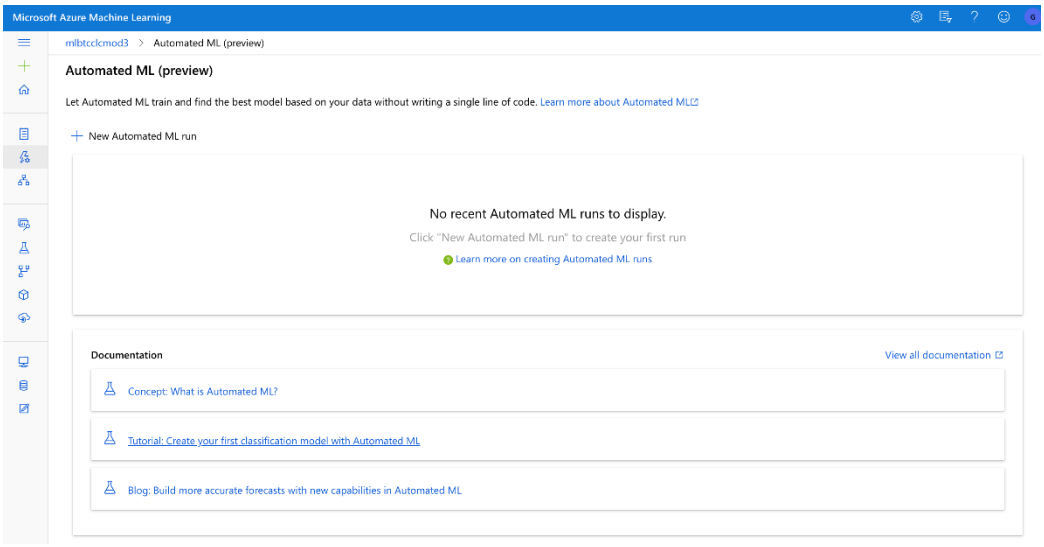
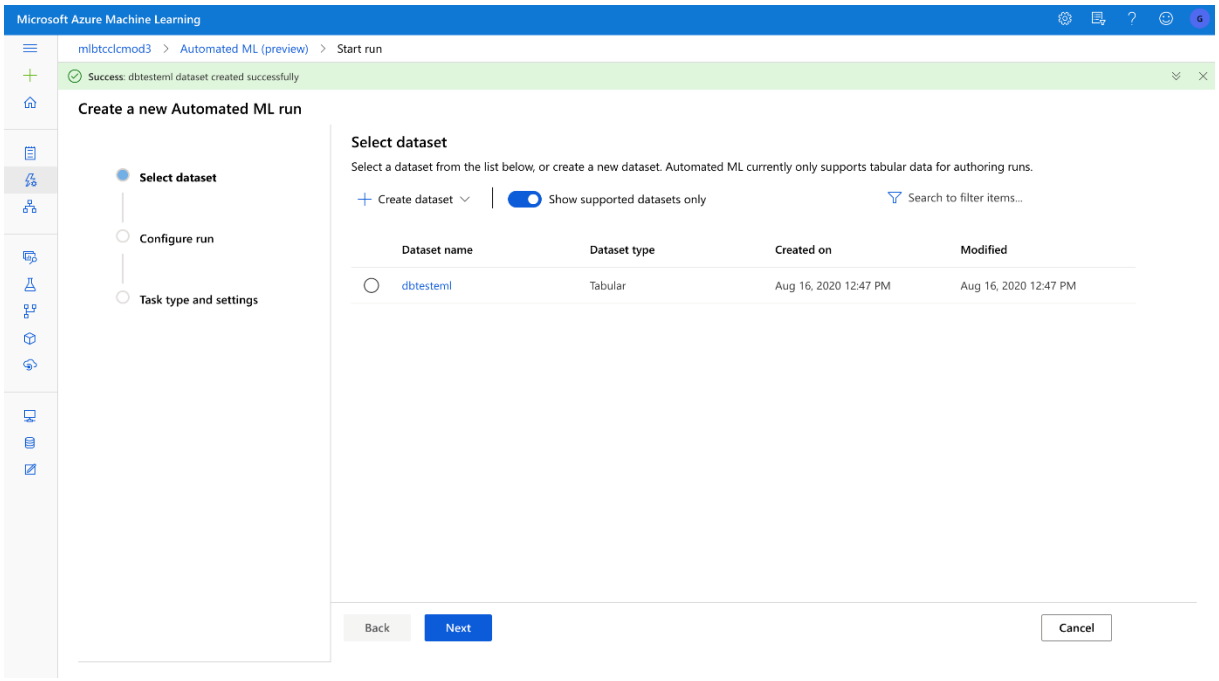


Figura 57 - Interface de criação de Machine Learning Automatizado no Azure.



Referências

MICROSOFT AZURE. *Plataforma de Dados*. 2020. Disponível em: <https://azure.microsoft.com/pt-br/overview/data-platform/>. Acesso em: 11 ago. 2020.