

當資料流入「熱」路徑，這份資料是可以改動的，且可以更新。除此之外，熱路徑約束了資料處理的延遲性（因為需要即時得出結果）。因此，受延遲約束特性影響，熱路徑能夠執行的計算類型僅限於可以快速完成的計算。這意味著此時我們選擇了一個提供近似值結果而非精確答案的演算法。舉個例子來說，算出資料集中特定項目的數量（比如網站訪問者數），你可以將每個資料納入計算（如果數量很多就會造成相當高的延遲），或者可以利用如 HyperLogLog 等演算法求得近似值。「熱」路徑的目標可以概括成：「以沒那麼精確的結果為代價，盡快將資料結果準備好。」

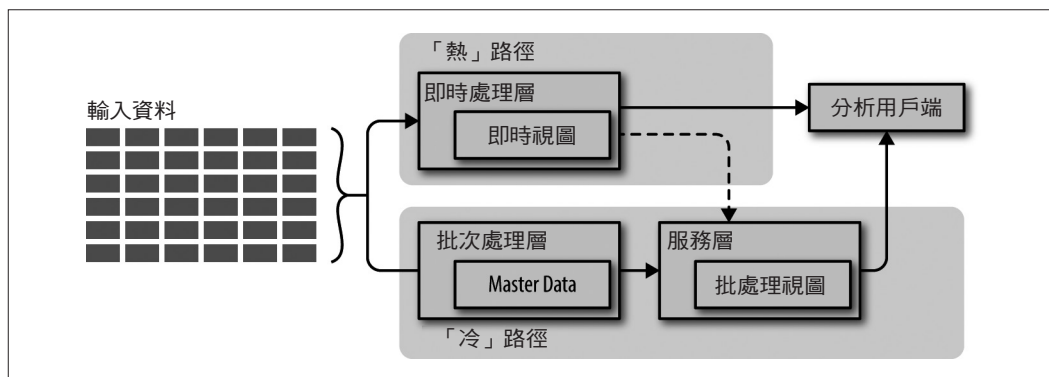


圖 1-2 Lambda 架構將流入管線的所有資料擷取至不可修改的儲存層中，在圖中以 Master Data 標註。這份資料由批次處理層（Batch layer）處理，接著以批處理視圖的格式輸出至服務層（Serving layer）。在即時處理層（Speed layer）輸入資料，進行低／零延遲計算，並以即時視圖的形式呈現。分析用戶端可以根據所需資料的時間急迫性高低，擇一使用即時處理層或服務層的視圖。在某些應用情境中，服務層可以同時承載即時視圖與批處理視圖。

最終，冷熱路徑會匯集於分析用戶端的應用上。客戶必須從中選擇一條擷取分析結果的路徑。客戶可以使用從熱路徑得來，較不準確但即時產出的分析結果，或者使用來自冷路徑，花費較長時間而得的精確結果。客戶在做出決定之時，有一項關鍵：當熱路徑在極短時間內得出結果時，冷路徑尚未產出分析結果。換個角度來想，熱路徑在一小段時間內所得出的結果，最終會被來自冷路徑，更加準確的分析結果所更新。從而得出熱路徑必須處理最小化資料量的效果。

創造 Lambda 架構的動機由來可能相當令人驚訝。沒錯，創造一個能夠即時處理資料的簡單架構很重要，但它問世的原因是為了提供人性化的容錯性。事實上，現在的科技發展到能夠確實保存所有原始資料的階段，Lambda 架構理解即使在生產中也會

Azure 匯入／匯出服務

匯入／匯出服務可以幫助你運送高達 6TB 硬碟容量的資料到當地的處理中心，而處理中心可以從你的硬碟複製資料，再以高速內部網路將資料安全地傳輸到指定的 Azure Blob 儲存體中，完成後再將硬碟寄回。當你必須寄送多於 6TB 的資料時，可以依需要寄送多張硬碟。至於費用，Azure 會向你收取每台硬碟 80 美元的固定費用，同時需要支付運送費用。



供應地區

在本書撰寫之時，除了澳洲、巴西及日本之外，大多數 Azure 地區皆可使用 Azure 匯入／匯出服務。

使用匯入／匯出服務加載資料的大致流程如圖 2-2 所示，我們將在接下來的步驟一一說明。

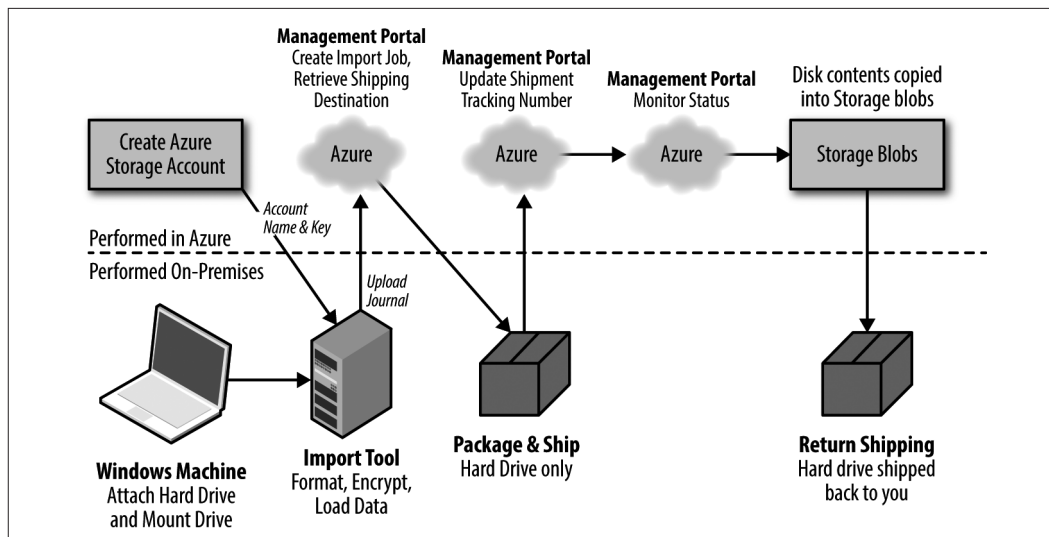


圖 2-2 使用匯入／匯出服務將資料運送到 Azure 的大致流程。

1. 建立 Azure 儲存體帳戶，記住帳戶名稱和密碼。
2. 將硬碟連接到你的 Windows 電腦。使用 WAImportExport 工具（檔案名為 *WAImportExport.exe*），在你的硬碟上啟用 BitLocker 加密，將檔案複製到硬碟上，接著準備好有關該工作的中繼資料檔案。

有了 Azure Files 元件，你可以從支援 SMB 3.0 的本地用戶端服務（基本上是 Windows 8 及 Windows Server 2012 以後的版本）安裝網路分享。不過，這裡有一個隱憂：出於安全考量，許多網路服務供應商屏蔽了 SMB 所要求的 TCP 端口 445 埠。換句話說，雖然 Azure Files 可幫助你透過 SMB 在本地電腦和 Azure Files 之間建立連接，但你的 ISP 可能會阻斷連接。這個不利因素常常排除掉使用 SMB 傳輸的選項。

混合式連接與 Azure Data Factory

混合雲此一術語，代表能夠在本地資料中心與雲端之間，支援雙向通訊的服務。一種建立上述連接的方式為採用站對站網路，我們將在下一節討論。另一種方式則是使用混合式連接，以下將會介紹。

混合式連接的概念圍繞在 Azure 中運行的服務需要存取僅在本地可用的服務或資源，並且只需要出站 TCP 或 HTTP 連接，這樣做對網路配置的影響最小。

通常會進行如下設定：

1. 選定一個可與 Azure 進行混合式連接的服務。
2. 透過入口網站下載安裝程式到本機電腦。
3. 安裝完成後，回到入口網站設置代理的通訊目標（例如，SQL Server 的主機名稱和連接埠，或是檔案伺服器的 IP 位址。）
4. 設置完成後，使用該代理的 Azure 服務即能與本機電腦通訊。

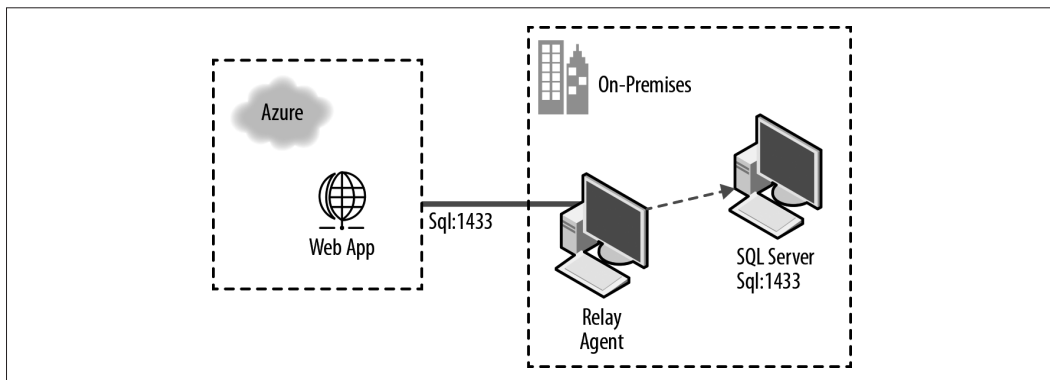


圖 2-50 使用 Azure 服務的網路應用（app）與本機電腦運行的 SQL Server 的混合式連接示例。

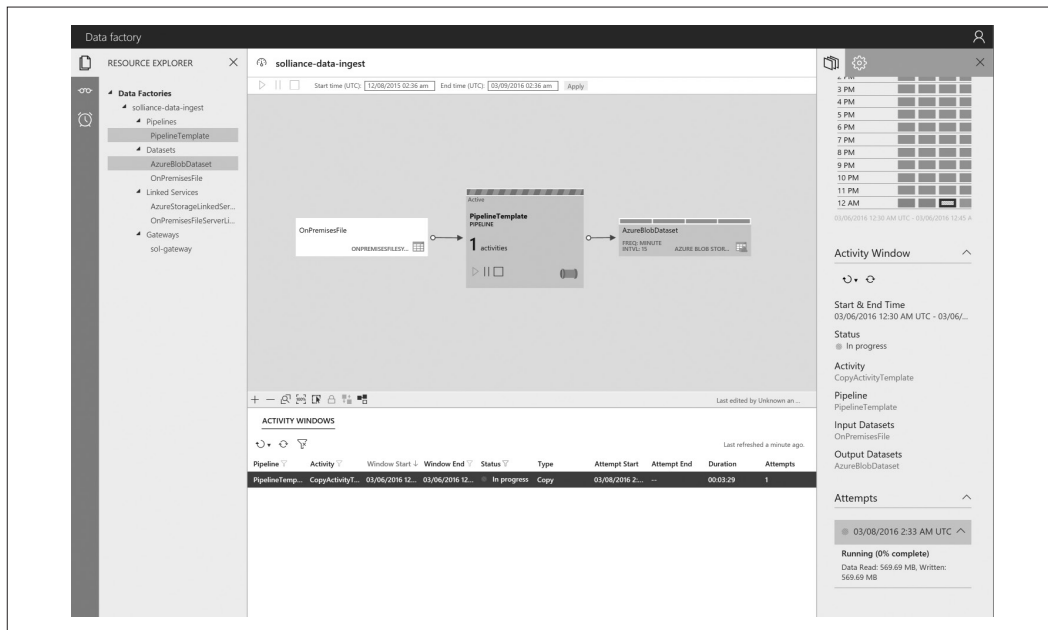


圖 2-63 顯示管線活動的 Data Factory Monitoring App。

從檔案共享擷取資料到 Azure Data Lake Store

上一節我們介紹了本機檔案連結服務與資料集，基於上述內容，將展示如何將資料傳輸到 Azure Data Lake Store。

在進入這一節內容之前，你應該已經建立好 Azure Data Lake Store。詳細設定過程將會在第三章進行介紹，在此你可以先利用 Azure 入口網站建立一個。

1. 點選「新建」。
2. 選擇「資料+儲存」。
3. 選擇 Azure Data Lake Store。
4. 提供名稱並選擇訂用帳戶、資源群組與區域。
5. 點擊「建立」。

塊。當一個資料區塊被儲存，在預設情況下會將其複製到叢集中三個不同磁碟上。這在發生故障時提供了可用性，同時支援並行計算，因為這時有更多的資料副本可供讀取（圖 3-11）。

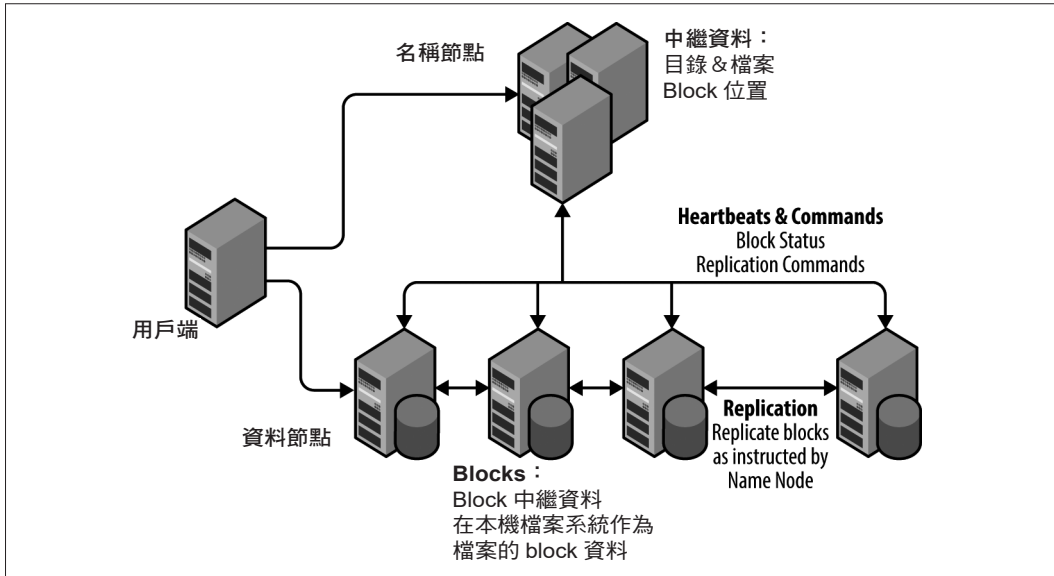


圖 3-11 HDFS 拓撲結構圖。

認識了這種拓撲結構，我們來介紹一下用戶端應用程式如何讀寫 HDFS 的關鍵所在。

從 HDFS 讀取時，使用 HDFS API 的用戶端應用程式向檔案的名稱節點發出請求，然後接收資料節點和區塊列表以便讀取檔案。用戶端接著從每個資料節點請求相應資料區塊（圖 3-12）。

在寫入 HDFS 時，用戶端應用程式首先與名稱節點通訊並檢索即將寫入的資料節點列表。然後，用戶端從列表中的第一個資料節點接收一個區塊列表，並利用這些區塊開始寫入檔案（圖 3-13）。

雖然這並不會讓你搖身一變成為專業的 HDFS 管理人員，但希望上述說明有助於理解 HDFS 如何運行，並清楚解釋如何實現線性擴展、並行讀取和高可用性等重要優勢。你將在下列各節內容發現，在 Azure 服務中，比起 HDFS 本身，你會更加關注構建於 HDFS 之上的各種服務。

如何在 Azure 中使用 HDFS ？

在 Azure 中，有幾種使用 HDFS 的方式。最顯而易見的方式即是，配置 HDInsight 叢集並使用與其一同配置的 HDFS（圖 3-14）。在這種情況下，連接到叢集虛擬機器的磁碟提供了實際的儲存空間。這種方式的缺點是在存取資料之前必須先運行叢集；為了提供儲存，你需要為運行計算支付成本（不管是金額還是啟動叢集所耗費的時間）。這是因為如果關閉叢集，則使用叢集而建立的名稱節點和資料節點也會一併關閉，如果沒有這些節點，則無法存取由本機 HDFS 儲存的資料。透過在叢集外進行外部儲存，可以使多個叢集和應用程式同時存取資料。出於上述原因，很少採用叢集本地儲存方式。

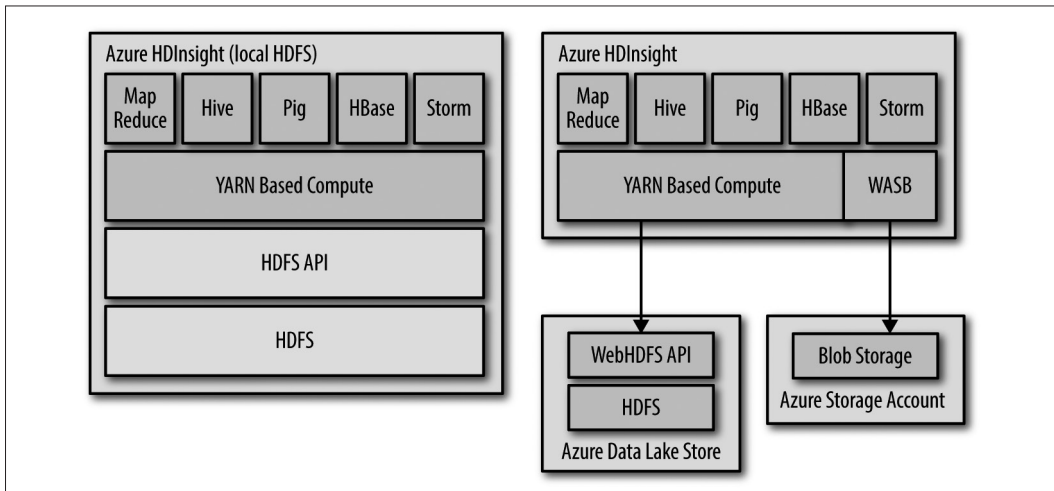


圖 3-14 在眾多 Azure 服務中「出現」的 HDFS。

外部儲存的選項是使用與 HDFS 兼容的儲存體，例如 Azure Blob 儲存體或 Azure Data Lake Store 所提供的儲存體。例如，在 HDInsight 叢集中，你的應用程式可以使用 *wasb:* 方案，透過 HDFS API 存取實際儲存在 Azure Blob 儲存體中的檔案。Windows Azure Storage Blob (*wasb:*) 是一個基於 HDFS API 的擴充程式；從技術層面來看，它由 *hadoop-azure* 模組提供，該模組是 Hadoop 發布版的一部分。

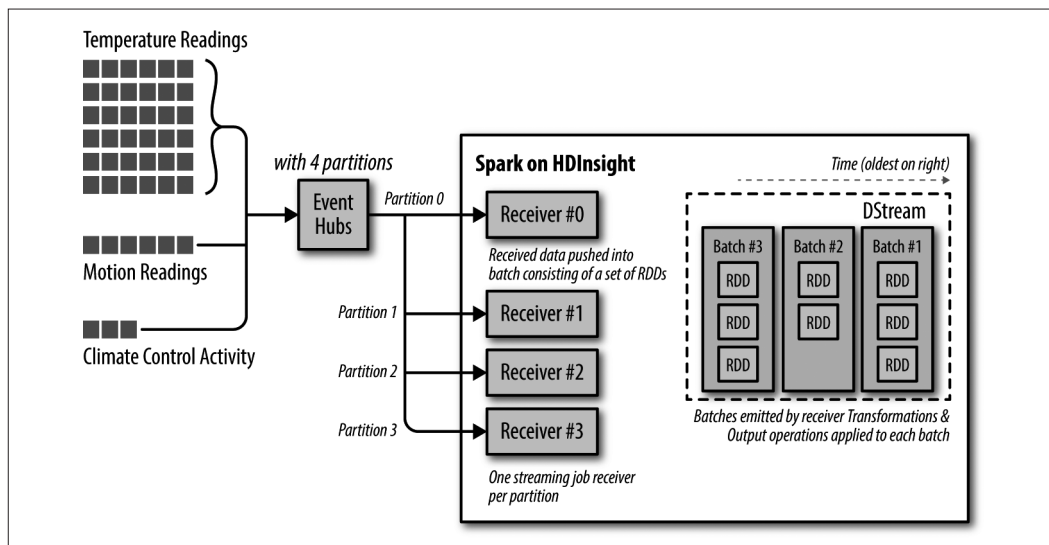


圖 5-2 從 Event Hub 取用事件，並以 DStream 格式由 Spark 串流處理。

Spark 串流應用程式是長時間執行的應用程式，其會接收來自內嵌來源的資料，接著套用轉換以處理資料，然後將資料推送至一個或多個目的地。每個 Spark 串流應用程式包含下列步驟：

1. 建立一個指向 Spark 叢集的 `StreamingContext`。
2. 從 `SparkContext` 建立 `StreamingContext`，並定義批次間隔（比如 2 秒、10 分鐘）。
3. 使用 `StreamingContext`，從輸入來源建立輸入 `DStream`。
4. 將轉換套用至 `DStream`，實作串流運算。
5. 藉由套用輸出作業，將轉換結果推送至目的地系統。
6. 引動 `StreamingContext.start`，啟動串流應用程式。
7. 使用 `StreamingContext.awaitTermination()`，等待處理（自動或因出錯而）終止。或者使用 `StreamingContext.stop()` 手動終止處理。

建立串流應用程式後，可以在本機電腦或在 Azure HDInsight 上的 Spark 叢集中運行。

接下來的內容將討論 Spark 串流應用程式的實作、部署和運行等詳細步驟。

- Parquet
- JSON
- ORC
- Text 檔案
- Hive 資料表
- JDBC 來源



Spark Packages

除了內建資源以外，還有許多可用的資料來源。請查看 Spark Packages 網站 (<https://sparkpackages.org/>) 了解可與其他資料來源進行互動的函式庫。

以 HDInsight 叢集來說，你可能會從與叢集綁定的 Azure 儲存體帳戶、附加的 Azure 儲存體帳戶或 Azure Data Lake Store 中讀取一般檔案（參見圖 6-14）。

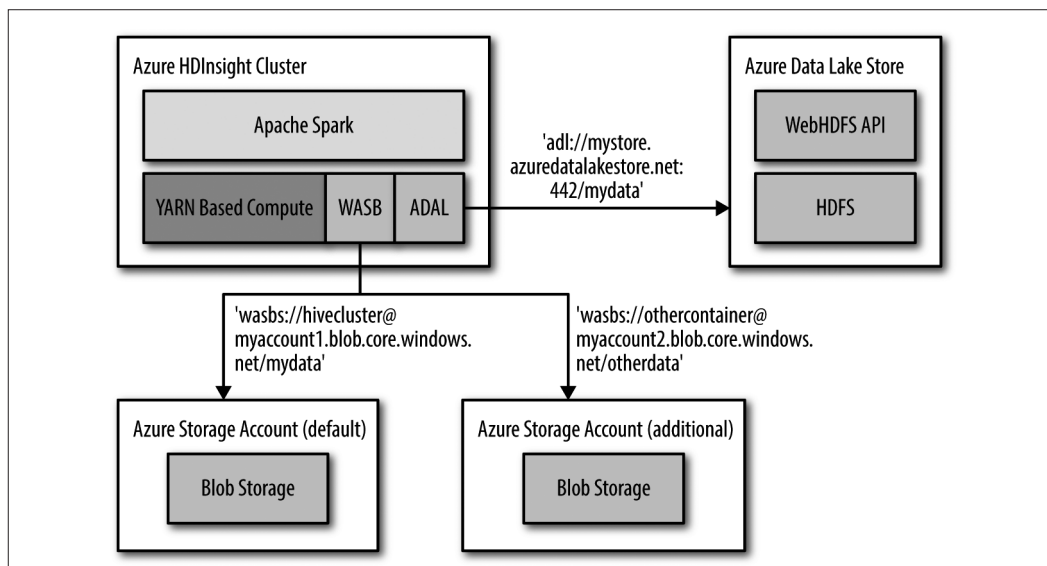


圖 6-14 範例：以 Spark on HDInsight 存取儲存於 Azure 的檔案。

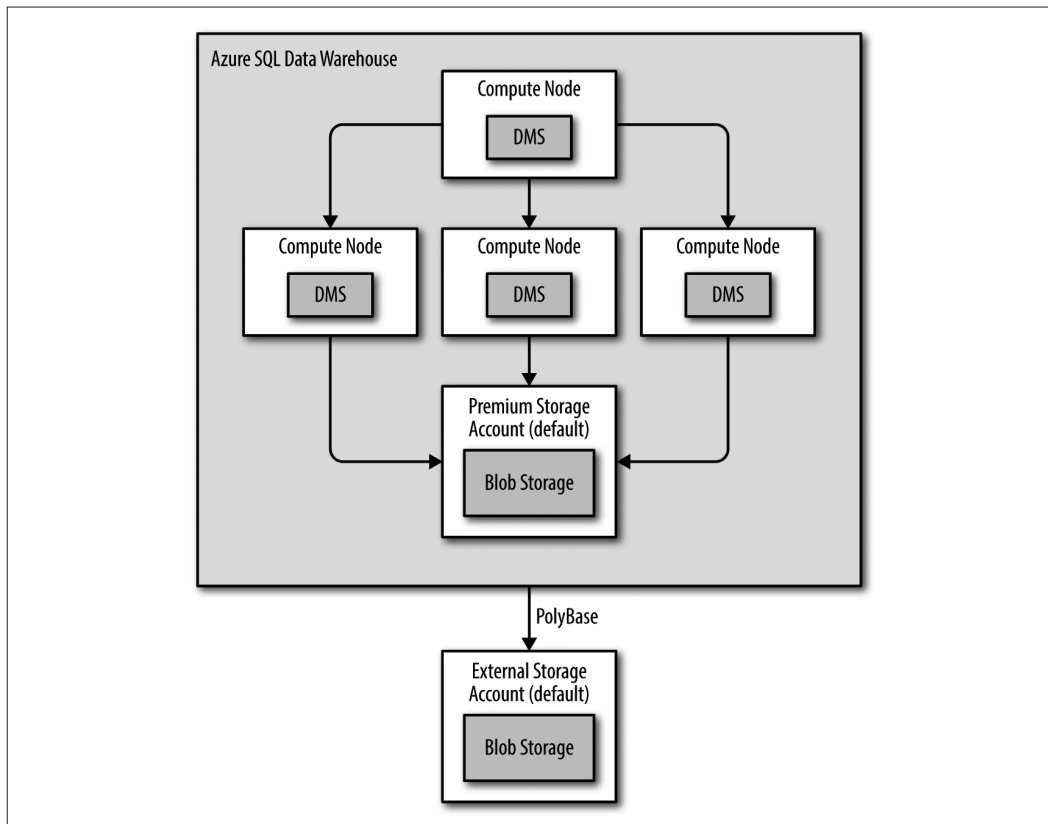


圖 6-15 SQL Database 的主要組成元素。

利用 SQL Database，將 CPU、記憶體、網路和 IO 組合到稱為「資料倉儲單位 (DWU)」的計算規模單位中。運行叢集時，可以增加或刪除 DWU 來調整可用資源。

除了外部和內部資料表外，SQL Database 還支援許多你可能希望的傳統 SQL Server 功能，包括資料表上的索引（叢集和非叢集的 B- 樹狀索引，以及叢集列索引）、暫存資料表、分割資料表、儲存過程、用戶自定義函數（僅限返回 scalar 值）、非實體化視圖、資料庫模式和資料庫。

使用 SQL Database

SQL Database 提供 PolyBase 功能，可以從 Blob 儲存體加載一般檔案資料。SQL Database 具備 Extract、Load 和 Transform 等功能，可從 Azure 儲存體擷取資料，將資料加載到 SQL Data Warehouse 的 Premium Storage 中，並根據需執行轉換。

不同於其他處理解決方案的另一特點是，Azure Data Lake Analytics 定義外部資料表和受管理資料表（內部資料表）的概念。本文撰寫之時，外部資料表被用來查詢儲存在 Azure SQL 資料庫、Azure SQL Database，以及在 Azure 虛擬機器中運行的 SQL Server 的資料。

你可以使用 U-SQL 查詢儲存在 Azure Blob 儲存體和 Azure Data Lake Store 中的檔案，不過，這些檔案被定義為封裝在 U-SQL 視圖或 USQL 資料表值函式中的查詢—不像其他選項的外部資料表的呈現方式。

Data Lake Analytics 將內部資料表稱為受管理資料表。就像 SQL Database 一樣，受管理資料表「擁有」資料。表格定義（中繼資料）以及表格資料都透過中繼資料系統進行管理。受管理資料表的資料被儲存在預設的 Azure Data Lake Store 中，在配置時與 Azure Data Lake Analytics 綁定。請參見圖 6-18。

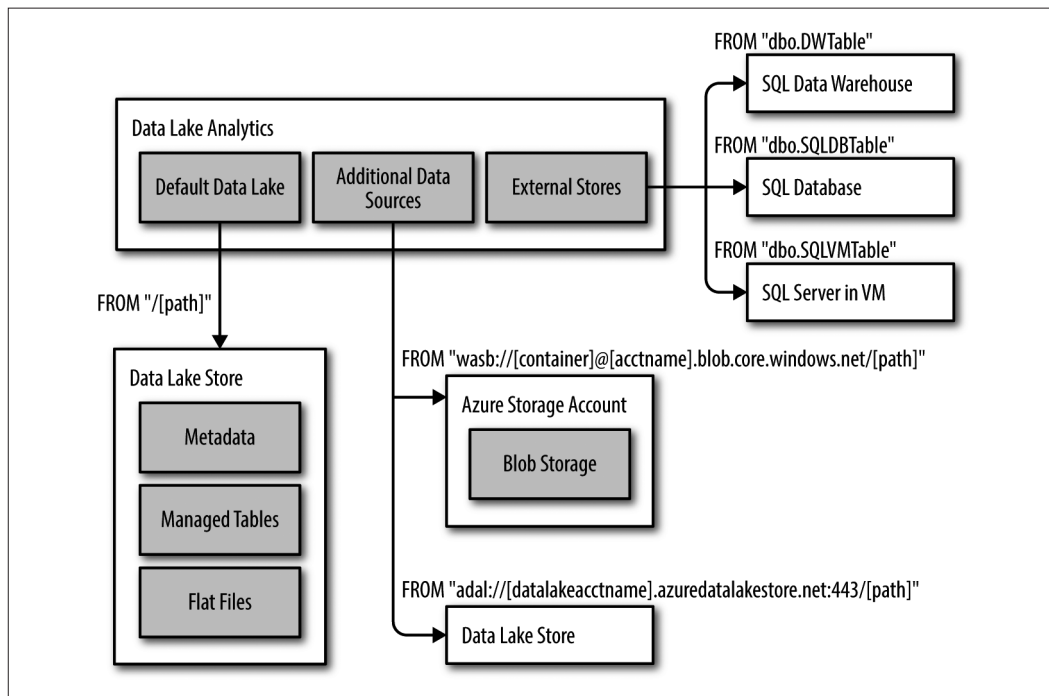


圖 6-18 Data Lake Analytics 概觀，以及用在 U-SQL 腳本中定位來源的 FROM 子句範例。

在上述查詢中，**PARTITION** 子句將 **Origin** 資料行辨識為包含分割每一資料列的值的資料行。進一步檢視上述查詢內容，可以發現我們在 **Origin** 的原始位置註解掉它，然後將在最後一列再放上 **Origin**。**Hive** 遵照一個慣例，即 **PARTITION** 子句中依序引用的資料行，在查詢最後的資料行以相同順序列出。在這個例子中，我們只引用一行，**Origin**。如果在 **PARTITION** 子句中有三個（好比 **Origin**、**Year** 和 **Quarter**）列，那麼 **FROM** 子句之前的最後三列則必須以 **Origin**、**Year**、**Quarter** 的順序寫出。有了這則語法，我們可以動態地將資料插入到正確的分割區中——這正是 **Hive** 被稱為**動態分割**的原因。

如果查看磁碟上的 **departureflightdata**（根據你的叢集，可能存放在 **Azure** 儲存體或 **Azure Data Lake Store** 中），你可以清楚看到分割區結構，如圖 7-5 所示。

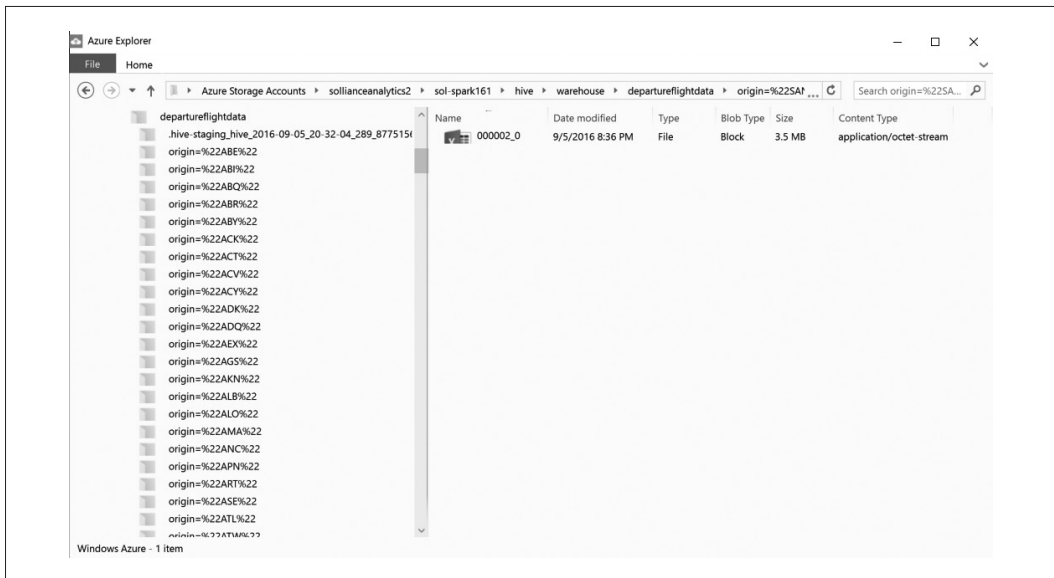


圖 7-5 以出發地機場代碼進行分割後的資料夾結構。

現在，我們的資料集以出發地機場被適當分割，讓我們以單一機場（**SAN**）的視角，找出超過 15 分鐘的延誤數量。針對你的 **Hive** 實例運行以下查詢：

```
SELECT Month, Count(*)
FROM departureflightdata
WHERE origin='SAN' AND DepDelay > 15
GROUP BY Month;
```

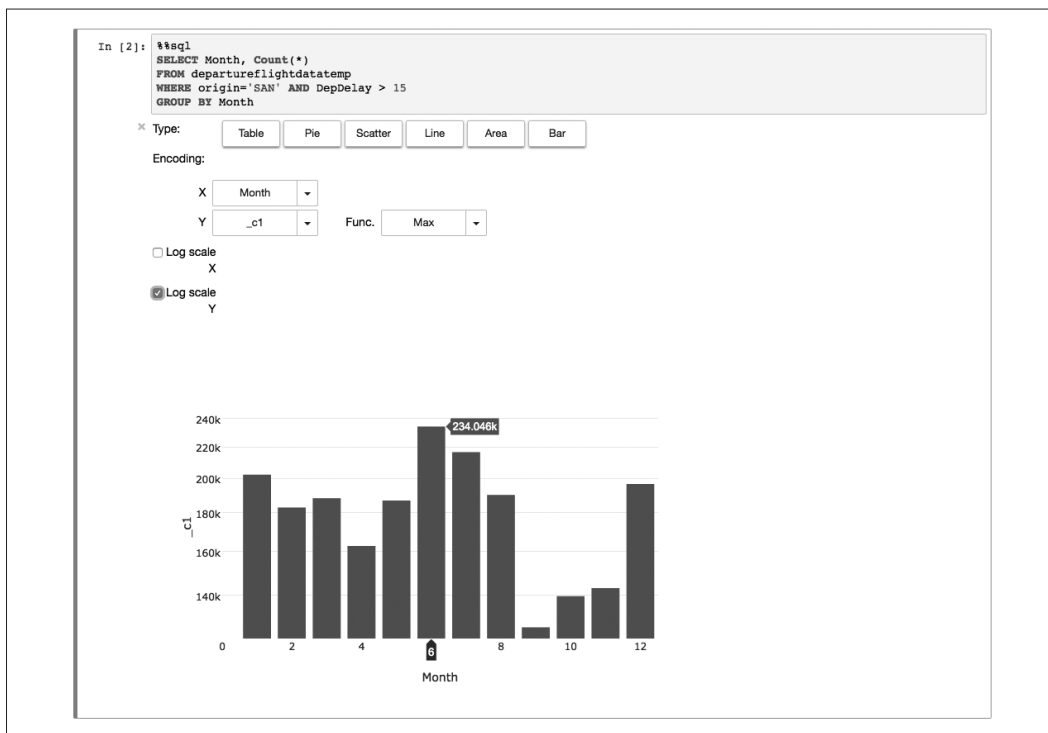


圖 7-9 針對分割區資料進行 SQL 查詢，在 Jupyter notebook 中以柱狀圖顯示查詢結果。

以 USQL 進行互動式查詢

Azure Data Lake Analytics 以批次分析為主，但它所支援的一些功能，有助於降低查詢處理時間，接近互動式查詢層級。

與 SQL Database 雷同，Azure Data Lake Analytics 支援索引、分割資料和分散資料，不過尚有一些限制。

目前而言，Azure Data Lake Analytics 中的資料表必須具備叢集索引（不支援資料行存放區索引和堆積資料表）。資料表可以被分割到數個分割區或是直接分散到分散區中。此外，分割區還可以繼續拆分到分散區中。除了定義叢集索引之外，每個資料表還必須定義資料分散模式，以確定如何將資料列分配到（組成資料表或分割區的）分散區中。