



將 SSIS ETL 任務遷移至 AWS

AWS 方案指引



AWS 方案指引: 將 SSIS ETL 任務遷移至 AWS

Copyright © 2026 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Amazon 的商標和商業外觀不得用於任何非 Amazon 的產品或服務，也不能以任何可能造成客戶混淆、任何貶低或使 Amazon 名譽受損的方式使用 Amazon 的商標和商業外觀。所有其他非 Amazon 擁有的商標均為其各自擁有者的財產，這些擁有者可能附屬於 Amazon，或與 Amazon 有合作關係，亦或受到 Amazon 贊助。

Table of Contents

簡介	1
目標業務成果	1
遷移階段	2
探索階段	2
分析階段	3
判斷遷移方法	4
實作階段	6
AWS SCT	6
AWS Glue	6
Amazon EMR	7
測試	8
最佳實務	10
常見問答集	12
遷移期間是否應該增強 SSIS 任務？	12
如何監控 上的任務 AWS？	12
何時可以停用內部部署 SSIS 任務？	12
後續步驟	13
相關資源	14
文件歷史紀錄	15
.....	xvi

將 SSIS ETL 任務遷移至 AWS

Durga Prasad 和 Harpreet Singh , Amazon Web Services (AWS)

2021 年 12 月 ([文件歷史記錄](#))

Amazon Web Services (AWS) 提供開發和執行擷取、轉換和載入 (ETL) 或擷取、載入和轉換 (ELT) 任務和大數據工作負載的服務。Microsoft SQL Server Integration Services (SSIS) 是一種內部部署 ETL 工具，具有圖形界面，可用於建置 ETL 任務。根據目標狀態架構，您可以使用 AWS Glue 圖形界面或 Apache Spark 架構搭配 Python 程式庫，在 AWS 雲端中建置 ETL 任務。

本指南說明將 SSIS ETL/ELT 任務從現場部署遷移至 的步驟 AWS。它討論遷移階段、最佳實務和建議，以減少遷移工作並改善體驗。此資訊適用於任何目標 AWS 架構。

本指南適用於以下項目的程式或專案經理、產品擁有者、解決方案架構師和開發人員：

- AWS 基於各種原因從 SSIS 遷移到 ，例如現代化或節省成本。
- 主要使用 ETL 和大數據工作負載 AWS 的服務。
- 尋找雲端型解決方案，以利用無伺服器模型和彈性定價。

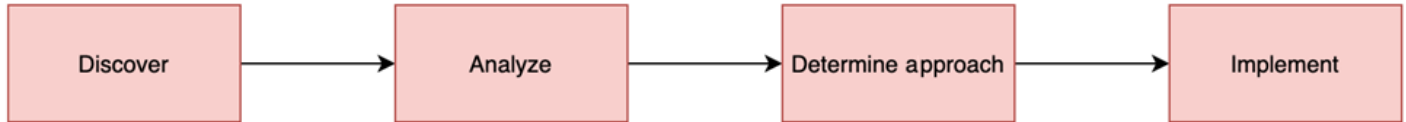
目標業務成果

將 SSIS 任務遷移至 AWS 雲端可協助您達成下列主要成果：

- 以符合成本效益的方式現代化現有的內部部署 ETL 程序
- 更快地回應組織的資訊需求
- 合併現有程序並淘汰未使用的程序，以減少維護和營運開銷
- 建立基礎以滿足組織未來的資料需求

遷移階段

將 SSIS ETL 程序遷移至 AWS 雲端包含四個主要階段：探索、分析、判斷方法和實作。



以下各節會討論這些階段：

- [探索階段](#)
- [分析階段](#)
- [判斷遷移方法](#)
- [實作階段](#)

探索階段

在探索階段中，您可以建立您要遷移至的 SSIS 套件清單 AWS。不同的開發團隊遵循不同的樣式、標準和模式來開發 ETL 任務。我們建議您檢閱組織的現有文件，以了解這些模式。不過，文件通常不完整。您可以從 ETL 指令碼自動擷取重要資訊。這可節省手動工作量和時間、減少人為錯誤，並標準化遷移方法。以下是您要擷取的一些重要詳細資訊：

- 控制流程任務總數
- 控制流程任務的詳細資訊
- 資料流程任務的總數
- 使用的資料流程轉換
- 事件處理常式
- 連線管理員

使用此資訊來了解組織中使用的 ETL 模式、評估其複雜性，以及識別要遷移此資訊的適當 AWS 服務。

從 SSIS 遷移這些 ETL 詳細資訊會形成大量的遷移工作。不過，其他屬性可以提供設計和架構決策的洞見。其中一些 SSIS 屬性為：

- [檢查點](#)，用於 SSIS，從故障點重新啟動任務
- [傳播變數](#)，這有助於 SSIS 套件在特定使用案例中成功，即使發生錯誤
- [交易隔離層級](#)，可控制從資料庫讀取的資料品質
- [記錄](#)，以了解目前設計所擷取的日誌類型及其儲存位置

探索階段的結果可以是庫存，如下表所示。

inventory

count	Package	Flow	Task
1	SSIS_Package_1	control_flow	Microsoft.ExecuteSQLTask
1	SSIS_Package_1	data_flow	Microsoft.BDD
1	SSIS_Package_1	data_flow	Microsoft.ConditionalSplit
2	SSIS_Package_1	data_flow	Microsoft.OLEDBDestination
1	SSIS_Package_1	data_flow	Microsoft.OLEDBSource
1	SSIS_Package_2	control_flow	Microsoft.DbMaintenanceFileCleanupTask
1	SSIS_Package_2	control_flow	Microsoft.ExecuteSQLTask
1	SSIS_Package_2	control_flow	Microsoft.ScriptTask
1	SSIS_Package_2	control_flow	STOCK:FOREACHLOOP
1	SSIS_Package_2	control_flow	STOCK:FORLOOP

此清查可能包含下列資訊：

- 套件：要遷移的 SSIS 套件名稱
- 流程：[控制流程](#)或[資料流程](#)
- 任務：控制流程任務或資料流程元件的名稱
- 計數：在 SSIS 套件中使用任務的次數

分析階段

分析探索階段的資訊可協助您更了解模式並設計目標架構。請依照這些指標來改善分析結果：

- **套件層級**的記錄選項會記錄事件，並擷取每個元件的執行時間資訊。使用自訂記錄，讓您可以設定日誌檔案以包含其他資訊，例如執行 IDs 和其他執行時間值。
- 將錯誤記錄重新導向至不同的儲存位置，以進行分析、更正和重新處理。
- 了解在您內部部署 SSIS 環境中實作的資料封存和清除策略。
- 使用 SSIS 隨附的任務（例如傳送**郵件任務**）或自訂指令碼（例如指令碼任務）來傳送提醒和通知。<https://docs.microsoft.com/en-us/sql/integration-services/control-flow/script-task>
- 了解範圍內**轉換**的行為，以避免資料損毀。例如，**查詢轉換**是兩個資料集之間的等聯結，其比較會區分大小寫。

您可以根據持續時間（分鐘或小時）或關卡（簡單、中等或複雜），將庫存中的每個項目對應到預估的複雜性。此擴展的庫存如下表所示，是分析階段的結果。

inventory

count	Package	Flow	Task	Effort	Complexity
1	SSIS_Package_1	control_flow	Microsoft.ExecuteSQLTask	30	S
1	SSIS_Package_1	data_flow	Microsoft.BDD	0	N/A
1	SSIS_Package_1	data_flow	Microsoft.ConditionalSplit	30	S
2	SSIS_Package_1	data_flow	Microsoft.OLEDBDestination	60	M
1	SSIS_Package_1	data_flow	Microsoft.OLEDBSource	30	S
1	SSIS_Package_2	control_flow	Microsoft.DbMaintenanceFileCleanupTask	60	M
1	SSIS_Package_2	control_flow	Microsoft.ExecuteSQLTask	30	S
1	SSIS_Package_2	control_flow	Microsoft.ScriptTask	120	C
1	SSIS_Package_2	control_flow	STOCK:FOREACHLOOP	30	S
1	SSIS_Package_2	control_flow	STOCK:FORLOOP	30	S

判斷遷移方法

若要決定遷移方法，您可以使用您在上一個階段對現有模式執行的分析。您組織的未來資料和分析需求也同樣重要。傳統的內部部署 ETL 工具可處理關聯式資料模型和結構化資料。如果您有半結構化和非結構化資料要處理，您可以使用 AWS AWS Glue 或 Amazon EMR 等服務進行遷移。可能影響遷移方法的其他因素包括：

- 您想要使用圖形界面（例如 [AWS Glue Studio](#)）或自訂架構（例如 Spark/Python 程式庫）
- 您是否能夠安全存取內部部署來源和 AWS 目標

- 團隊所需的技能和訓練
- 稽核和合規要求

您可以從三種遷移方法中進行選擇：大型 Bang、分段和提升和轉移。下表比較這三種方法。

方法	Description	使用案例	優點和缺點
大碰撞	在特定期間內遷移所有 SSIS 套件。	• 複雜性、範圍和目標架構很清楚。 • 團隊具備必要的技能，或學習曲線較淺。	• 高風險。 • 比分階段方法花費更少的時間。 • 您可以使用 AWS Glue、Amazon EMR 或自訂架構。
分段	為每個不同的模式和複雜性識別一個 SSIS 套件。將套件遷移至 AWS、測試和比較結果與現有架構。	• 時間不是限制條件。 • 您希望針對不同的 ETL 模式進行不同的設計。	• 比大型巨型方法風險更低，但需要更多時間和精力。 • 您可以使用 AWS Glue、Amazon EMR 或自訂架構。
提升和轉移	像往常一樣遷移目前的架構 AWS。	• 不再支援您的內部部署硬體。 • 您沒有資源可立即規劃遷移。	• 最少的遷移工作量和所需的時間。 • 現有解決方案的問題會保持開啟 AWS。 • SSIS 套件會照原樣執行。不需要其他 ETL 工具或架構。

比較來源和目標系統上的資料對於成功遷移至關重要。由於現有生產系統從來源系統定期更新，因此此比較可能會變得令人困惑。因此，當您決定遷移方法時，我們建議您也決定資料驗證策略。

- 在特定日期和時間從來源系統的生產環境備份所有適用的資料庫和檔案。
- 在所有任務成功載入來自備份來源資料的資料後，從目標系統上的生產環境備份所有資料庫。

- 在測試環境中還原來源資料，並執行新的任務。
- 同意來源和目標（舊和新）資料庫之間的有效差異百分比。例如，您可以決定可接受低於 1% 的差異。
- 列出要涵蓋的所有驗證規則。
- 盡可能自動化比較，並涵蓋所有規則。

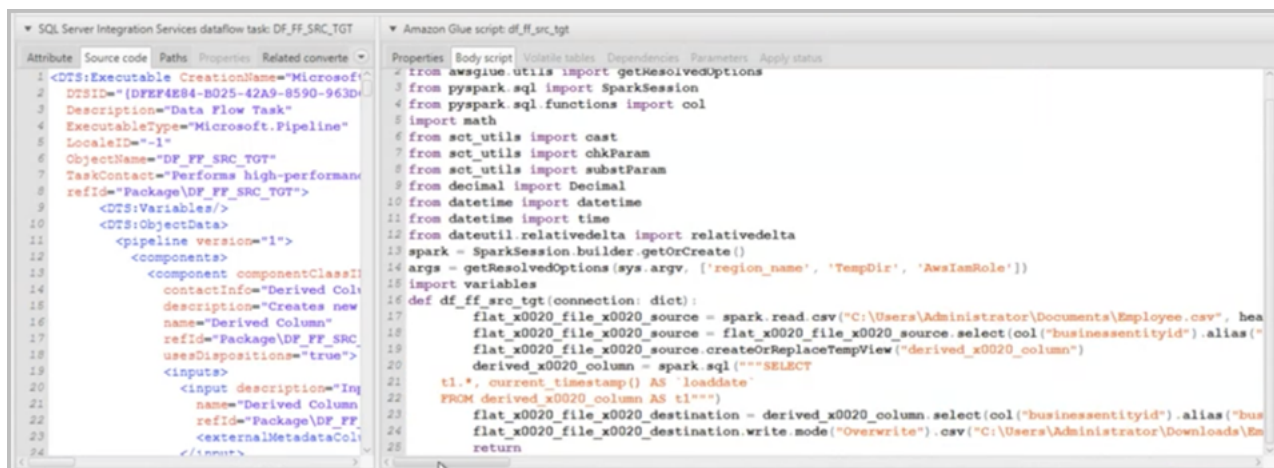
實作階段

遵循大型 Bang 或分階段方法的遷移需要新的開發和測試。AWS Schema Conversion Tool (AWS SCT) 可以自動從 SSIS 套件產生 AWS Glue 任務。這可大幅減少遷移時間和工作量。或者，您可以使用 AWS Glue Studio 進行以圖形介面為基礎的開發，或建置可在 AWS Glue 或 Amazon EMR 上執行的 Spark 程式庫。

下列各節提供使用 AWS SCT AWS Glue 和 Amazon EMR 的實用指標。

AWS SCT

下列畫面圖顯示由轉換 AWS Glue 的任務指令碼 AWS SCT。

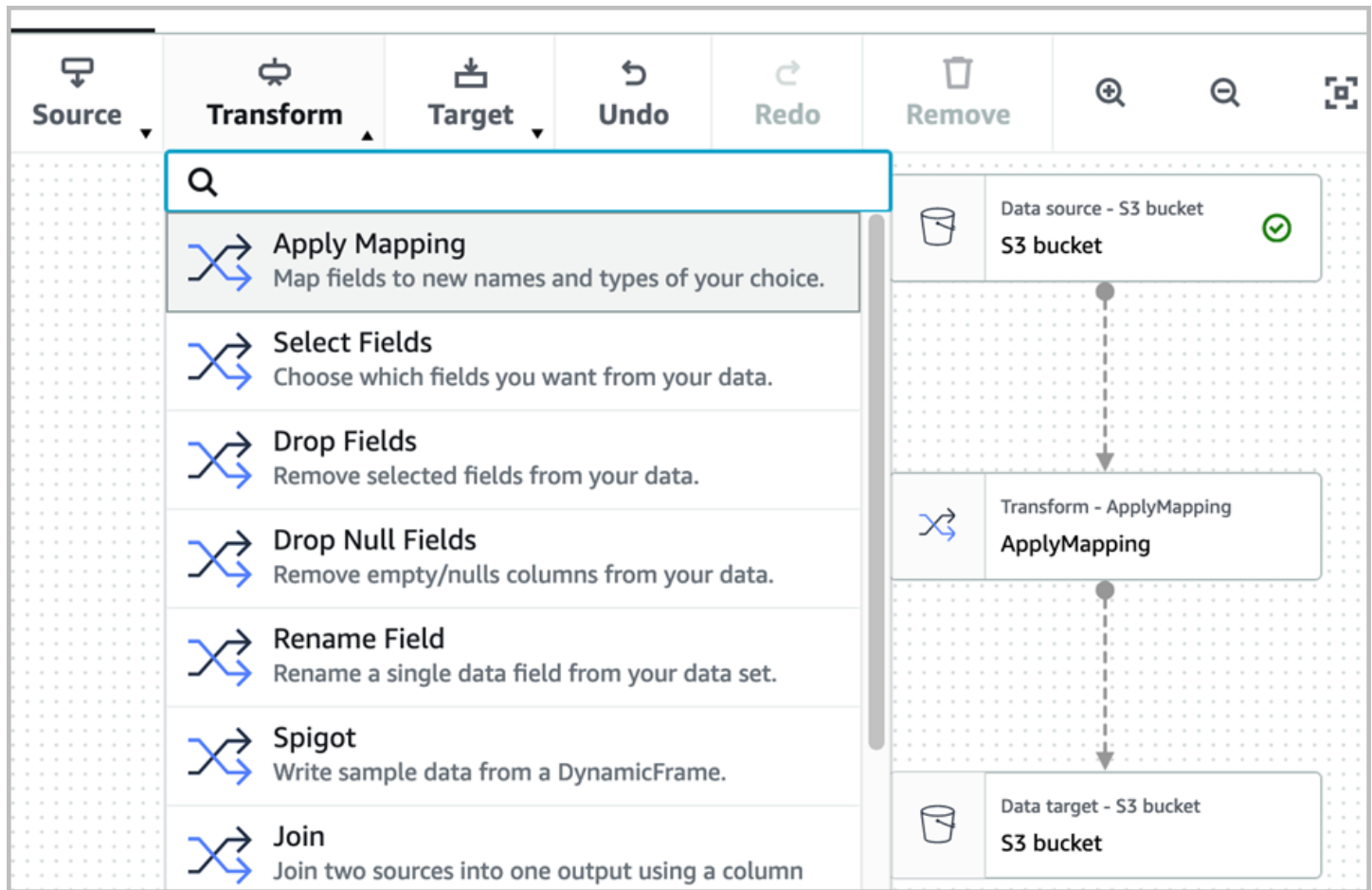


AWS SCT 可以將 SSIS 套件大量轉換為 AWS Glue 任務。您可以編輯指令碼來更新現有的邏輯，或根據您的新設計新增新的邏輯。建議您遵循 AWS SCT 轉換指令碼中的命名慣例來自訂指令碼。

如需詳細資訊，請參閱 AWS SCT 文件中的[將 SSIS 轉換為 AWS Glue 使用 AWS SCT](#)。

AWS Glue

AWS Glue Studio 提供類似於 SSIS 的圖形界面和開發體驗，如下畫面所示。



如果您不想使用圖形界面，您也可以使用 AWS Glue 主控台中所需的 Python 程式庫來執行自訂指令碼。如需詳細資訊，請參閱 AWS Glue 文件中的[提供您自己的自訂指令碼](#)。

AWS Glue 提供一組用於處理資料的內建轉換。這些類似於 SSIS 資料流程轉換。當您使用 遷移 SSIS ETL 任務時，請遵循下列最佳實務 AWS Glue：

- 準備從 AWS Glue 轉換到同等 SSIS 轉換的映射。
- 如果您的轉換無法映射到 AWS Glue 轉換，請使用 Python 或 Scala 自訂指令碼來建置轉換。
- 對於自訂記錄（例如資料列讀取、資料列寫入或記錄錯誤），除了 [Amazon CloudWatch](#) 之外，還請使用自訂指令碼。
- 新增[開發端點](#)以在本機開發和偵錯自訂指令碼。

Amazon EMR

您可以執行自訂指令碼（以 Python 或 Scala 撰寫）或在 EMR 叢集中編譯的 Python 程式庫，如下所示 AWS Glue。請遵循以下最佳實務：

- 從記憶體最佳化執行個體類型開始，同時使用 Spark 架構建立 EMR 叢集。(SSIS 使用記憶體緩衝區。)
- 建置等同於每個 SSIS 任務或轉換的一般 Python 方法。例如，在下圖中，採用兩個資料影格做為輸入的方法會產生第三個資料影格，其具有來自兩個資料影格的相符記錄做為輸出。這可做為[合併聯結轉換](#)。

```
AWS: Add Debug Configuration | AWS: Edit Debug Configuration
def merge_join_spark(spark: SparkSession, df1: DataFrame, df2: DataFrame, join_type: str, join_column: str,
                    merge_fetch_columns: str):
    """Merge/Join directly
    spark: Current Spark session
    df1: First dataframe for merge join
    df2: Second dataframe for merge join
    join_type: Left/Inner/Outer join
    join_column : Column to be merge joined on
    merge_fetch_columns: Columns required in the output dataframe
    """
    try:
        spark.catalog.dropTempView("df1")
        spark.catalog.dropTempView("df2")
        df1.createOrReplaceTempView("df1")
        df2.createOrReplaceTempView("df2")
        merge_sql = f"select src.*, {merge_fetch_columns} from df1 {join_type} join df2 on df1.{join_column} = df2.{join_column}".format(
            merge_fetch_columns=merge_fetch_columns, join_type=join_type, join_column=join_column)
        logger.debug("Merge query is : " + merge_sql)
        output_df = spark.sql(merge_sql)
    except Exception as ex:
        logger.error("Merge Execution Failed for " + str(ex))
        raise Exception("Merge Execution Failed " + str(ex))
    finally:
        spark.catalog.dropTempView("df1")
        spark.catalog.dropTempView("df2")
    return output_df
```

測試

需要測試架構來驗證資料的完整性和正確性。此架構應涵蓋所有現有的案例，以及您遷移任務時所做的任何改善 AWS。

- 完整性驗證：
 - 所有任務都會遷移至其目標狀態。
 - 所有功能都會在每個任務中遷移。
 - 所有類型的日誌都可使用，包括任務執行詳細資訊、錯誤訊息、錯誤記錄和資料列計數。
- 正確性驗證：
 - 在現有和新的環境中，資料的品質是一致的。
 - 所有資料表的所有資料欄都相符，或資料表已改善 AWS。
 - 所有稽核和記錄資訊都相符。

您也應該驗證遷移任務的效能是否符合現有任務的效能。

最佳實務

將 SSIS 任務遷移至 時，請遵循這些一般最佳實務 AWS：

- 如果您要將資料庫遷移至 AWS 受管服務，例如 Amazon Relational Database Service (Amazon RDS)，請忽略不適用或管理的[維護任務](#)（例如備份、還原、更新統計資料或檢查資料庫完整性）AWS。
- 建置可重複使用的[指令碼](#)和方法，以標準化開發並減少工作量。
- 一律在符合生產資源的基礎設施和資料磁碟區上測試遷移的任務。
- SSIS 套件採用 XML 格式。建置 XML 剖析器公用程式，盡可能自動化遷移活動。下列範例顯示 XML 格式的 SSIS 套件。

```
DTS:LastModifiedProductVersion="11.0.2100.60"  
DTS:LocaleID="1033"  
DTS:ObjectName="Package"  
DTS:PackageType="5"  
DTS:VersionBuild="1"  
DTS:VersionGUID="{EC16490E-6783-4BE6-92CA-FDD5BC644F88}">  
<DTS:Property  
  DTS:Name="PackageFormatVersion">6</DTS:Property>  
<DTS:ConnectionManagers>  
  <DTS:ConnectionManager
```

下圖顯示範例 XML 剖析器。

```

def getlistofallexecutables(directory):
    """
    Iterate the root directory with all SSIS packages (.dtsx)
    Get the control flow tasks and data flow task components
    """
    try:
        lst=[]
        for fileName in os.listdir(directory):
            if fileName.endswith('.dtsx'):
                cfgfilename=os.path.basename(fileName).split('.')[0]
                tree = etree.parse(directory+fileName)
                root = tree.getroot()
                prefix = '{www.microsoft.com/|
                exetag = prefix + 'SqlServer/Dts}Executable'
                dataflow='Microsoft.Pipeline'
                childtag=prefix+ 'SqlServer/Dts}ExecutableType'
                objdata=prefix+ 'SqlServer/Dts}ObjectData'
                pipeline='pipeline'
                components= 'components'
                componentclassid="componentClassID"
                for cnt, ele in enumerate(root.xpath(".*/*")):
                    if ele.tag==exetag:
                        attr=ele.attrib[childtag]#controlflow
                        flow=cfgfilename+',control_flow,'
                        if attr!=dataflow:
                            lst.append(flow+attr)
                        if attr==dataflow:
                            for child0 in ele:
                                if child0.tag==objdata:
                                    for child1 in child0:
                                        if child1.tag==pipeline:
                                            for child2 in child1:
                                                if child2.tag==components:
                                                    for child3 in child2:
                                                        df_component=child3.attrib[componentclassid]#
                                                        flow=cfgfilename+',data_flow,'
                                                        lst.append(flow+df_component)
                return lst
    except Exception as e:

```

- 使用檢查總和和雜湊值來測試正確性和完整性。
- 如果您要建置自訂程式庫，請實作執行緒以平行執行任務。這可改善任務效能。
- 只有在所有任務都處於穩定狀態 AWS 一段時間後，才能停用現有環境。
- 使用 Amazon CloudWatch 進行記錄，以減少記錄自訂工作。
- 使用 Amazon Simple Notification Service (Amazon SNS) 取代傳送通知的自訂任務。使用 Amazon Simple Email Service (Amazon SES) 取代自訂電子郵件任務。

常見問答集

本節提供有關將 SSIS ETL 任務遷移至 的常見問題解答 AWS。

遷移期間是否應該增強 SSIS 任務？

是。同時在現有任務和新任務中實作任何重要的增強功能或錯誤修正。低優先順序變更可在遷移後實作。

如何監控 上的任務 AWS？

您可以使用 [Amazon CloudWatch](#) 來監控任務和傳送規則型提醒。

何時可以停用內部部署 SSIS 任務？

我們建議您將舊任務和新任務平行保留一段時間，直到兩個環境中的資料效能、準確性和完整性相同。您也可以複寫報告系統，以連接和比較兩個環境的報告。當報告相同時，您可以停用 SSIS 任務。

後續步驟

下一步，我們建議您建置概念驗證，以判斷目標架構的部署方法。針對您的概念驗證：

- 使用 AWS Glue Studio 圖形界面建立 AWS Glue 任務。
- 使用 AWS Schema Conversion Tool (AWS SCT) 從 SSIS 套件產生 AWS Glue 指令碼。
- 使用 Spark 程式庫建置自訂遷移架構。
- 建置測試架構。
- 識別用於持續整合和持續部署 (CI/CD) 的工具和程序。

相關資源

- [AWS Glue 使用 將 SSIS 轉換為 AWS SCT](#)
- [Amazon RDS 備份和還原](#)
- [AWS Glue 文件](#)
- [AWS Glue FAQs](#)
- [AWS Glue 任務的持續記錄](#)
- [Amazon EMR 文件](#)
- [Amazon EMR FAQs](#)
- [使用 CloudWatch 監控指標](#)
- [使用 PyDeequ 加速大規模資料遷移驗證](#)
- [AWS 資料和分析能力合作夥伴](#)

文件歷史紀錄

下表描述了本指南的重大變更。如果您想收到有關未來更新的通知，可以訂閱 [RSS 摘要](#)。

變更	描述	日期
初次出版	—	2021 年 12 月 6 日

本文為英文版的機器翻譯版本，如內容有任何歧義或不一致之處，概以英文版為準。