



使用者指南

AWS Ground Station



AWS Ground Station: 使用者指南

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

什麼是 AWS Ground Station ?	1
常用案例	1
後續步驟	2
AWS Ground Station 運作方式	3
衛星加入	3
任務描述檔合成	3
聯絡排程	5
聯絡執行	6
數位分身	8
了解 AWS Ground Station 核心元件	8
任務設定檔	10
組態	12
Dataflow 端點群組	20
AWS Ground Station 代理程式	27
開始使用	29
註冊 AWS 帳戶	29
建立具有管理存取權的使用者	29
將 AWS Ground Station 許可新增至 AWS 您的帳戶	30
加入衛星	32
客戶加入程序概觀	32
(選用) 命名衛星	33
公有廣播衛星	35
規劃您的資料流程通訊路徑	36
非同步資料交付	36
同步資料交付	37
規劃您的遙測	38
建立組態	39
資料交付組態	39
遙測組態 (選用)	39
衛星組態	39
建立任務描述檔	39
了解後續步驟	40
AWS Ground Station 位置	42
尋找地面站點位置的 AWS 區域	42

AWS Ground Station 支援的 AWS 區域	43
數位分身可用性	43
專用天線	44
在地面工作站檢視天線	44
範例：列出地面工作站的天線	44
檢視地面站點保留	45
列出保留	45
保留類型	45
程式碼範例	46
AWS Ground Station 網站遮罩	48
客戶特定的遮罩	48
網站遮罩對可用聯絡時間的影響	48
AWS Ground Station 站點功能	49
了解 AWS Ground Station 如何使用 ephemerides	52
預設暫時性資料	52
提供自訂暫時性資料	53
概觀	53
範例：搭配使用客戶提供的 ephemeride AWS Ground Station	53
提供 TLE ephemeris 資料	53
提供 OEM ephemeris 資料	59
提供方位提升暫時性資料	67
使用自訂 ephemeris 預留聯絡人	77
概觀	77
聯絡預留工作流程	78
工作流程 1：列出可用的聯絡人，然後保留	78
工作流程 2：直接聯絡預留	82
監控聯絡狀態變更	86
最佳實務和考量	88
了解使用哪些 ephemeris	89
TLE 和 OEM 暫時化	89
方位海拔半合子	89
新暫時性差異對先前排程聯絡人的影響	90
取得衛星的目前暫時性變化	90
使用預設 ephemeris 的衛星 GetSatellite 傳回範例	91
使用自訂 ephemeris 的衛星 GetSatellite 範例	91
列出方位角升高環霧化物	92

還原至預設 ephemeris 資料	93
還原 TLE 和 OEM 暫時化	93
管理方位角升高環羈化物	93
使用資料流程	95
AWS Ground Station 資料平面界面	95
使用跨區域資料交付	96
設定 Amazon S3	96
設定 Amazon VPC	96
使用 AWS Ground Station 代理程式的 VPC 組態	97
具有資料流程端點的 VPC 組態	99
設定 Amazon EC2	101
提供的常見軟體	102
AWS Ground Station Amazon Machine Image AMIs)	102
使用遙測	103
遙測的運作方式	103
可用的遙測類型	103
區域可用性	104
設定遙測	104
步驟 1：建立先決條件 AWS 資源	104
步驟 2：建立 TelemetrySinkConfig	106
步驟 3：將遙測新增至您的任務設定檔	106
步驟 4：排程聯絡	107
後續步驟	107
了解遙測資料	107
資料格式概觀	108
指向遙測	108
追蹤遙測	110
從 Kinesis Data Streams 串流讀取資料	111
結構描述版本控制和演變	112
使用聯絡人	114
了解聯絡生命週期	114
AWS Ground Station 聯絡狀態	116
聯絡人資料保留	117
了解聯絡帳單	117
頻寬定義	118
排程模式	118

CancelContact	118
案例 1：單一聯絡人	119
案例 2：單一已停止的聯絡	119
案例 3：單一重複	120
案例 4：簡短重複	121
案例 5：多個重複項目	122
案例 6：多次停止	123
案例 7：沒有重複的多天線地面站	124
案例 8：具有重複聯絡人的多天線地面站	125
更新聯絡人和聯絡人版本控制	126
聯絡版本控制的運作方式	126
更新聯絡人	126
聯絡版本狀態	128
程式碼範例	128
考量事項	132
AWS Ground Station 數位分身	133
AWS Ground Station 專用天線	134
什麼是專用天線	134
增強的保留可見性	134
相關資源	135
監控	136
使用 事件自動化	136
AWS Ground Station 事件類型	137
聯絡事件時間表	137
Ephemeris 事件	140
使用 CloudTrail 記錄 API 呼叫	141
AWS Ground Station CloudTrail 中的資訊	141
了解 AWS Ground Station 日誌檔案項目	142
使用 Amazon CloudWatch 檢視指標	143
AWS Ground Station 指標和維度	144
檢視 指標	147
安全	153
身分和存取權管理	153
目標對象	154
使用身分驗證	154
使用政策管理存取權	155

AWS Ground Station 如何使用 IAM	156
身分型政策範例	161
疑難排解	163
AWS 受管政策	165
AWSGroundStationAgentInstancePolicy	165
AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy	166
政策更新	166
使用服務連結角色	167
Ground Station 的服務連結角色許可	167
為 Ground Station 建立服務連結角色	168
編輯 Ground Station 的服務連結角色	168
刪除 Ground Station 的服務連結角色	168
Ground Station 服務連結角色支援的區域	169
疑難排解	169
的靜態資料加密 AWS Ground Station	169
建立客戶自管金鑰	171
指定的客戶受管金鑰 AWS Ground Station	171
AWS Ground Station 加密內容	172
TLE 和 OEM ephemeris 資料的靜態加密	172
疊流高度暫時性的靜態加密	181
傳輸期間的資料加密 AWS Ground Station	189
AWS Ground Station 代理程式串流	189
Dataflow 端點串流	189
範例任務設定檔組態	190
JPSS-1 - 公有廣播衛星 (PBS) - 評估	190
使用 Amazon S3 資料交付的公有廣播衛星	191
通訊路徑	191
AWS Ground Station 組態	193
AWS Ground Station 任務描述檔	194
將它放在一起	195
使用資料流程端點的公有廣播衛星 (窄頻帶)	196
通訊路徑	196
AWS Ground Station 組態	203
AWS Ground Station 任務描述檔	204
將它放在一起	204
使用資料流程端點的公有廣播衛星 (解調和解碼)	206

通訊路徑	206
AWS Ground Station 組態	213
AWS Ground Station 任務描述檔	216
將它放在一起	217
使用 AWS Ground Station 代理程式的公有廣播衛星（寬頻）	218
通訊路徑	219
AWS Ground Station 組態	230
AWS Ground Station 任務描述檔	231
將它放在一起	232
疑難排解	234
對將資料交付至 Amazon EC2 的聯絡人進行故障診斷	234
步驟 1：確認您的 EC2 執行個體正在執行	234
步驟 2：判斷使用的資料流程應用程式類型	235
步驟 3：確認資料流程應用程式正在執行	235
步驟 4：確認您的資料流程應用程式串流已設定	237
步驟 5：確定您的接收器執行個體（多個）子網路中有足夠的可用 IP 地址	238
故障診斷失敗的聯絡案例	239
Dataflow 端點 FAILED 使用案例	239
AWS Ground Station 代理程式失敗的使用案例	240
故障診斷失敗的聯絡更新	240
同步驗證錯誤	241
非同步失敗代碼	242
檢查更新的狀態	244
故障診斷 FAILED_TO_SCHEDULE 聯絡人	245
不支援在天線下行解碼組態中指定的設定	245
一般故障診斷步驟	245
針對未處於 HEALTHY 狀態的 DataflowEndpointGroups 進行故障診斷	246
故障診斷無效的短暫性事件	246
了解暫時性驗證錯誤	246
TLE ephemerides 的常見驗證錯誤	247
OEM ephemerides 的常見驗證錯誤	247
方位角升高環霧化物的常見驗證錯誤	248
疑難排解步驟	249
完成錯誤碼參考	249
對未收到資料的聯絡人進行故障診斷	253
下行組態不正確	254

衛星動作	254
AWS Ground Station 中斷	254
對遙測進行故障診斷	254
常見的設定問題	254
遙測交付問題	257
資料格式問題	259
取得說明	259
配額和限制	261
服務條款	262
文件歷史記錄	263
AWS 詞彙表	268
.....	cclxix

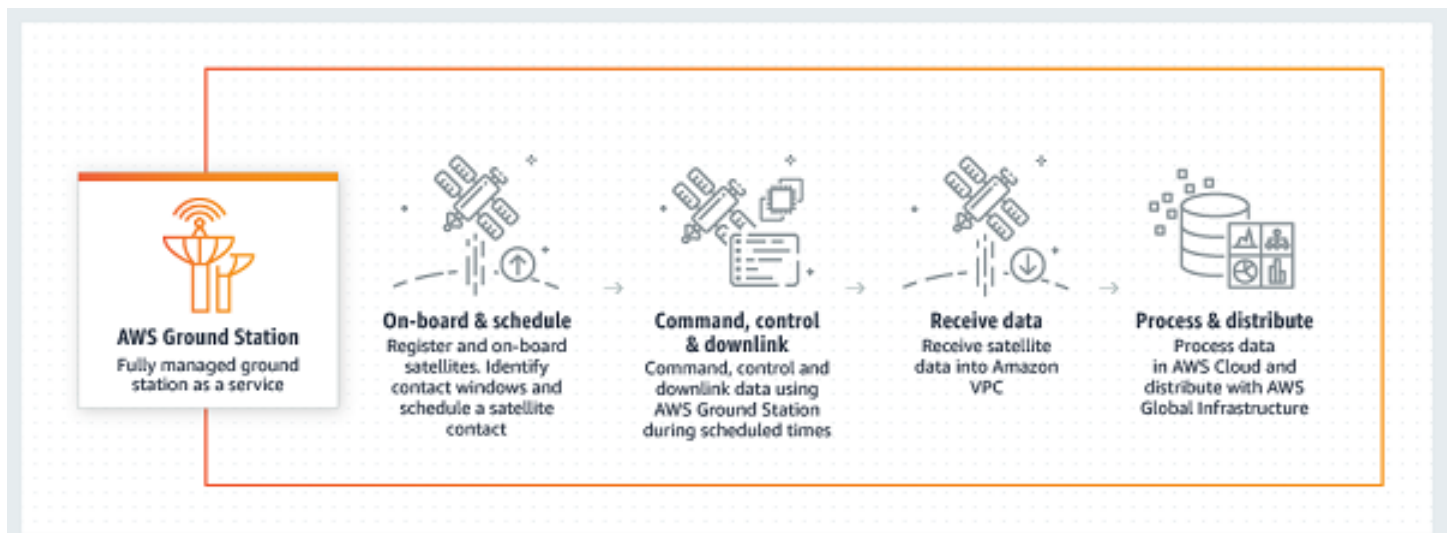
什麼是 AWS Ground Station ？

AWS Ground Station 是一種全受管服務，可在全球基礎設施中提供安全、快速且可預測的衛星通訊。使用 AWS Ground Station，您不再需要建置、管理或擴展自己的地面站基礎設施。AWS Ground Station 可讓您專注於創新和快速實驗擷取衛星資料的新應用程式，而不是花費資源來建置、操作和擴展自己的地面站。

使用 AWS 的低延遲、高頻寬全球光纖網路，您可以在天線系統接收數秒內開始處理您的衛星資料。這可讓您在幾秒鐘內將原始資料轉換為已處理的資訊或已分析的知識。

對於具有特殊需求的組織，AWS Ground Station 也提供 [AWS Ground Station 專用天線](#) - 可代表您 AWS 管理的自訂建置天線系統，可讓您專用存取根據您的規格建置的天線。

常用案例



AWS Ground Station 可讓您雙向與衛星通訊，並支援下列使用案例：

- 下行資料 – 從衛星接收資料、傳輸 X 頻帶和 S 頻帶頻率、即時交付至 Amazon EC2 執行個體 (VITA-49 格式)，或直接交付至您帳戶中的 Amazon S3 儲存貯體 ([PCAP 格式](#))。此外，對於使用支援的調變和編碼方案的衛星，您可以選擇接收解調和解碼的資料，或原始數位中繼頻率 (DigIF) 範例 (VITA-49 格式)。
- 上行資料 – 透過傳送要傳輸的 DigIF 資料 (VITA-49 格式)，將資料和命令傳送至接收 S 頻帶頻率的衛星 AWS Ground Station。
- 上行回應 – 透過在實體共置天線上接收傳輸的訊號，驗證傳送到太空船的命令，並執行其他進階任務。

- 軟體定義無線電 (SDR)/前端處理器 (FEP) – 使用可在 Amazon EC2 執行個體上執行的現有 SDR 和/或 FEP，即時處理資料以傳送/接收現有波形，並產生資料產品。
- 遙測、追蹤和命令 (TT&C) – 使用先前列出的使用案例組合來執行 TT&C，以管理您的衛星機群。
- 跨區域資料交付 – 使用來自單一 AWS 區域的 AWS Ground Station 全域天線網路同時操作多個聯絡人。
- 數位分身 – 以較低的成本測試排程、驗證組態和適當的錯誤處理，而無需使用生產天線容量。

後續步驟

建議您一開始先閱讀下列各節：

- 若要了解基本 AWS Ground Station 概念，請參閱 [AWS Ground Station 運作方式](#)。
- 若要了解如何設定要使用的帳戶和資源 AWS Ground Station，請參閱 [開始使用](#)。
- 若要以程式設計方式使用 AWS Ground Station，請參閱 [AWS Ground Station API 參考](#)。API 參考 AWS Ground Station 會詳細說明 的所有 API 操作。它還提供支援 Web 服務通訊協定的範例請求、回應和錯誤。您可以使用 [AWS CLI](#) 或 [AWS SDK](#)，以您選擇的語言撰寫與 互動的程式碼 AWS Ground Station。

AWS Ground Station 運作方式

AWS Ground Station 會操作地面型天線，以促進與衛星的通訊。天線可以執行的操作的物理特性是抽象的，稱為功能。您可以在 [AWS Ground Station 位置](#) 區段中參考天線的實體位置及其目前功能。[AWS Support Center Console](#) 如果您的使用案例需要額外的功能、其他位置產品或更精確的天線位置，請透過 [聯絡我們](#)。

若要使用其中一個 AWS Ground Station 天線，您必須在特定位置預留時間。此保留稱為聯絡人。若要成功排程聯絡人，AWS Ground Station 需要其他資料以確保其成功。

- 您的衛星必須加入一或多個位置 – 這可確保您在請求的位置擁有操作各種功能的核准。
- 您的衛星必須具有有效的暫時性 – 這可確保天線具有視線，並且可以在聯絡期間準確指向您的衛星。
- 您必須擁有有效的任務設定檔 – 這可讓您自訂此聯絡人的行為，包括接收和傳送資料到衛星的方式。您可以針對同一部車輛使用多個任務設定檔來建立不同的聯絡人，以適應您遇到的不同操作狀態或案例。

衛星加入

將衛星加入 AWS Ground Station 是一個多步驟程序，涉及資料收集、技術驗證、頻譜授權，以及整合和測試。本指南的 [衛星加入](#) 區段將引導您完成此程序。

任務描述檔合成

衛星頻率資訊、[資料平面](#) 資訊和其他詳細資訊會封裝在任務設定檔中。任務描述檔是組態元件的集合。這可讓您重複使用不同任務描述檔之間的組態元件，因為符合您的使用案例。由於任務描述檔不會直接參考個別衛星，而是僅包含其技術功能的相關資訊，因此具有相同組態的多個衛星也可以重複使用任務描述檔。

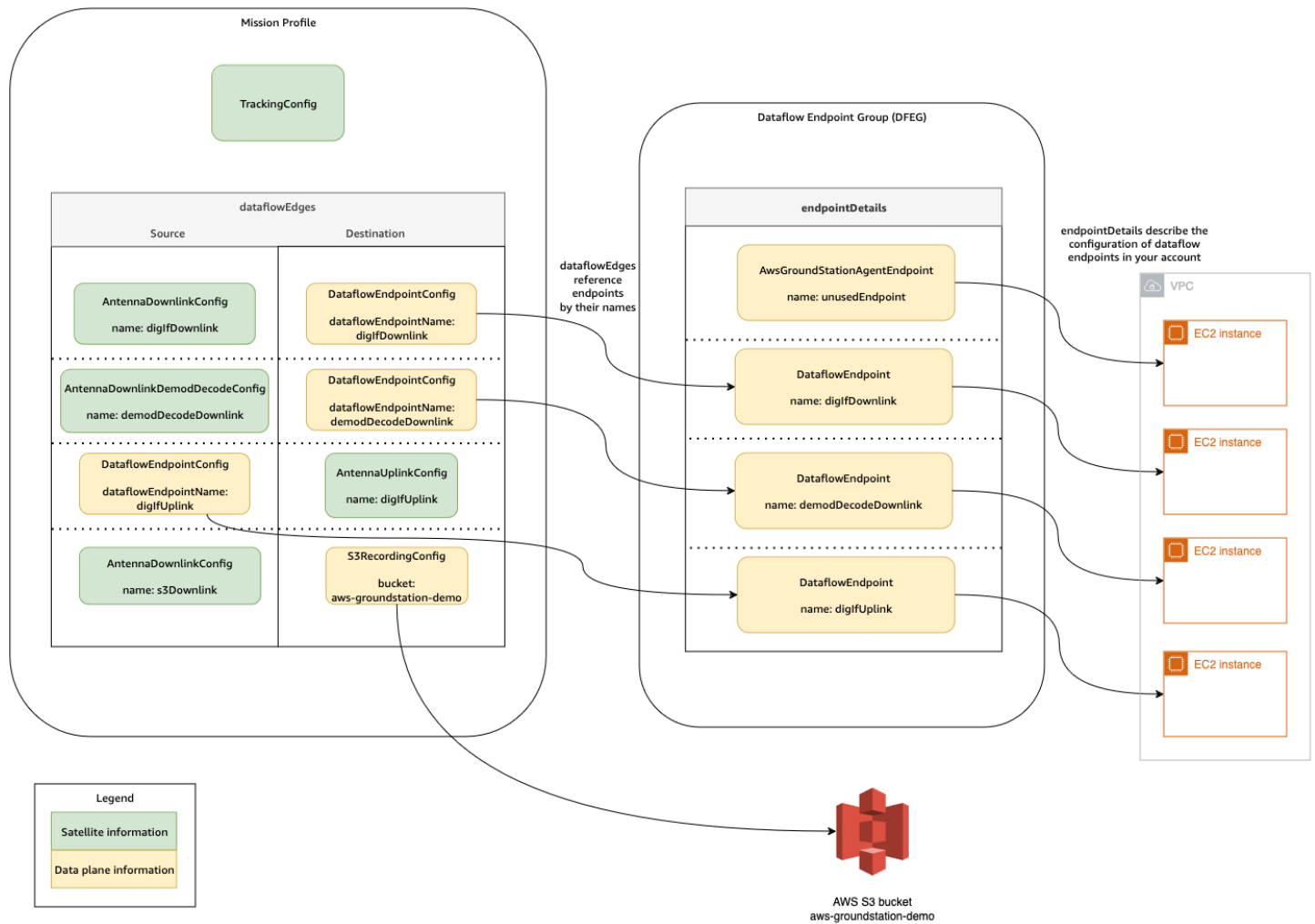
有效的任務描述檔將具有追蹤組態和一或多個資料流程。追蹤組態會指定您在聯絡期間追蹤的偏好設定。資料流程中的每個組態對都會建立來源和目的地。根據您的衛星及其操作模式，任務設定檔中的資料流程確切數量會有所不同，以代表您的上行和下行通訊路徑以及任何資料處理層面。

- 如需設定將在聯絡期間使用的 Amazon VPC、Amazon S3 和 Amazon EC2 資源的詳細資訊，請參閱 [使用資料流程](#)。
- 如需每個組態行為的詳細資訊，請參閱 [使用 AWS Ground Station 組態](#)。

- 如需所有預期參數的特定詳細資訊，請參閱 [使用 AWS Ground Station 任務設定檔](#)。
- 如需如何建立各種任務描述檔以支援使用案例的範例，請參閱 [範例任務設定檔組態](#)。

下圖顯示範例任務設定檔和其他所需的資源。請注意，此範例顯示此任務描述檔名稱 unusedEndpoint 不需要的資料流程端點，以示範彈性。此範例支援下列資料流程：

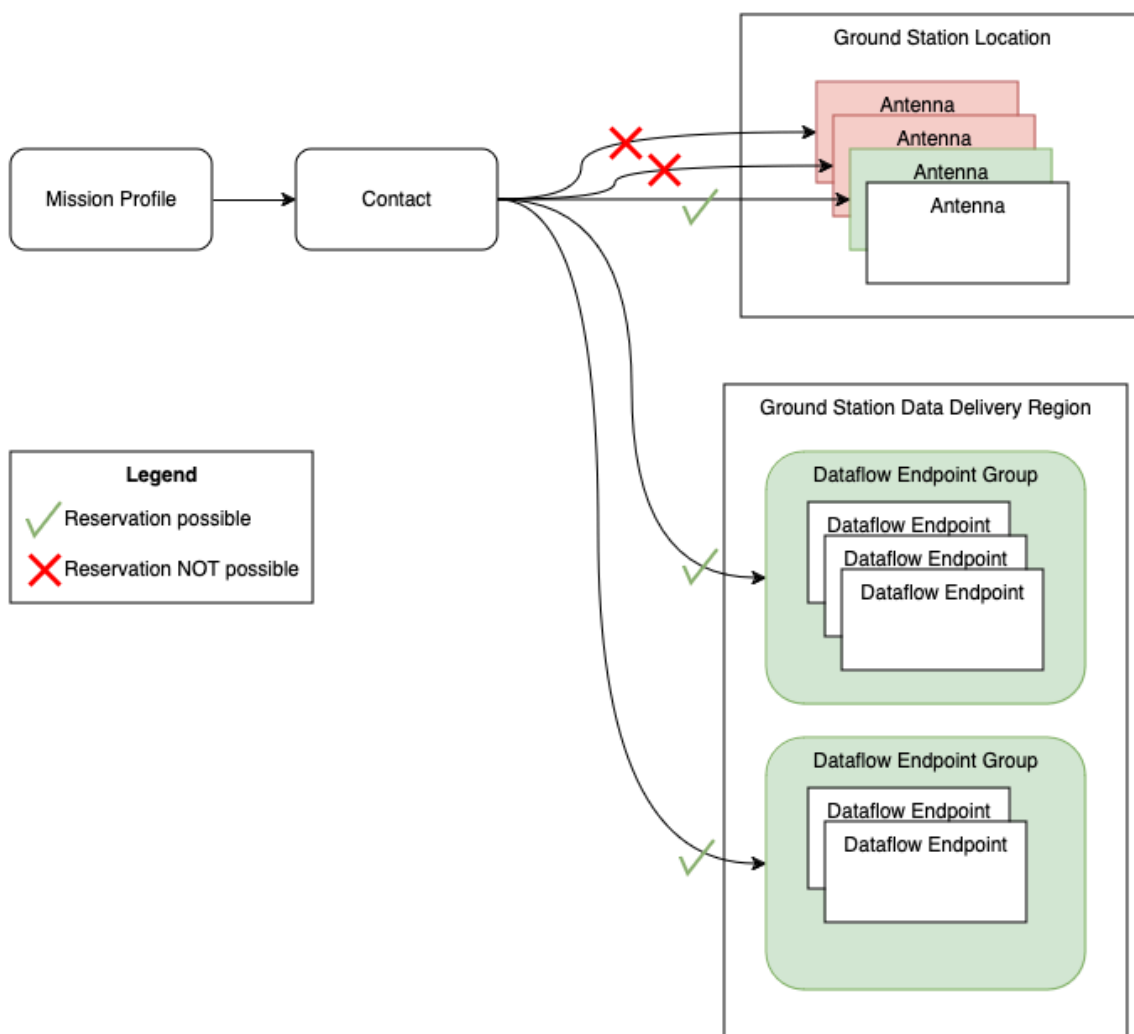
- 將數位中繼頻率資料同步下行至您管理的 Amazon EC2 執行個體。以 digIfDownlink 名稱表示。
- 數位中繼頻率資料非同步下行至 Amazon S3 儲存貯體。以儲存貯體名稱 aws-groundstation-demo 表示。
- 將解調資料和解碼資料同步下行至您管理的 Amazon EC2 執行個體。以名稱 demodDecodeDownlink 表示。
- 資料從您管理的 Amazon EC2 執行個體同步上行至 AWS Ground Station 受管天線。以 digIfUplink 名稱表示。



聯絡排程

使用有效的任務描述檔，您可以請求與加入衛星的聯絡。聯絡人保留請求是非同步的，可讓全域天線服務有時間在所有涉及的 AWS 區域中實現一致的排程。在此過程中，會評估請求的地面站位置上的各種天線，以判斷它們是否可用且能夠處理聯絡。在此過程中，也會評估您設定的資料流程端點，以判斷其可用性。進行此評估時，聯絡狀態將為 SCHEDULING。

此非同步排程程序將在請求的五分鐘內完成，但通常會在一分鐘內完成。請檢閱[AWS Ground Station 使用事件自動化](#)排程時間期間以事件為基礎的監控。



可執行且具有可用性的聯絡人會導致排程聯絡人。透過排程聯絡人，執行聯絡人所需的資源已保留給任務設定檔所定義的所需 AWS 區域。無法執行或沒有可用部分的聯絡人將導致 `FAILED_TO_SCHEDULE` 聯絡人。如需偵錯詳細資訊，請參閱[故障診斷 FAILED_TO_SCHEDULE 聯絡人](#)，請參閱。

聯絡執行

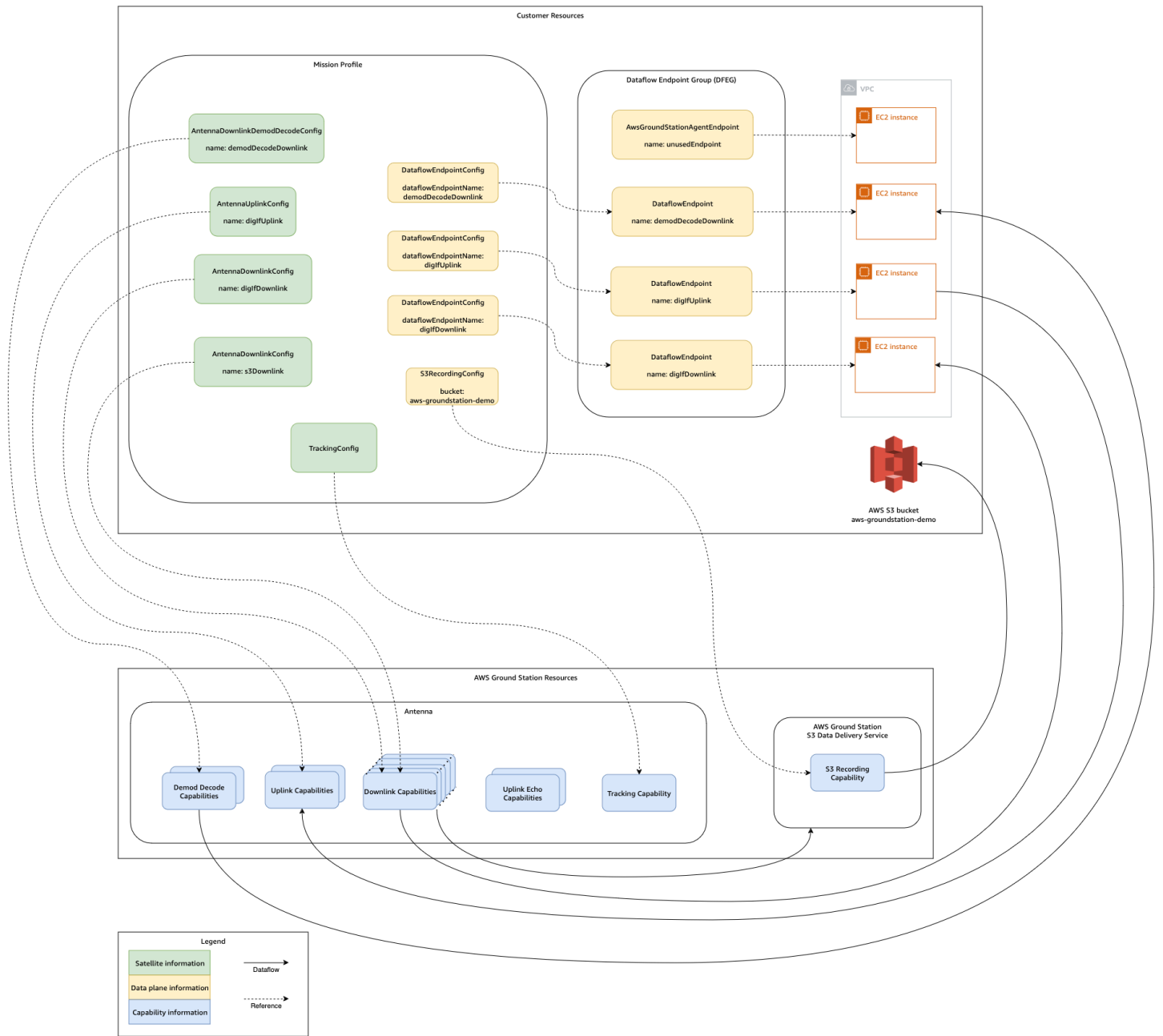
AWS Ground Station 會在您的聯絡保留期間自動協調您的 AWS 受管資源。如果適用，您必須負責協調任務描述檔定義為資料流程端點的 EC2 資源。AWS Ground Station 提供 [AWS EventBridge Events](#) 以自動化資源的協調作業，以降低成本。如需詳細資訊，請參閱 [AWS Ground Station 使用 事件自動化](#)。

在聯絡期間，系統會將有關聯絡效能的遙測傳送至 AWS CloudWatch。如需有關如何在執行期間監控聯絡人的資訊，請參閱 [了解使用 監控 AWS Ground Station](#)。

下圖透過顯示在聯絡期間協調的相同資源，繼續先前的範例。

Note

此範例中並未使用所有天線功能。例如，每個支援多個頻率和極化的天線都有十幾個天線下行功能。如需 AWS Ground Station 天線提供的每個功能類型數量及其支援的頻率和極化的詳細資訊，請參閱 [AWS Ground Station 站點功能](#)。



在聯絡結束時，AWS Ground Station 將評估您的聯絡績效，並確定最終聯絡狀態。未偵測到錯誤的聯絡將導致完成聯絡狀態。服務錯誤在聯絡期間造成資料交付問題的聯絡將導致 AWS_FAILED 狀態。用戶端或使用者錯誤在聯絡期間造成資料交付問題的聯絡將導致失敗狀態。超過聯絡時間的錯誤，在傳遞前或傳遞後期間，在判定期間不會納入考量。

如需詳細資訊，請參閱[了解聯絡生命週期](#)。

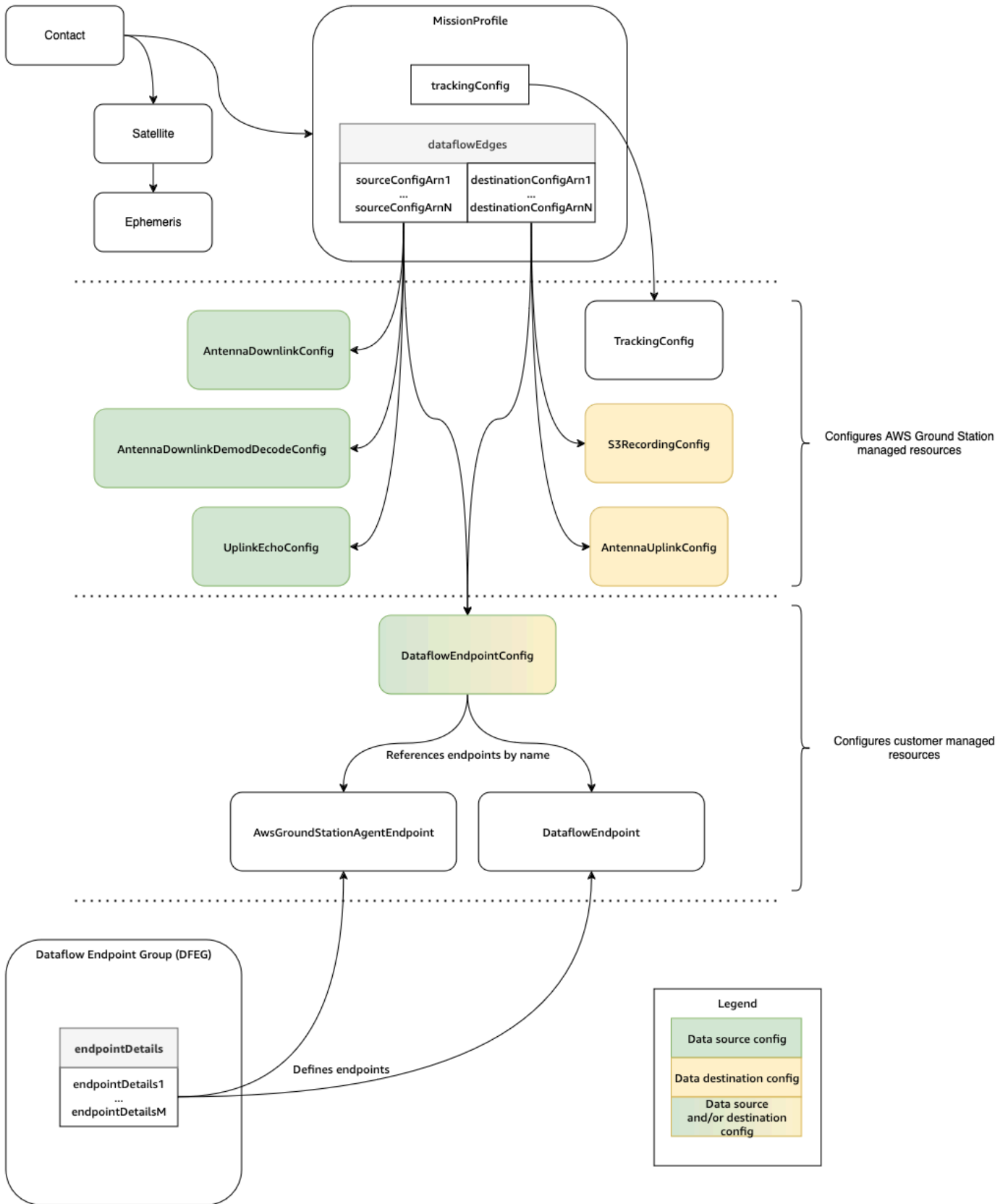
數位分身

的數位分身功能 AWS Ground Station 可讓您針對虛擬地面站位置排程聯絡人。這些虛擬地面站是生產地面站的確切複本，包括天線功能、站點遮罩和實際 GPS 座標。相較於生產地面站點，數位分身功能可讓您測試聯絡協調工作流程的成本一小部分。如需詳細資訊，請參閱「[使用 AWS Ground Station 數位分身功能](#)」。

了解 AWS Ground Station 核心元件

本節提供 AWS Ground Station 核心元件的詳細定義。

下圖顯示的核心元件 AWS Ground Station 及其彼此的關係。箭頭指出元件之間的相依性方向，其中每個元件指向其相依性。



下列主題詳細說明 AWS Ground Station 核心元件。

主題

- [使用 AWS Ground Station 任務設定檔](#)
- [使用 AWS Ground Station 組態](#)
- [使用 AWS Ground Station Dataflow 端點群組](#)
- [使用 AWS Ground Station 代理程式](#)

使用 AWS Ground Station 任務設定檔

任務設定檔包含如何執行接收的組態和參數。當您預留接收或搜尋可用的接收時，您提供打算使用的任務設定檔。任務設定檔會將所有組態集合在一起，並定義天線的設定方式，以及聯絡期間資料將前往何處。

任務設定檔可以跨共用相同無線電特性的衛星共用。您可以建立其他資料流程端點群組，以繫結要為星座執行的最大同時聯絡案例。

追蹤組態在任務描述檔中指定為唯一欄位。追蹤組態用於指定您在聯絡期間使用程式追蹤和自動追蹤的偏好設定。如需詳細資訊，請參閱[追蹤組態](#)。

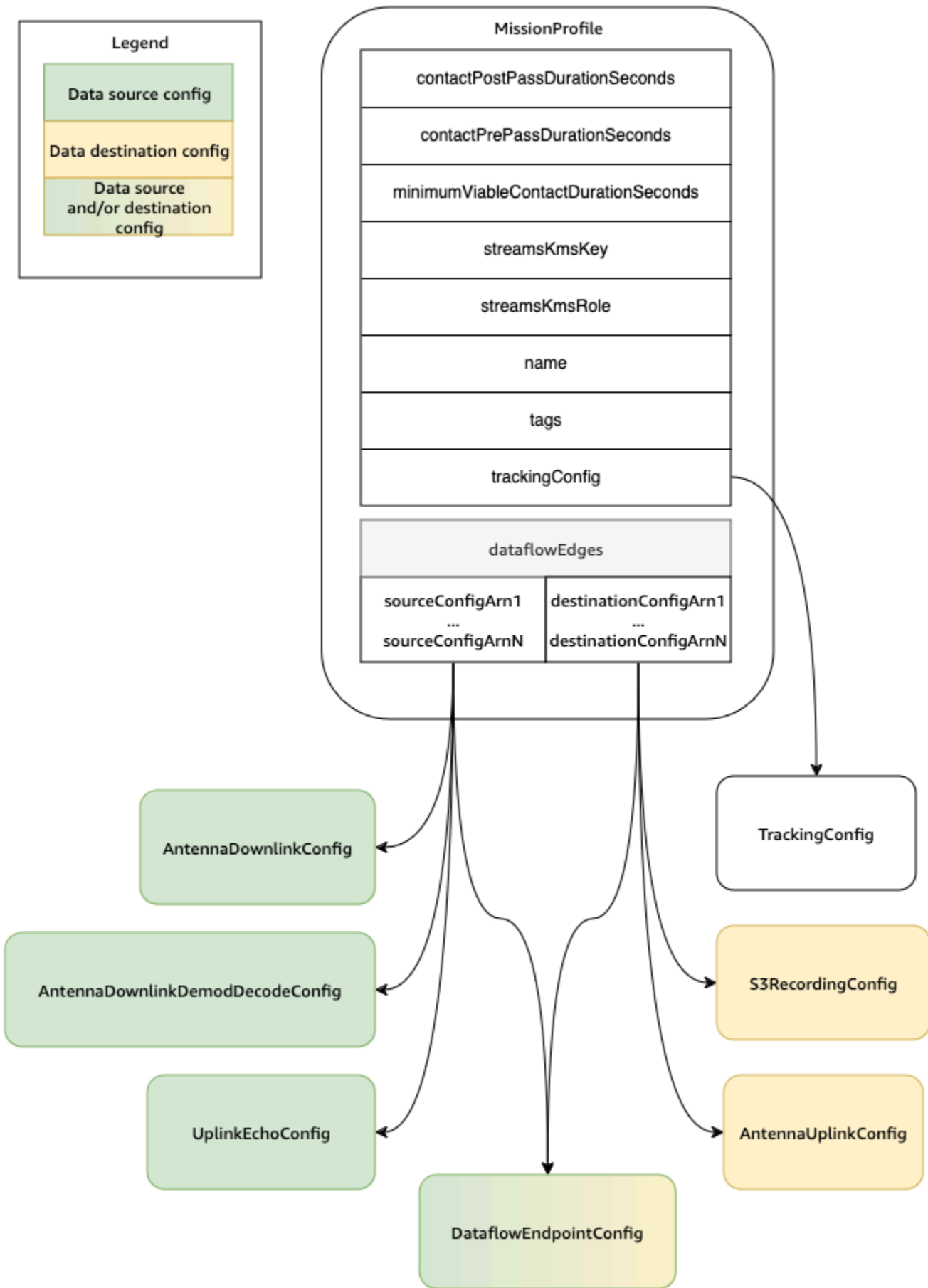
所有其他組態都包含在任務描述檔的 `dataflowEdges` 欄位中。這些組態可視為資料流程節點，每個節點代表可傳送或接收資料的 AWS Ground Station 受管資源及其相關聯的組態。`dataflowEdges` 欄位定義需要哪些來源和目的地資料流程節點（組態）。單一資料流程邊緣是兩個組態 [Amazon Resource Name \(ARNs\)](#) 的清單，第一個是來源組態，第二個是目的地組態。透過在兩個組態之間指定資料流程邊緣，您可以告知在聯絡期間，資料應 AWS Ground Station 從何處和流向何處。如需詳細資訊，請參閱[使用 AWS Ground Station 組態](#)。

`contactPrePassDurationSeconds` 和 `contactPostPassDurationSeconds` 可讓您指定與接收 CloudWatch Event 通知的聯絡人相關的時間。如需與您的聯絡相關的事件時間表，請閱讀 [了解聯絡生命週期](#)。

任務設定檔的 `name` 欄位有助於區別您建立的任務設定檔。

`streamsKmsRole` 和 `streamsKmsKey` 用於定義 AWS Ground Station 用來透過 AWS Ground Station 代理程式交付資料的加密。請參閱 [傳輸期間的資料加密 AWS Ground Station](#)。

`telemetrySinkConfigArn` 欄位是選用的，可讓您在聯絡期間啟用 AWS Ground Station 遙測。指定時，會在您的聯絡人執行期間，將近乎即時的遙測資料 AWS Ground Station 串流至您的帳戶。如需設定和使用遙測的詳細資訊，請參閱 [使用遙測](#)。



下列文件包含參數和範例的完整清單。

- [AWS::GroundStation::MissionProfile CloudFormation 資源類型](#)

使用 AWS Ground Station 組態

組態是資源，AWS Ground Station 用來定義聯絡人每個層面的參數。將您想要的組態新增至任務設定檔，然後執行接收時，即會使用該任務設定檔。您可以定義多種不同類型的組態。組態可分為三個類別：

- 追蹤組態
- Dataflow 組態
- 遙測組態

TrackingConfig 是唯一的追蹤組態類型。它用於在聯絡期間設定天線的自動追蹤設定，在任務設定檔中是必要的。

可在任務設定檔資料流程中使用的組態可視為資料流程節點，每個節點代表可傳送或接收資料的 AWS Ground Station 受管資源。任務描述檔需要至少一對這些組態，一個代表資料來源，另一個代表目的地。這些組態摘要如下表所示。

組態名稱	資料流程來源/目的地
AntennaDownlinkConfig	來源
AntennaDownlinkDemodDecodeConfig	來源
UplinkEchoConfig	來源
S3RecordingConfig	目標
AntennaUplinkConfig	目標
DataflowEndpointConfig	來源和/或目的地

TelemetrySinkConfig 是唯一的遙測組態類型。它用於設定遙測資料在聯絡期間交付的位置，在任務設定檔中是選用的。包含時，會在您的聯絡人執行期間，將近乎即時的遙測 AWS Ground Station 串流至您的帳戶。

如需如何使用 CloudFormation、或 AWS Ground Station API 在組態上執行操作的詳細資訊 AWS Command Line Interface，請參閱下列文件。特定組態類型的文件連結也提供如下。

- [AWS::GroundStation::Config CloudFormation 資源類型](#)
- [Config AWS CLI 參考](#)
- [Config API 參考](#)

追蹤組態

您可以在任務設定檔中使用追蹤組態，以判斷應否在接收期間啟用自動追蹤。此組態有單一參數：autotrack。autotrack 參數可有下列值：

- REQUIRED - 接收需要自動追蹤。
- PREFERRED - 接收偏好自動追蹤，但沒有自動追蹤仍可執行接收。
- REMOVED - 無任何自動追蹤應用於接收。

AWS Ground Station 將利用程式設計追蹤，當未使用自動追蹤時，該追蹤會根據您的暫時性變化來指向。如需如何建構 ephemeris 的詳細資訊 [了解 AWS Ground Station 如何使用 ephemerides](#)，請參閱。

Autotrack 將使用程式追蹤，直到找到預期的訊號為止。一旦發生這種情況，它將繼續根據訊號的強度進行追蹤。

如需如何使用 CloudFormation、或 AWS Ground Station API 追蹤組態執行操作的詳細資訊 AWS Command Line Interface，請參閱下列文件。

- [AWS::GroundStation::Config TrackingConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱 trackingConfig -> (structure) 一節)
- [TrackingConfig API 參考](#)

天線下行組態

您可以使用天線下行組態，在聯絡期間設定下行的天線。它們由頻譜組態組成，指定應該在下行聯絡期間使用的頻率、頻寬和極化。

此組態代表資料流程中的來源節點。它負責將無線電頻率資料數位化。從此節點串流的資料將遵循訊號資料/IP 格式。如需如何使用此組態建構資料流程的詳細資訊，請參閱 [使用資料流程](#)

如果您的下行使用案例需要解調或解碼，請參閱 [天線下行解調解碼組態](#)。

如需如何使用 CloudFormation AWS Command Line Interface、或 AWS Ground Station API 在天線下行組態上執行操作的詳細資訊，請參閱下列文件。

- [AWS::GroundStation::Config AntennaDownlinkConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱 `antennaDownlinkConfig -> (structure)` 一節)
- [AntennaDownlinkConfig API 參考](#)

天線下行解調解碼組態

天線下行示範解碼組態是一種更複雜且可自訂的組態類型，可用來執行具有解調和/或解碼的下行聯絡人。如果您有興趣執行這些類型的聯絡，請透過 開立 AWS 支援 票證 [AWS Support Center Console](#)。我們會協助您為使用案例定義正確的組態和任務設定檔。

此組態代表資料流程中的來源節點。它負責將無線電頻率資料數位化，並執行指定的解調和解碼。從此節點串流的資料將遵循解調/解碼的資料/IP 格式。如需如何使用此組態建構資料流程的詳細資訊，請參閱 [使用資料流程](#)

如需如何使用 AWS Command Line Interface、或 AWS Ground Station API 在天線下行示範解碼組態上執行操作的詳細資訊 CloudFormation，請參閱下列文件。

- [AWS::GroundStation::Config AntennaDownlinkDemodDecodeConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱 `antennaDownlinkDemodDecodeConfig -> (structure)` 一節)
- [AntennaDownlinkDemodDecodeConfig API 參考](#)

天線上行組態

您可以使用天線上行組態，在聯絡期間設定上行的天線。它們由具有頻率、極化和目標有效等向輻射功率 (EIRP) 的頻譜組態組成。如需如何設定上行回傳聯絡人的詳細資訊，請參閱 [天線上行回應組態](#)。

此組態代表資料流程中的目的地節點。它會將提供的數位無線電頻率資料訊號轉換為類比訊號，並將其發出以供衛星接收。串流到此節點的資料預期符合訊號資料/IP 格式。如需如何使用此組態建構資料流程的詳細資訊，請參閱 [使用資料流程](#)

如需如何使用 CloudFormation AWS Command Line Interface、或 AWS Ground Station API 在天線上行組態上執行操作的詳細資訊，請參閱下列文件。

- [AWS::GroundStation::Config AntennaUplinkConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱antennaUplinkConfig -> (structure)一節)
- [AntennaUplinkConfig API 參考](#)

天線上行回應組態

上行回應組態會通知天線如何執行上行回應。上行回應可用來驗證傳送到您太空船的命令，並執行其他進階任務。這可透過記錄由 AWS Ground Station 天線傳輸的實際訊號（即上行連結）來實現。這會回應天線傳送回資料流程端點的訊號，且應符合傳輸的訊號。上行回應組態包含上行組態的 ARN。天線在執行上行回應時，使用來自 ARN 指向之上行組態的參數。

此組態代表資料流程中的來源節點。從此節點串流的資料將符合訊號資料/IP 格式。如需如何使用此組態建構資料流程的詳細資訊，請參閱 [使用資料流程](#)

如需如何使用 AWS Command Line Interface、或 AWS Ground Station API 在上行回應組態上執行操作的詳細資訊 CloudFormation，請參閱下列文件。

- [AWS::GroundStation::Config UplinkEchoConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱uplinkEchoConfig -> (structure)一節)
- [UplinkEchoConfig API 參考](#)

資料流程端點組態

Note

Dataflow 端點組態僅用於將資料交付至 Amazon EC2，而不會用於將資料交付至 Amazon S3。

您可以使用資料流程端點組態來指定資料流程端點群組中的哪些資料流程端點，以及您希望資料在聯絡期間從哪個端點傳輸到哪個端點。資料流程端點組態的兩個參數會指定資料流程端點的名稱和區域。保留聯絡人時，AWS Ground Station 會分析您指定的任務設定檔，並嘗試尋找區域中 AWS 包含任務設定檔中包含之資料流程端點組態所指定所有資料流程端點的資料流程端點群組。如果找到合適的資料流程端點群組，聯絡狀態會變成排程，否則會變成 FAILED_TO_SCHEDULE。如需聯絡人可能狀態的詳細資訊，請參閱 [AWS Ground Station 聯絡狀態](#)。

資料流程端點組態的 `dataflowEndpointName` 屬性會指定資料流程端點群組中的哪些資料流程端點，在聯絡期間資料將流向或流自哪個資料流程端點。

`dataflowEndpointRegion` 屬性會指定資料流程端點所在的區域。如果您的資料流程端點組態中指定了區域，會 AWS Ground Station 尋找指定區域中的資料流程端點。如果未指定區域，AWS Ground Station 將預設為聯絡人的地面工作站區域。如果您的資料流程端點的區域與聯絡人的地面工作站區域不同，則聯絡人會被視為跨區域資料交付聯絡人。如需跨區域資料流程的詳細資訊 [使用資料流程](#)，請參閱。

如需資料流程不同命名方案如何讓您的使用案例受益的 [使用 AWS Ground Station Dataflow 端點群組秘訣](#)，請參閱。

如需如何使用此組態建構資料流程的詳細資訊，請參閱 [使用資料流程](#)

如需如何使用 AWS Command Line Interface、或 AWS Ground Station API 在資料流程端點組態上執行操作的詳細資訊 CloudFormation，請參閱下列文件。

- [AWS::GroundStation::Config DataflowEndpointConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱 `dataflowEndpointConfig -> (structure)` 一節)
- [DataflowEndpointConfig API 參考](#)

Amazon S3 錄製組態

Note

Amazon S3 記錄組態僅用於將資料交付至 Amazon S3，不會用於將資料交付至 Amazon EC2。

此組態代表資料流程中的目的地節點。此節點會將來自資料流程來源節點的傳入資料封裝為 pcap 資料。如需如何使用此組態建構資料流程的詳細資訊，請參閱 [使用資料流程](#)

您可以使用 S3 錄製組態來指定 Amazon S3 儲存貯體，您想要將下行資料與使用的命名慣例一起交付到該儲存貯體。以下指定有關這些參數的限制和詳細資訊：

- Amazon S3 儲存貯體的名稱必須以 `aws-groundstation` 開頭。
- IAM 角色必須具有信任政策，允許 `groundstation.amazonaws.com` 服務主體擔任該角色。如需範例，請參閱 [下列範例信任政策](#) 一節。在組態建立期間，組態資源 ID 不存在，信任政策必須使用星號 (*) 取代 `your-config-id`，並且可以在使用組態資源 ID 建立之後更新。

範例信任政策

如需如何更新角色信任政策的詳細資訊，請參閱 [《IAM 使用者指南》中的管理 IAM 角色](#)。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "999999999999"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:groundstation:us-east-1:999999999999:config/s3-recording/your-config-id"
        }
      }
    }
  ]
}
```

- IAM 角色必須具有 IAM 政策，允許角色對儲存貯體執行 s3:GetBucketLocation 動作，並對儲存貯體的物件執行 s3:PutObject 動作。如果 Amazon S3 儲存貯體具有儲存貯體政策，則儲存貯體政策也必須允許 IAM 角色執行這些動作。如需範例，請參閱 [以下的範例角色政策](#) 一節。

範例角色政策

如需如何更新或連接角色政策的詳細資訊，請參閱 [《IAM 使用者指南》中的管理 IAM 政策](#)。

JSON

```

{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation"
      ],
      "Resource": [
        "arn:aws:s3:::your-bucket-name"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "s3:PutObject"
      ],
      "Resource": [
        "arn:aws:s3:::your-bucket-name/*"
      ]
    }
  ]
}

```

- 命名 S3 資料物件時將使用字首。您可以指定選用金鑰來取代，這些值將取代為來自您聯絡詳細資訊的對應資訊。例如，`{satellite_id}/{year}/{month}/{day}` 將取代的字首，並使用類似的輸出產生 `fake_satellite_id/2021/01/10`

替代的選用索引鍵：`{satellite_id} | {config-name} | {config-id} | {year} {month} | {day} |`

如需如何使用 CloudFormation AWS Command Line Interface、或 AWS Ground Station API 在 S3 錄製組態上執行操作的詳細資訊，請參閱下列文件。

- [AWS::GroundStation::Config S3RecordingConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱 `s3RecordingConfig -> (structure)` 一節)
- [S3RecordingConfig API 參考](#)

遙測接收器組態

您可以使用遙測接收器組態來指定您希望遙測資料在衛星聯絡期間交付的位置。遙測接收器組態是選用的，且會新增至您的任務設定檔，以排程已啟用遙測功能的聯絡人。以下指定有關這些參數的限制和詳細資訊：

- IAM 角色必須具有信任政策，允許groundstation.amazonaws.com服務主體擔任該角色。如需範例，[請參閱下列範例信任政策](#)一節。

範例信任政策

如需如何更新角色信任政策的詳細資訊，請參閱 [《IAM 使用者指南》中的管理 IAM 角色](#)。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.amazonaws.com"
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```

- IAM 角色必須具有允許角色在串流上執行 `kinesis:DescribeStream`、`kinesis:PutRecord` 和 `kinesis:PutRecords` 動作的 IAM 政策。如需範例，[請參閱以下的範例角色政策](#)一節。

範例角色政策

如需如何更新或連接角色政策的詳細資訊，請參閱 [《IAM 使用者指南》中的管理 IAM 政策](#)。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "kinesis:DescribeStream",

```

```
    "kinesis:PutRecord",
    "kinesis:PutRecords"
  ],
  "Resource": "arn:aws:kinesis:us-east-2:999999999999:stream/your-stream-name"
}
]
```

當您在任務設定檔中包含遙測接收器組態時，AWS Ground Station 會在聯絡期間將遙測資料串流至您的帳戶。如需遙測類型、資料格式和設定必要 AWS 資源的詳細資訊，請參閱 [使用遙測](#)。

如需如何使用 CloudFormation AWS Command Line Interface、或 AWS Ground Station API 在遙測接收器組態上執行操作的詳細資訊，請參閱下列文件。

- [AWS::GroundStation::Config TelemetrySinkConfig CloudFormation 屬性](#)
- [組態 AWS CLI 參考](#) (請參閱 `telemetrySinkConfig -> (structure)` 一節)
- [TelemetrySinkConfig API 參考](#)

使用 AWS Ground Station Dataflow 端點群組

Dataflow 端點會定義您希望在聯絡期間同步串流資料的位置。資料流程端點一律建立為資料流程端點群組的一部分。透過將多個資料流程端點包含在一個群組中，您宣告可在單一接收中一起使用全部的指定端點。例如，如果接收需要將資料分別傳送到三個資料流程端點，您必須在符合任務設定檔資料流程端點組態的單一資料流程端點群組中有三個端點。

Dataflow 端點群組版本

AWS Ground Station 支援兩個版本的資料流程端點群組：

- DataflowEndpointGroup - 原始實作，支援使用 [資料流程端點](#) 的上行和下行，以及僅 [AWS Ground Station 代理程式端點](#) 的下行
- DataflowEndpointGroupV2 - 更新版本，可同時支援 AWS Ground Station 客服人員端點的上行和下行資料流程，並改善清晰度和功能

Dataflow 端點群組比較

功能	DataflowEndpointGroup	DataflowEndpointGroupV2
支援的端點類型	DataflowEndpoint , AwsGroundStationAgentEndpoint	DownlinkAwsGroundStationAgentEndpoint、UplinkAwsGroundStationAgentEndpoint
支援上行的端點	DataflowEndpoint	UplinkAwsGroundStationAgentEndpoint
支援下行的端點	DataflowEndpoint , AwsGroundStationAgentEndpoint	DownlinkAwsGroundStationAgentEndpoint

DataflowEndpointGroupV2 旨在支援上行資料流程，並讓資料流程端點群組周圍的語言更清晰。針對所有新的使用案例，我們建議使用 [UplinkAwsGroundStationAgentEndpoint](#) 和 [DownlinkAwsGroundStationAgentEndpoint](#) 端點搭配 [DataflowEndpointGroupV2](#)。DataflowEndpointGroup 仍支援回溯相容性，但 DataflowEndpointGroupV2 提供增強功能和更清晰的組態選項。

 Tip

執行聯絡人時，資料流程端點會以您選擇的名稱識別。這些名稱在帳戶中不需要是唯一的名稱。這允許使用相同的任務設定檔同時執行不同衛星和天線的多個聯絡人。如果您有具有相同操作特性的衛星星座，這會很有用。您可以擴展資料流程端點群組的數量，以符合衛星星座所需的同時聯絡數目上限。

當資料流程端點群組中的一或多項資源正為某項接收使用時，整個群組在接收期間會為該接收保留。您可以同時執行多個聯絡人，但這些聯絡人必須在不同的資料流程端點群組上執行。

⚠ Important

Dataflow 端點群組必須處於 HEALTHY 狀態，才能使用它們排程聯絡人。如需如何對未處於 HEALTHY 狀態的資料流程端點群組進行故障診斷的資訊，請參閱 [針對未處於 HEALTHY 狀態的 DataflowEndpointGroups 進行故障診斷](#)。

如需如何使用 CloudFormation AWS Command Line Interface、或 AWS Ground Station API 在資料流程端點群組上執行操作的詳細資訊，請參閱下列文件。

- [AWS::GroundStation::DataflowEndpointGroup CloudFormation 資源類型](#)
- [Dataflow 端點群組 AWS CLI 參考](#)
- [Dataflow 端點群組 API 參考](#)

資料流程端點

資料流程端點群組的成員是資料流程端點。支援的端點類型取決於您使用的資料流程端點群組版本。

DataflowEndpointGroup 端點

DataflowEndpointGroup 支援使用[資料流程端點](#)的上行和下行，以及僅[AWS Ground Station 代理程式端點](#)的下行。對於這兩種類型的端點，您將在建立資料流程端點群組之前建立支援的建構（例如 IP 地址）。如需使用哪些資料流程端點類型以及如何設定支援建構[使用資料流程](#)的建議，請參閱。

下列各節說明兩種支援的端點類型。

⚠ Important

單一資料流程端點群組內的所有資料流程端點必須為相同類型。您無法將[AWS Ground Station 客服人員端點](#)與相同群組中的 [Dataflow 端點](#)混合。如果您的使用案例需要這兩種類型的端點，您必須為每個類型建立單獨的資料流程端點群組。

對於 DataflowEndpointGroupV2，您可以在相同群組中混合 [UplinkAwsGroundStationAgentEndpoint](#) 和 [DownlinkAwsGroundStationAgentEndpoint](#)。

AWS Ground Station 代理程式端點

Agent AWS Ground Station Endpoint 利用 AWS Ground Station Agent 做為軟體元件來終止連線。若要建構 AWS Ground Station 代理程式端點，您只會填入 EndpointDetails

AwsGroundStationAgentEndpoint的欄位。如需 AWS Ground Station 代理程式的詳細資訊，請參閱完整的 [AWS Ground Station 代理程式使用者指南](#)。

AwsGroundStationAgentEndpoint包含下列項目：

- Name - 資料流程端點名稱。若要讓聯絡人使用此資料流程端點，此名稱必須符合資料流程端點組態中使用的名稱。
- EgressAddress - 用來從代理程式輸出資料的 IP 和連接埠地址。
- IngressAddress - 用來將資料傳入代理程式的 IP 和連接埠地址。

Dataflow 端點

Dataflow 端點利用聯網應用程式做為軟體元件來終止連線。當您想要上行數位訊號資料、下行小於 50MHz 的數位訊號資料，或下行解調/解碼訊號資料時，請使用資料流程端點。若要建構 Dataflow 端點，您將填入 EndpointDetails 的 Endpoint 和 Security Details 欄位。

Endpoint包含下列項目：

- Name - 資料流程端點名稱。若要讓聯絡人使用此資料流程端點，此名稱必須符合資料流程端點組態中使用的名稱。
- Address - 使用的 IP 和連接埠地址。

SecurityDetails包含下列項目：

- roleArn - AWS Ground Station 將擔任在 VPC 中建立彈性網路界面 (ENIs) 之角色的 Amazon Resource Name (ARN)。這些 ENI 可做為聯絡期間串流之資料的傳入和傳出點。
- securityGroupIds - 要連接到彈性網路界面的安全群組。
- subnetIds - 子網路清單，其中 AWS Ground Station 可能會放置彈性網路介面，以將串流傳送至您的執行個體。如果指定多個子網路，它們必須可以彼此路由。如果子網路位於不同的可用區域 (AZs)，您可能會產生跨可用區域資料傳輸費用。

傳入的 IAM 角色roleArn必須具有信任政策，允許groundstation.amazonaws.com服務主體擔任該角色。如需範例，請參閱[以下範例信任政策](#)一節。在端點建立期間，端點資源 ID 不存在，因此信任政策必須使用星號 (*) *your-endpoint-id*。這可以在建立後更新為使用端點資源 ID，以便將信任政策範圍限定於該特定資料流程端點群組。

IAM 角色必須具有允許 AWS Ground Station 設定 ENIs 的 IAM 政策。如需範例，[請參閱以下的範例角色政策](#)一節。

範例信任政策

如需如何更新角色信任政策的詳細資訊，請參閱 [《IAM 使用者指南》](#) 中的 [管理 IAM 角色](#)。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "aws:SourceAccount": "999999999999"
        },
        "ArnLike": {
          "aws:SourceArn": "arn:aws:groundstation:us-east-1:999999999999:dataflow-endpoint-group/your-endpoint-id"
        }
      }
    }
  ]
}
```

範例角色政策

如需如何更新或連接角色政策的詳細資訊，請參閱 [《IAM 使用者指南》](#) 中的 [管理 IAM 政策](#)。

JSON

```
{
  "Version": "2012-10-17",
```

```
"Statement": [
  {
    "Effect": "Allow",
    "Action": [
      "ec2:CreateNetworkInterface",
      "ec2>DeleteNetworkInterface",
      "ec2:CreateNetworkInterfacePermission",
      "ec2>DeleteNetworkInterfacePermission",
      "ec2:DescribeSubnets",
      "ec2:DescribeVpcs",
      "ec2:DescribeSecurityGroups"
    ],
    "Resource": "*"
  }
]
```

DataflowEndpointGroupV2 端點

DataflowEndpointGroupV2 推出特殊的端點類型，可提供更清晰的組態和增強的功能：

- [UplinkAwsGroundStationAgentEndpoint](#) - 針對上行資料流程最佳化
- [DownlinkAwsGroundStationAgentEndpoint](#) - 針對下行資料流程最佳化

這些特殊化端點會將一般 [AwsGroundStationAgentEndpoint](#) 取代為方向特定的組態，讓您更輕鬆地設定和管理資料流程。

上行 AWS Ground Station 代理程式端點

[UplinkAwsGroundStationAgentEndpoint](#) 專為上行資料流程而設計，並提供更清晰的組態選項。當您需要將資料提供給 AWS Ground Station 以上行至衛星時，請使用此端點類型。

UplinkAwsGroundStationAgentEndpoint 包含下列項目：

- Name - 資料流程端點名稱。若要讓聯絡人使用此資料流程端點，此名稱必須符合資料流程端點組態中使用的名稱。
- IngressAddressAndPort - 單一 IP 和連接埠地址，用於向代理程式輸入資料
- AgentIpAndPortAddress - 代理程式通訊的連接埠範圍

下行 AWS Ground Station 代理程式端點

[DownlinkAwsGroundStationAgentEndpoint](#) 已針對下行資料流程進行最佳化，包括窄頻下行、寬頻解調/解碼，以及上行回聲案例。

`DownlinkAwsGroundStationAgentEndpoint` 包含下列項目：

- `Name` - 資料流程端點名稱。若要讓聯絡人使用此資料流程端點，此名稱必須符合資料流程端點組態中使用的名稱。
- `EgressAddressAndPort` - 代理程式資料輸出的單一 IP 和連接埠地址
- `AgentIpAndPortAddress` - 代理程式通訊的連接埠範圍

建立資料流程端點群組

您可以使用下列任一版本建立資料流程端點群組：

`CreateDataflowEndpointGroup`

使用 [CreateDataflowEndpointGroup](#) 實現回溯相容性，或當您需要使用一般 [AwsGroundStationAgentEndpoint](#) 或 [DataflowEndpoint](#) 類型時。

`CreateDataflowEndpointGroupV2`

針對新實作使用 [CreateDataflowEndpointGroupV2](#)，以利用支援上行和下行資料流程的特殊端點類型。此 API 僅支援 [UplinkAwsGroundStationAgentEndpoint](#) 和 [DownlinkAwsGroundStationAgentEndpoint](#)。

遷移考量事項

如果您目前正在使用 `DataflowEndpointGroup`，則可以繼續使用現有的組態，而無需變更。AWS Ground Station 會維持完整的回溯相容性。

如果您想要遷移以使用新的 `DataflowEndpointGroupV2`，且目前正在搭配 `Dataflow Endpoint Application` 使用 [DataflowEndpoint](#) 來接收資料，則需要遷移以改用 AWS Ground Station 代理程式。如果您已經使用 AWS Ground Station 代理程式進行下行，也可以使用相同的代理程式執行個體進行上行 - 不需要額外的代理程式執行個體。

若要遷移至 `DataflowEndpointGroupV2`：

1. 如果從 `DataflowEndpoint` 遷移，請遵循 AWS Ground Station 客服人員 [AWS Ground Station 使用者指南設定客服人員](#)

2. 識別您的資料流程方向並建立適當的端點類型 ([UplinkAwsGroundStationAgentEndpoint](#) 或 [DownlinkAwsGroundStationAgentEndpoint](#))
3. 建立參考這些端點的 [DataflowEndpointGroupV2](#)
4. 建立依名稱參考新 DataflowEndpointGroupV2 的新[資料流程端點組態](#)
5. 建立新的任務描述檔，將資料流程端點組態參考為資料流程邊緣
6. 使用新的任務設定檔來排程聯絡人
7. 在部署到生產環境之前測試您的組態

如需完整工作流程的詳細資訊，請參閱 [了解 AWS Ground Station 核心元件](#) 和 [建立組態](#)。

使用 AWS Ground Station 代理程式

AWS Ground Station 客服人員可讓您在 AWS Ground Station 聯絡期間接收（下行）同步寬頻數位中繼頻率 (DigIF) 資料流程。

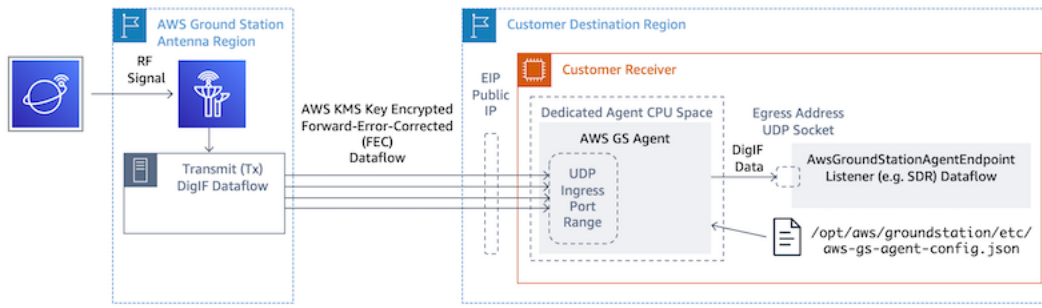
運作方式

您可以選擇兩個資料交付選項：

1. 資料交付至 EC2 執行個體 - 資料交付至您擁有的 EC2 執行個體。您可以管理 AWS Ground Station 代理程式。如果您需要近乎即時的資料處理，此選項可能最適合您。如需 EC2 資料交付的相關資訊，請參閱[使用資料流程](#)一節。
2. 資料交付至 S3 儲存貯體 - 資料交付至您的 AWS S3 儲存貯體完全由 管理 AWS Ground Station。如需 S3 資料交付的相關資訊，請參閱 [開始使用](#) 指南。

這兩種資料交付模式都需要您建立一組 AWS 資源。強烈建議您使用 CloudFormation 來建立 AWS 資源，以確保可靠性、準確性和支援能力。每個聯絡人只能將資料交付至 EC2 或 S3，但不能同時交付至兩者。

下圖顯示使用軟體定義的無線電 (SDR) 或類似接聽程式，從 AWS Ground Station 天線區域到 EC2 執行個體的 DigIF 資料流程。



其他資訊

如需詳細資訊，請參閱完整的[AWS Ground Station 客服人員使用者指南](#)。

開始使用

開始之前，您應該先熟悉其中的基本概念 AWS Ground Station。如需詳細資訊，請參閱[AWS Ground Station 運作方式](#)。

以下是 AWS Identity and Access Management (IAM) 的最佳實務以及您需要的許可。設定適當的角色之後，您可以開始遵循其餘的步驟。

註冊 AWS 帳戶

如果您沒有 AWS 帳戶，請完成下列步驟來建立一個。

註冊 AWS 帳戶

1. 開啟 <https://portal.aws.amazon.com/billing/signup>。
2. 請遵循線上指示進行。

部分註冊程序需接收來電或簡訊，並在電話鍵盤輸入驗證碼。

當您註冊時 AWS 帳戶，AWS 帳戶根使用者會建立。根使用者有權存取該帳戶中的所有 AWS 服務和資源。作為安全最佳實務，請將管理存取權指派給使用者，並且僅使用根使用者來執行[需要根使用者存取權的任務](#)。

AWS 會在註冊程序完成後傳送確認電子郵件給您。您可以隨時登錄 <https://aws.amazon.com/> 並選擇我的帳戶，以檢視您目前的帳戶活動並管理帳戶。

建立具有管理存取權的使用者

註冊後 AWS 帳戶，請保護 AWS 帳戶根使用者、啟用 AWS IAM Identity Center 和建立管理使用者，以免將根使用者用於日常任務。

保護您的 AWS 帳戶根使用者

1. 選擇根使用者並輸入 AWS 帳戶您的電子郵件地址，以帳戶擁有者[AWS 管理主控台](#)身分登入。在下一頁中，輸入您的密碼。

如需使用根使用者登入的說明，請參閱 AWS 登入 使用者指南中的[以根使用者身分登入](#)。

2. 若要在您的根使用者帳戶上啟用多重要素驗證 (MFA)。

如需說明，請參閱《IAM 使用者指南》中的[為您的 AWS 帳戶 根使用者（主控台）啟用虛擬 MFA 裝置](#)。

建立具有管理存取權的使用者

1. 啟用 IAM Identity Center。

如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的[啟用 AWS IAM Identity Center](#)。

2. 在 IAM Identity Center 中，將管理存取權授予使用者。

如需使用 IAM Identity Center 目錄 做為身分來源的教學課程，請參閱AWS IAM Identity Center 《使用者指南》中的[使用預設值設定使用者存取 IAM Identity Center 目錄](#)。

以具有管理存取權的使用者身分登入

- 若要使用您的 IAM Identity Center 使用者簽署，請使用建立 IAM Identity Center 使用者時傳送至您電子郵件地址的簽署 URL。

如需使用 IAM Identity Center 使用者登入的說明，請參閱AWS 登入 《使用者指南》中的[登入 AWS 存取入口網站](#)。

指派存取權給其他使用者

1. 在 IAM Identity Center 中，建立一個許可集來遵循套用最低權限的最佳實務。

如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的[建立許可集](#)。

2. 將使用者指派至群組，然後對該群組指派單一登入存取權。

如需指示，請參閱《AWS IAM Identity Center 使用者指南》中的[新增群組](#)。

將 AWS Ground Station 許可新增至 AWS 您的帳戶

若要在不要求管理使用者 AWS Ground Station 的情況下使用，您需要建立新的政策並將其連接到 AWS 您的帳戶。

1. 登入 AWS 管理主控台 並開啟 [IAM 主控台](#)。
2. 建立新政策。使用下列步驟：

- a. 在導覽窗格中，選擇 Policies (政策)，然後選擇 Create Policy (建立政策)。
 - b. 在 JSON 標籤中，使用下列值之一編輯 JSON。使用最適合您的應用程式的 JSON。
- 對於 Ground Station 管理權限，將動作設定為 `Groundstation : *`，如下所示：

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "groundstation:*"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

- 對於唯讀權限，請將 Action (動作) 設為 `groundstation:get*`、`groundstation:list*` 和 `groundstation:describe*`，如下所示：

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "groundstation:Get*",
        "groundstation:List*",
        "groundstation:Describe*"
      ],
      "Resource": [
        "*"
      ]
    }
  ]
}
```

```
    ]
  }
}
```

- 如需透過多重驗證來增加安全性，將 Action (動作) 設定為 `groundstation:*` 和將 Condition/Bool (條件/布林) 設定為 `aws:MultiFactorAuthPresent:true`，如下所示：

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": "groundstation:*",
      "Resource": "*",
      "Condition": {
        "Bool": {
          "aws:MultiFactorAuthPresent": true
        }
      }
    }
  ]
}
```

3. 在 IAM 主控台中，將您建立的政策連接到所需的使用者。

如需 IAM 使用者和連接政策的詳細資訊，請參閱 [IAM 使用者指南](#)。

加入衛星

將衛星加入 AWS Ground Station 是一個多步驟程序，涉及資料收集、技術驗證、頻譜授權，以及整合和測試。也需要保密協議 (NDAs)。

客戶加入程序概觀

衛星加入是一種手動程序，可在 AWS Ground Station 主控台頁面的 [衛星和資源](#) 區段中找到。以下說明整體程序。

1. 檢閱 [AWS Ground Station 位置](#) 區段，判斷您的衛星是否符合地理和無線電頻率特性。

- 若要開始讓衛星加入 AWS Ground Station，請在 AWS Ground Station 主控台頁面的[衛星和資源區段上提交衛星](#)加入問卷。請包含任務和衛星需求的簡短摘要，包括您的組織名稱、所需的頻率、衛星的啟動時間、衛星的軌道類型，以及您是否計劃使用[使用 AWS Ground Station 數位分身功能](#)。
- 審核並核准您的請求後，AWS Ground Station 會在您計劃使用的特定位置申請法規授權。此步驟的持續時間會根據位置和任何現有法規而有所不同。
- 取得此核准後，您的衛星就會顯示供您使用。AWS Ground Station 會傳送成功更新的通知給您。

(選用) 命名衛星

加入後，您可能想要將名稱新增至衛星記錄，以便更輕鬆地辨識它。AWS Ground Station 使用聯絡人頁面時，主控台能夠顯示使用者定義的衛星名稱以及 Norad ID。顯示衛星名稱可讓您在排程時更輕鬆地選取正確的衛星。若要這樣做，可以使用[標籤](#)。

您可以使用 AWS CLI 或其中一個 AWS 開發套件，透過 tag-resource API 來標記 AWS Ground Station SDKs。 <https://docs.aws.amazon.com/cli/latest/reference/groundstation/tag-resource.html> 本指南將介紹如何使用 AWS Ground Station CLI 在中標記公有廣播衛星 Aqua (Norad ID 27424)us-west-2。

AWS Ground Station CLI

AWS CLI 可用來與 互動 AWS Ground Station。使用 AWS CLI 標記您的衛星之前，必須滿足下列 AWS CLI 先決條件：

- 確定 AWS CLI 已安裝。如需安裝的資訊 AWS CLI，請參閱[安裝 AWS CLI 第 2 版](#)。
- 確定 AWS CLI 已設定。如需設定的相關資訊 AWS CLI，請參閱[設定 AWS CLI 第 2 版](#)。
- 將常用的組態設定和登入資料儲存在 AWS CLI 維護的檔案中。您需要這些設定和登入資料來預留和管理 AWS Ground Station 聯絡人 AWS CLI。如需儲存組態和登入資料設定的詳細資訊，請參閱[組態和登入資料檔案設定](#)。

AWS CLI 設定好並準備好使用後，請檢閱 [AWS Ground Station CLI 命令參考](#) 頁面，以熟悉可用的命令。使用此服務時，請遵循 AWS CLI 命令結構，並使用 為命令加上字首 groundstation，以指定 AWS Ground Station 做為您要使用的服務。如需 AWS CLI 命令結構的詳細資訊，請參閱 [AWS CLI 頁面中的命令結構](#)。以下提供命令結構範例。

```
aws groundstation <command> <subcommand> [options and parameters]
```

命名衛星

首先，您需要取得要標記之衛星的 ARN (ASN)。這可以透過 AWS CLI 中的 [list-satellites](#) API 完成：

```
aws groundstation list-satellites --region us-west-2
```

執行上述 CLI 命令將傳回類似以下的輸出：

```
{
  "satellites": [
    {
      "groundStations": [
        "Ohio 1",
        "Oregon 1"
      ],
      "noradSatelliteID": 27424,
      "satelliteArn":
"arn:aws:groundstation::111111111111:satellite/11111111-2222-3333-4444-555555555555",
      "satelliteId": "11111111-2222-3333-4444-555555555555"
    }
  ]
}
```

尋找您要標記的衛星，並記下 `satelliteArn`。標記的一個重要注意事項是，[Tag-resource](#) API 需要區域 ARN，而 [list-satellites](#) 傳回的 ARN 是全域的。對於下一個步驟，您應該將 ARN 擴大為您想要查看標籤的區域（可能是您排程的區域）。在此範例中，我們使用 `us-west-2`。透過此變更，ARN 將來自：

```
arn:aws:groundstation::111111111111:satellite/11111111-2222-3333-4444-555555555555
```

至：

```
arn:aws:groundstation:us-west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555
```

若要在主控台中顯示衛星名稱，衛星必須具有 "Name" 標籤做為金鑰。此外，因為我們使用的是 AWS CLI，引號必須以反斜線逸出。標籤看起來會類似：

```
{\"Name\": \"AQUA\"}
```

接著，您將呼叫 [tag-resource](#) API 來標記衛星。這可以使用 AWS CLI 類似 來完成：

```
aws groundstation tag-resource --region us-west-2 --resource-arn  
arn:aws:groundstation:us-  
west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555 --tags  
'{\"Name\":\"AQUA\"}'
```

執行此操作後，您將能夠在 主控台中 AWS Ground Station 看到您為衛星設定的名稱。

變更衛星的名稱

如果您想要變更衛星的名稱，您可以使用相同的“Name”金鑰再次使用衛星 ARN 呼叫 [tag-resource](#)，但在標籤中具有不同的值。這將更新現有的標籤，並在主控台中顯示新名稱。對此的範例呼叫如下所示：

```
aws groundstation tag-resource --region us-west-2 --resource-arn  
arn:aws:groundstation:us-  
west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555 --tags  
'{\"Name\":\"NewName\"}'
```

移除衛星的名稱

可以使用 [untag-resource](#) API 移除衛星的名稱集。此 API 需要具有標籤所在區域的衛星 ARN，以及標籤索引鍵清單。對於名稱，標籤索引鍵為“Name”。使用 AWS CLI 呼叫此 API 的範例如下所示：

```
aws groundstation untag-resource --region us-west-2 --resource-arn  
arn:aws:groundstation:us-  
west-2:111111111111:satellite/11111111-2222-3333-4444-555555555555 --tag-keys Name
```

公有廣播衛星

除了加入您自己的衛星之外，您還可以請求加入支援公開廣播衛星，以提供可公開存取的下行通訊路徑。這可讓您使用 從這些衛星 AWS Ground Station 下行資料。

Note

您將無法向上連結至這些衛星。您只能使用可公開存取的下行通訊路徑。

AWS Ground Station 支援加入下列衛星以下行直接廣播資料：

- Aqua
- SNPP
- JPSS-1/NOAA-20
- Terra

加入後，即可存取這些衛星以供立即使用。AWS Ground Station 會維護許多預先設定的 CloudFormation 範本，讓您更輕鬆地開始使用服務。如需 AWS Ground Station 如何使用的範例 [範例任務設定檔組態](#)，請參閱。

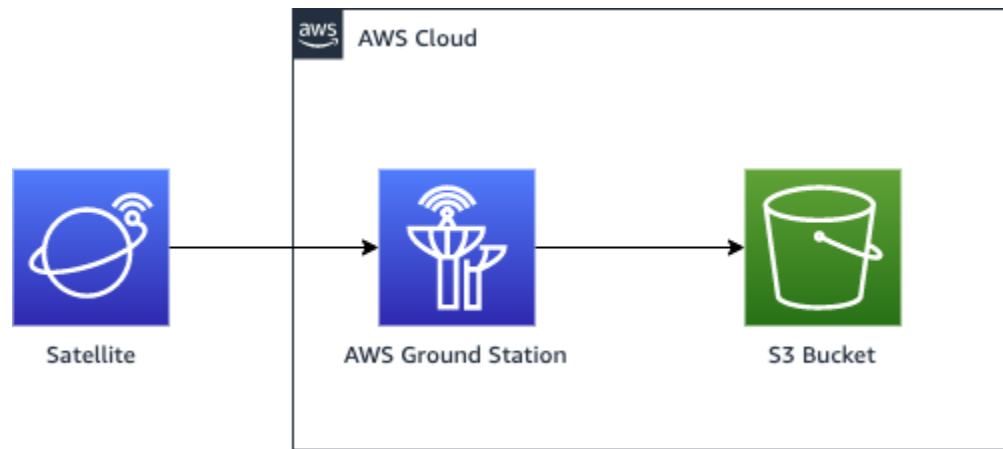
如需有關這些衛星，以及這些衛星傳輸的資料種類詳細資訊，請參閱 [Aqua](#)、[JPSS-1/NOAA-20 and SNPP](#) 和 [Terra](#)。

規劃您的資料流程通訊路徑

您可以選擇衛星上每個通訊路徑的同步和非同步通訊。根據您的衛星和使用案例，您可能需要一種或兩種類型。同步通訊路徑允許近乎即時的上行連結，以及窄頻和寬頻下行操作。非同步通訊路徑僅支援窄頻和寬頻下行操作。

非同步資料交付

透過將資料交付至 Amazon S3，您的聯絡資料會以非同步方式交付至您帳戶中的 Amazon S3 儲存貯體。您的聯絡資料會以封包擷取 (pcap) 檔案傳遞，以允許將聯絡資料重播至軟體定義的無線電 (SDR)，或從 pcap 檔案擷取承載資料進行處理。當天線硬體收到聯絡資料時，pcap 檔案每 30 秒會交付到您的 Amazon S3 儲存貯體，以允許在需要時在聯絡期間處理聯絡資料。收到後，您可以使用自己的後製處理軟體或使用 Amazon SageMaker AI 或 Amazon Rekognition 等其他 AWS 服務來處理資料。資料交付至 Amazon S3 僅適用於從衛星下行資料；無法從 Amazon S3 將資料上行至衛星。



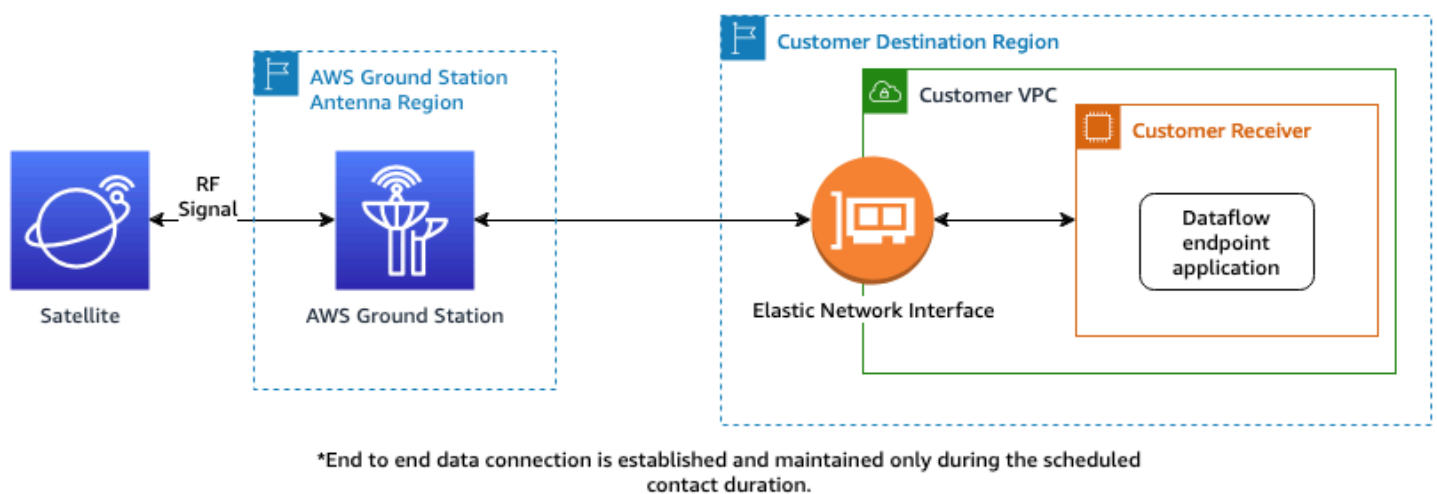
若要使用此路徑，您需要為 建立 Amazon S3 儲存貯體 AWS Ground Station ，以將資料交付至其中。在下一個步驟中，您也需要在下一個步驟中建立 S3 錄製組態。如需儲存貯體命名 [Amazon S3 錄製組態](#) 的限制，以及如何指定 檔案使用的命名慣例，請參閱。

同步資料交付

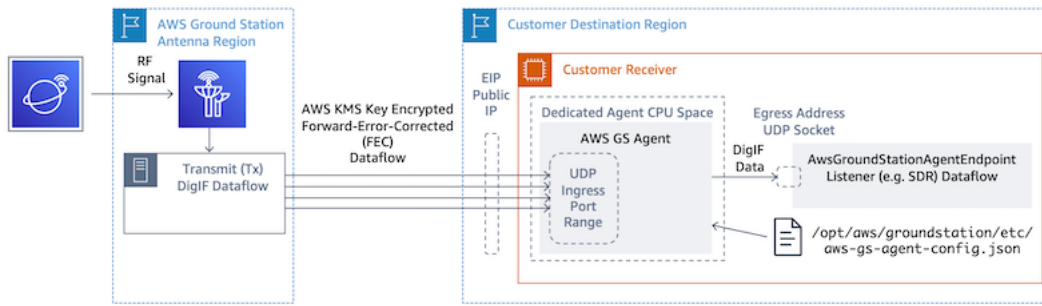
將資料交付至 Amazon EC2 時，您的聯絡資料會串流到 Amazon EC2 執行個體或從中串流。您可以在 Amazon EC2 執行個體上即時處理資料，或轉送資料以進行後續處理。

若要使用同步路徑，您需要使用 來設定 Amazon EC2 執行個體，並建立一或多個 Dataflow 端點群組。若要設定 Amazon EC2 執行個體，請參閱 [設定 Amazon EC2](#)。若要建立 Dataflow 端點群組，請參閱 [使用 AWS Ground Station Dataflow 端點群組](#)。

如果您使用的是資料流程端點組態，則會顯示通訊路徑。



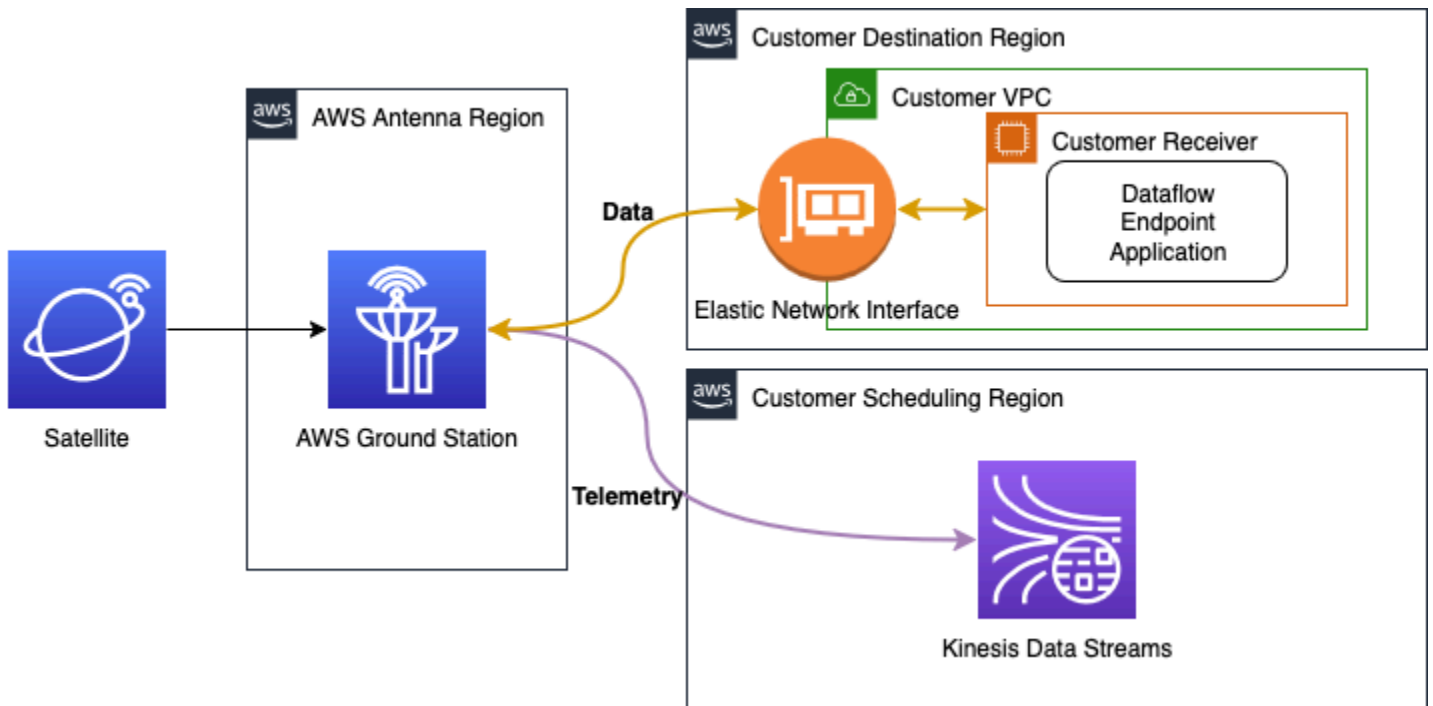
如果您使用 AWS Ground Station 代理程式組態，則會顯示通訊路徑。



規劃您的遙測

AWS Ground Station 遙測是一項選用功能，可在衛星聯絡期間將指標從 AWS Ground Station 天線串流至 AWS 您的帳戶。這可讓您近乎即時地監控聯絡人效能，並建置自訂監控解決方案。

透過 AWS Ground Station 遙測，來自 AWS Ground Station 天線的指標會直接串流到您的帳戶。遙測資料會在聯絡開始時開始串流，並在整個聯絡期間持續。遙測資料從天線硬體取樣時，會以近乎即時的方式交付至您的帳戶。收到後，您可以使用自己的後製處理軟體來處理資料，或使用 Amazon Data Firehose 或 等其他 AWS 服務 AWS Lambda。



在下一個步驟中，您將建立任務設定檔所需的組態。如果您想要啟用遙測，除了追蹤組態和資料流程組態之外，還會建立遙測接收器組態。如需詳細的設定說明，請參閱 [設定遙測](#)。

如需 TelemetrySinkConfig 的詳細資訊，請參閱 [遙測接收器組態](#)。

建立組態

在此步驟中，您已視需要識別衛星、通訊路徑和 IAM、Amazon EC2 和 Amazon S3 資源。在此步驟中，您將建立 AWS Ground Station 可存放其個別參數的組態。

資料交付組態

要建立的第一個組態與您希望資料交付的位置和方式相關。使用上一個步驟的資訊，您將建構下列許多組態類型。

- [Amazon S3 錄製組態](#) - 將資料交付到您的 Amazon S3 儲存貯體。
- [資料流程端點組態](#) - 將資料交付到您的 Amazon EC2 執行個體。

遙測組態（選用）

如果您想要在聯絡期間接收近乎即時的遙測，您可以建立 TelemetrySinkConfig。此組態是選用的，並指定 AWS Ground Station 將在何處傳送遙測資料。

- [遙測接收器組態](#) - 將遙測資料交付至您的帳戶。

如需詳細的設定說明，請參閱 [設定遙測](#)。

衛星組態

衛星組態會關聯 AWS Ground Station 如何與您的衛星通訊。您將參考在 [中](#) 收集的資訊 [加入衛星](#)。

- [追蹤組態](#) - 設定偏好在聯絡期間如何實際追蹤您的車輛。這是任務設定檔建構的必要項目。
- [天線下行組態](#) - 交付數位化的無線電頻率資料。
- [天線下行解調解碼組態](#) - 交付解調和解碼的無線電頻率資料。
- [天線上行組態](#) - 將資料上行至您的衛星。
- [天線上行回應組態](#) - 交付上行訊號資料的回應。

建立任務描述檔

透過在上一個步驟中建構的組態，您已識別如何追蹤衛星、與衛星通訊的可能方式，以及如何在聯絡執行期間啟用近乎即時的遙測。在此步驟中，您將建構一或多個任務設定檔。任務描述檔代表可能組態的彙總成可接著排程和操作的預期行為。

如需最新參數，請參閱 [AWS::GroundStation::MissionProfile CloudFormation 資源類型](#)

1. 為您的任務描述檔命名。這可讓您快速了解其在系統中的使用情況。例如，如果您有個別satellite-wideband-narrowband-nominal-operations操作和satellite-narrowband-emergency-operations。
2. 設定您的追蹤組態。
3. 設定您的最短可行聯絡持續時間。這可讓您篩選潛在的聯絡人，以符合您的任務需求。
4. 設定用於在傳輸期間加密資料的 streamsKmsKey 和 streamsKmsRole。這用於所有 AWS Ground Station 客服人員資料流程。
5. 設定您的資料流程。使用您在上一個步驟中建立的組態，建立您的資料流程以符合電信業者訊號。
6. 【選用】 設定您的傳遞前和傳遞後聯絡持續時間秒數。這用於在聯絡之前和之後分別發出每個聯絡事件。如需詳細資訊，請參閱[AWS Ground Station 使用 事件自動化](#)。
7. 【選用】 設定您的telemetrySinkConfigArn 以在聯絡期間啟用遙測。這可讓您直接在帳戶中接收近乎即時的遙測，以進行監控和分析。如需詳細資訊，請參閱[使用遙測](#)。
8. 【選用】 您可以將標籤與任務描述檔建立關聯。這些可用於協助以程式設計方式區分您的任務描述檔。

您可以參考 [範例任務設定檔組態](#)，只查看一些潛在的組態。

了解後續步驟

現在您已有加入的衛星和有效的任務描述檔，您已準備好排定聯絡人並與衛星通訊 AWS Ground Station。

您可以透過下列其中一種方式排程聯絡人：

- [AWS Ground Station 主控台](#)。
- CLI [reserve-contact](#) AWS 命令。
- AWS 開發套件。 [ReserveContact](#) API。

如需有關 如何 AWS Ground Station 追蹤衛星軌跡以及如何使用該資訊的資訊，請參閱 [了解 AWS Ground Station 如何使用 ephemerides](#)。

AWS Ground Station 會維護許多預先設定的 CloudFormation 範本，讓您更輕鬆地開始使用服務。如需 AWS Ground Station 如何使用的範例 [範例任務設定檔組態](#)，請參閱。

處理數位中繼頻率資料，或從提供給您的解調和解碼資料 AWS Ground Station，將取決於您的特定使用案例。下列部落格文章可協助您了解一些可用的選項：

- [使用 AWS Ground Station Amazon S3 資料交付的自動化地球觀察](#)（及其相關聯的 GitHub 儲存庫 [awslabs/aws-groundstation-eos-pipeline](#)）
- [使用 虛擬化衛星地面區段 AWS](#)
- [地球觀察使用 AWS Ground Station：如何引導](#)
- [使用 AWS Ground Station WideBand DigIF 和 Amphinicy Blink SDR](#)（及其相關聯的 GitHub 儲存庫 [aws-samples/aws-groundstation-wbdigif-snpp](#)）[建置高輸送量衛星資料下行架構](#) GitHub [aws-samples/aws-groundstation-wbdigif-snpp](#)

AWS Ground Station 位置

AWS Ground Station 提供與 AWS 基礎設施區域全球網路近在咫尺的全球地面站網路。您可以從任何支援的 AWS 區域設定使用這些位置。這包括交付資料的 AWS 區域。



尋找地面站點位置 AWS 的區域

AWS Ground Station 全球網路包含的地面站位置不是實際位於其連線的 [AWS 區域中](#)。您可以透過 AWS SDK [ListGroundStation](#) 回應擷取您有權存取的地面工作站清單。地面站點位置的完整清單如下所示，即將推出更多。請參閱入門指南，為您的衛星新增或修改網站核准。

Ground Station 名稱	Ground Station 位置	AWS 區域名稱	AWS 區域代碼	備註
阿拉斯加 1	美國阿拉斯加	美國西部 (奧勒岡)	us-west-2	不是實際位於 AWS 區域
巴林 1	巴林	Middle East (Bahrain)	me-south-1	
開普敦 1	南非開普敦	Africa (Cape Town)	af-south-1	

Ground Station 名稱	Ground Station 位置	AWS 區域名稱	AWS 區域代碼	備註
Dubbo 1	達博，澳洲	亞太地區 (悉尼)	ap-southeast-2	不是實際位於 AWS 區域
夏威夷 1	美國夏威夷	美國西部 (奧勒岡)	us-west-2	不是實際位於 AWS 區域
愛爾蘭 1	愛爾蘭	歐洲 (愛爾蘭)	eu-west-1	
俄亥俄州 1	美國俄亥俄州	美國東部 (俄亥俄)	us-east-2	
奧勒岡州 1	美國奧勒岡	美國西部 (奧勒岡)	us-west-2	
Punta Arenas 1	Punta Arenas , 智利	南美洲 (聖保羅)	sa-east-1	不是實際位於 AWS 區域
首爾 1	首爾、南韓	亞太區域 (首爾)	ap-northeast-2	
新加坡 1	新加坡	亞太區域 (新加坡)	ap-southeast-1	
斯德哥爾摩 1	瑞典斯德哥爾摩	Europe (Stockholm)	eu-north-1	

AWS Ground Station 支援的 AWS 區域

您可以透過 AWS 開發套件或從支援的 AWS 區域 AWS Ground Station 提供資料和設定您的聯絡人。您可以在端點和[AWS Ground Station 配額中檢視支援的區域及其相關聯的端點](#)。

數位分身可用性

[使用 AWS Ground Station 數位分身功能](#) 適用於所有 AWS Ground Station 可使用的 [AWS 區域](#)。數位分身地面站是生產地面站的確切副本，字首修改為「數位分身」地面站名稱。例如，「數位分身俄亥俄 1」是數位分身地面站，是「俄亥俄 1」生產地面站的確切副本。

專用天線

除了上述公開可用的地面站位置之外，AWS Ground Station 還提供專用天線。專用天線是一種自訂建置的天線系統，可代表您 AWS 管理。專用天線不限於現有的 AWS Ground Station 地面站位置，並且可以使用公有地面站以外的功能建置，如中所述[AWS Ground Station 站點功能](#)。不公開專用天線的位置和功能。

如需專用天線的詳細資訊，請參閱 [AWS Ground Station 專用天線](#)。若要進一步了解或開始使用專用天線，AWS 支援 請透過 聯絡 [AWS Support Center Console](#)。

在地面工作站檢視天線

每個地面站位置都有一或多個天線。您可以使用 [ListAntennas](#) API 在地面工作站檢視天線。此 API 會在指定的地面站傳回天線，包括每個天線的名稱。

天線資訊在與 [ListGroundStationReservations](#) API 結合時很有用，可了解地面工作站的容量和可用性。如需檢視保留的詳細資訊，請參閱 [檢視地面站點保留](#)。

若要呼叫 ListAntennas，您必須讓衛星加入到地面站，或具有地面站的方位海拔暫時性許可。如需詳細資訊，請參閱[提供方位提升暫時性資料](#)。

範例：列出地面工作站的天線

下列範例使用適用於 Python 的 AWS SDK (Boto3) 列出地面工作站的所有天線。

```
import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# The ground station ID to list antennas for.
# Use the ListGroundStations API to find available ground station IDs.
ground_station_id = "Ohio 1"

# List all antennas at a ground station.
# This is useful for understanding the capacity of a ground station
# and for planning multi-antenna operations.
print(f"Listing antennas for ground station '{ground_station_id}'...")

paginator = ground_station_client.get_paginator("list_antennas")
page_iterator = paginator.paginate(
```

```
groundStationId=ground_station_id,
PaginationConfig={
    "MaxItems": 100,
    "PageSize": 20,
},
)

for page in page_iterator:
    for antenna in page["antennaList"]:
        print(f"  Antenna: {antenna['antennaName']}")
        print(f"    Ground Station: {antenna['groundStationName']}")
        print(f"    Region: {antenna['region']}")
        print()
```

檢視地面站點保留

您可以使用 [ListGroundStationReservations](#) API，在地面工作站檢視跨天線的保留。預留代表天線上的時間區塊，包括您的排程聯絡人。[AWS Ground Station 專用天線](#) 客戶也會看到維護時段。

此資訊可協助您了解規劃聯絡排程時的天線可用性，並讓您了解地面工作站天線上發生的情況。

列出保留

若要列出保留，請呼叫 [ListGroundStationReservations](#)，其中包含地面工作站識別符和時間範圍。API 會在指定的時段內，傳回地面工作站所有天線的保留。

您看到的保留取決於您的存取層級：

- 公有 AWS Ground Station 客戶 — 您只能看到自己的聯絡人保留。不包含其他帳戶擁有的維護時段和聯絡人。
- AWS Ground Station 專用天線客戶 — 您可以在專用天線上查看所有保留，包括維護時段和其他帳戶排程的聯絡人。僅包含您擁有的聯絡人的聯絡人識別符。如需詳細資訊，請參閱[AWS Ground Station 專用天線](#)。

保留類型

每個保留都有一個類型，指出天線時間的用途：

- 聯絡人 — 聯絡人保留代表為衛星通訊預留的天線時間。保留開始和結束時間反映完整的天線保留，包括傳遞前和傳遞後時間，而不只是衛星傳遞時段。

- 維護 — 維護保留代表天線因維護而無法使用的期間。維護保留包括maintenanceType指出維護是計劃還是計劃外的。

程式碼範例

下列範例使用適用於 Python 的 AWS SDK (Boto3) 列出未來 7 天的地面站點保留，包括依保留類型篩選。

```
import boto3
from datetime import datetime, timezone, timedelta

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# The ground station ID to list reservations for
ground_station_id = "Ohio 1"

# Define the time range to query. Reservations include both your
# scheduled contacts and maintenance windows at the ground station.
start_time = datetime.now(timezone.utc)
end_time = start_time + timedelta(days=7)

# List all reservations at a ground station for the next 7 days.
# You can filter by reservation type to see only contacts or
# only maintenance windows.
print(f"Listing reservations for ground station '{ground_station_id}'...")
print(f"Time range: {start_time} to {end_time}")

paginator = ground_station_client.get_paginator("list_ground_station_reservations")
page_iterator = paginator.paginate(
    groundStationId=ground_station_id,
    startTime=start_time,
    endTime=end_time,
    PaginationConfig={
        "MaxItems": 100,
        "PageSize": 20,
    },
)

for page in page_iterator:
    for reservation in page["reservationList"]:
        reservation_type = reservation["reservationType"]
```

```
    antenna_name = reservation["antennaName"]
    res_start = reservation["startTime"]
    res_end = reservation["endTime"]

    print(f"  Type: {reservation_type}")
    print(f"    Antenna: {antenna_name}")
    print(f"    Start: {res_start}")
    print(f"    End: {res_end}")

    details = reservation["reservationDetails"]
    if "contact" in details:
        contact_id = details["contact"].get("contactId", "N/A")
        print(f"    Contact ID: {contact_id}")
    elif "maintenance" in details:
        maintenance_type = details["maintenance"]["maintenanceType"]
        print(f"    Maintenance Type: {maintenance_type}")

    print()

# For Dedicated Antenna customers, you can also filter to show only maintenance windows
print("Listing only maintenance reservations...")

page_iterator = paginator.paginate(
    groundStationId=ground_station_id,
    startTime=start_time,
    endTime=end_time,
    reservationTypes=["MAINTENANCE"],
    PaginationConfig={
        "MaxItems": 100,
        "PageSize": 20,
    },
)

for page in page_iterator:
    for reservation in page["reservationList"]:
        maintenance_type = reservation["reservationDetails"]["maintenance"][
            "maintenanceType"
        ]
        print(
            f"  {maintenance_type} maintenance on {reservation['antennaName']}: "
            f"{reservation['startTime']} to {reservation['endTime']}"
        )
```

AWS Ground Station 網站遮罩

每個 AWS Ground Station [天線位置](#) 都有相關聯的網站遮罩。這些遮罩會在指向某些方向時阻止該位置的天線傳輸或接收，通常接近地平線。遮罩可能會考慮：

- 天線周圍地理地形的功能 – 例如，這包含像是山巒或建築物等會封鎖無線電頻率 (RF) 訊號或防止傳輸的物體。
- 無線電頻率干擾 (RFI) – 這會影響接收（影響下行訊號到 AWS Ground Station 天線的外部 RFI 來源）和傳輸 (AWS Ground Station 天線傳輸的 RF 訊號對外部接收器造成負面影響) 的能力。
- 法律授權 – 每個區域中操作 AWS Ground Station 的本機網站授權可能包括特定限制，例如用於傳輸的最小高度角度。

這些網站遮罩可能會隨著時間而變更。例如，新的建築物可以在天線位置附近建構，RFI 來源可能會變更，或者法律授權可能會以不同的限制續約。根據保密協議 (NDA)，您可以使用 AWS Ground Station 網站遮罩。

客戶特定的遮罩

除了每個網站的 AWS Ground Station 網站遮罩之外，由於您自己的法律授權在指定區域中與衛星通訊的限制，您可能還有其他遮罩。您可以在 AWS Ground Station case-by-case 設定這類遮罩，以確保使用 AWS Ground Station 與這些衛星通訊時符合規範。如需詳細資訊，請聯絡 AWS Ground Station 團隊。

網站遮罩對可用聯絡時間的影響

網站遮罩有兩種：上行（傳輸）網站遮罩和下行（接收）網站遮罩。

使用 ListContacts 操作列出可用的聯絡時間時，AWS Ground Station 會根據衛星在下行遮罩上升的時間傳回可見性時間。可用的聯絡時間是以這個下行遮罩可見性時段為基礎。這可確保當您的衛星低於下行遮罩時，不會預留時間。

即使任務設定檔在資料流程邊緣包含 [天線上行組態](#)，[上行網站遮罩](#) 也不會套用至可用的聯絡時間。這可讓您使用下行的所有可用聯絡時間，即使上行時間由於上行網站遮罩而部分無法使用。不過，上行訊號在保留給衛星聯絡人的部分或全部時間內可能無法傳輸。排程上行傳輸時，您必須負責考量提供的上行遮罩。

無法用於上行的聯絡部分會根據聯絡期間的衛星軌跡，相對於天線位置的上行網站遮罩而有所不同。在上行和下行網站遮罩相似的區域中，此持續時間通常很短。在其他區域中，上行遮罩可能遠高於下行網

站遮罩，這可能會導致大量的聯絡持續時間無法使用上行。即使上行無法使用部分預留時間，仍會向您收取完整的聯絡時間費用。

AWS Ground Station 站點功能

為了簡化您的體驗，AWS Ground Station 決定一組常見的天線類型功能，然後將多個天線部署到地面站位置。加入步驟的一部分可確保您的衛星與特定位置的天線類型相容。預留聯絡人時，您可以間接判斷使用的天線類型。這可確保您在特定地面站點的體驗隨著時間保持不變，無論使用哪些天線。由於各種環境問題，例如現場的天氣，聯絡人的特定效能會有所不同。

目前，所有網站都支援下列功能：

Note

除非另有說明，否則下表中的每一列都會指出獨立的通訊路徑。存在重複的資料列，以反映我們的多通道功能，允許同時使用多個通訊路徑。

功能類型	頻率範圍	頻寬範圍	Polarization	Common Name (通用名稱)	備註
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP	X 頻帶寬頻下行	此功能需要使用 AWS Ground Station 代理程式 。
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP		
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP		Alaska 1 或 Punta Arenas 1 不支援此功能。
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	RHCP		每個位置每個極化的彙總頻寬不得超過 400MHz。
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		

功能類型	頻率範圍	頻寬範圍	Polarization	Common Name (通用名稱)	備註
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		所有使用的頻率範圍都必須不重疊。
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antenna-downlink	7750 - 8500 MHz	50 - 400 MHz	LHCP		
antenna-downlink	2200 - 2290 MHz	高達 40 MHz	RHCP	S 頻帶下行	一次只能使用一個極化
antenna-downlink	2200 - 2290 MHz	高達 40 MHz	LHCP		
antenna-downlink	7750 - 8500 MHz	高達 40 MHz	RHCP	X 頻帶窄頻帶下行	一次只能使用一個極化
antenna-downlink	7750 - 8500 MHz	高達 40 MHz	LHCP		
antenna-uplink	2025 - 2110 MHz	高達 40 MHz	RHCP	S 頻帶上行連結	一次只能使用一個極化
antenna-uplink	2025 - 2110 MHz	高達 40 MHz	LHCP		EIRP 20-50 dBW
antenna-uplink-echo	2025 - 2110 MHz	2 MHz	RHCP	上行回應	符合天線上行限制
antenna-uplink-echo	2025 - 2110 MHz	2 MHz	LHCP		

功能類型	頻率範圍	頻寬範圍	Polarization	Common Name (通用名稱)	備註
antenna-downlink-demod-decode	7750 - 8500 MHz	高達 500 MHz	RHCP	X 頻帶解調和解碼下行	
antenna-downlink-demod-decode	7750 - 8500 MHz	高達 500 MHz	LHCP		
追蹤	N/A	N/A	N/A	N/A	支援自動追蹤和程式追蹤

* RHCP = 右手圓形極化，LHCP = 左手圓形極化。如需極化的詳細資訊，請參閱[環形極化](#)。

了解 AWS Ground Station 如何使用 ephemerides

[暫時性](#)、[複數暫時性](#)，是提供天文物件軌跡的檔案或資料結構。從歷史上來看，此檔案僅指表格式資料，但逐漸會導向至表示太空船軌跡的多種資料檔案。

Ephemeris API 允許將自訂暫時性樣板上傳到 AWS Ground Station，以便與衛星搭配使用。這些暫時性物質會覆寫來自 [Space-Track](#) 的預設暫時性物質（請參閱：[預設暫時性資料](#)）。我們支援以 Orbit Ephemeris Message (OEM)、雙行元素 (TLE) 和方位提升格式接收暫時性資料。

AWS Ground Station 使用 ephemeris 資料，根據 AWS Ground Station 網路中提供的 ephemeris 和正確命令天線來判斷聯絡人何時可用。根據預設，如果您的衛星有指派的 [NORAD ID](#)，則不需要採取任何動作來 AWS Ground Station 提供 ephemeride。

上傳自訂暫時化可以改善追蹤品質、處理沒有可用 [Space-Track](#) 暫時化的早期操作 AWS Ground Station，以及考慮操作。

或者，AWS Ground Station 支援方位提升格式，可讓您直接指定天線指向方向，而無需提供衛星軌道資訊。這適用於因為衛星軌跡資訊不精確或未知而需要精確天線指向的情況。

主題

- [預設暫時性資料](#)
- [提供自訂暫時性資料](#)
- [使用自訂 ephemeris 預留聯絡人](#)
- [了解使用哪些 ephemeris](#)
- [取得衛星的目前暫時性變化](#)
- [還原至預設 ephemeris 資料](#)

預設暫時性資料

根據預設，AWS Ground Station 會使用來自 [Space-Track](#) 的公開可用資料，而且不需要採取任何動作來提供 AWS Ground Station 這些預設的短暫性資料。這些暫時性元素是與衛星 [NORAD ID](#) 相關聯的 [雙行元素集 \(TLEs\)](#)。所有預設 ephemeride 的優先順序為 0。因此，透過 ephemeris API 上傳的任何未過期自訂 ephemeride 一律會予以覆寫，其優先順序必須為 1 或更高。

沒有 NORAD ID 的衛星必須將自訂暫時性資料上傳至其中 AWS Ground Station。例如，剛啟動或刻意從 [Space-Track](#) 目錄省略的衛星不會有 NORAD ID，而且需要上傳自訂短暫化。如需提供自訂暫時性資料的詳細資訊，請參閱：[提供自訂暫時性資料](#)。

提供自訂暫時性資料

Important

ephemeris API 目前處於預覽狀態

僅在需要時提供對 Ephemeris API 的存取。如果您需要上傳自訂暫時性資料的功能，請透過 開立 AWS 支援 票證 [AWS Support Center Console](#)。您的團隊將與您合作，為您的特定需求啟用此功能。

概觀

Ephemeris API 允許將自訂暫時性樣板上傳到 AWS Ground Station，以便與衛星搭配使用。這些暫時性物質會覆寫來自 [Space-Track](#) 的預設暫時性物質（請參閱：[預設暫時性資料](#)）。我們支援以 Orbit Ephemeris Message (OEM)、雙行元素 (TLE) 和方位提升格式接收暫時性資料。

AWS Ground Station 將 ephemerides 視為 [個人化用量資料](#)。如果您使用此選用功能，AWS 將使用 ephemeris 資料來提供故障診斷支援。

上傳自訂暫時化可以改善追蹤品質、處理沒有可用 [Space-Track](#) 暫時化項目的操作 AWS Ground Station，以及考慮操作。

若要對無效的暫時性事件進行故障診斷，請參閱：[故障診斷無效的短暫性事件](#)

範例：搭配 使用客戶提供的 ephemeride AWS Ground Station

如需搭配 使用客戶提供的暫時性資料的詳細指示 AWS Ground Station，請參閱 [搭配 使用客戶提供的暫時性資料 AWS Ground Station](#)，以及其相關聯的 GitHub 儲存庫 [aws-samples/aws-groundstation-cpe](#)。

提供 TLE ephemeris 資料

Important

ephemeris API 目前處於預覽狀態

僅在需要時提供對 Ephemeris API 的存取。如果您需要上傳自訂暫時性資料的功能，請透過 開立 AWS 支援 票證 [AWS Support Center Console](#)。您的團隊將與您合作，為您的特定需求啟用此功能。

概觀

雙行元素 (TLE) 集是描述衛星軌道的標準化格式。Ephemeris API 允許將 TLE 暫時性資料上傳至 AWS Ground Station，以便與衛星搭配使用。這些暫時性物質會覆寫來自 [Space-Track](#) 的預設暫時性物質 (請參閱：[預設暫時性資料](#))。

AWS Ground Station 將 ephemerides 視為 [個人化用量資料](#)。如果您使用此選用功能，AWS 將使用 ephemeris 資料來提供故障診斷支援。

上傳自訂 TLE 暫時化可以改善追蹤品質、處理沒有可用 [Space-Track](#) 暫時化的早期操作 AWS Ground Station，以及計算操作。

Note

在為衛星指派衛星目錄號碼之前提供自訂 ephemeris 時，您可以將 00000 用於 TLE 的衛星目錄號碼欄位，以及 000 TLE 國際指標欄位的啟動號碼部分 (例如 24000A 用於 2024 年啟動的車輛)。

如需 TLEs 格式的詳細資訊，請參閱 [雙行元素集](#)。

建立 TLE ephemeris

您可以使用 API 中的 [CreateEphemeris](#) 動作來建立 TLE ephemeris。AWS Ground Station 此動作將使用請求內文中的資料或從指定的 S3 儲存貯體上傳暫時性資料。

請務必注意，上傳 ephemeris 會將 ephemeris 設定為 `VALIDATING` 並啟動非同步工作流程，以驗證和產生來自 ephemeris 的潛在聯絡人。只有在 ephemeris 通過此工作流程後，`ENABLED` 才會將其用於聯絡人。您應該輪詢 [DescribeEphemeris](#) 的暫時性狀態，或使用 CloudWatch 事件來追蹤暫時性狀態變更。

若要對無效的暫時性事件進行故障診斷，請參閱：[故障診斷無效的短暫性事件](#)

範例：透過 API 建立雙行元素 (TLE) 集合暫時性

AWS SDKs 和 CLI，AWS Ground Station 透過 [CreateEphemeris](#) 呼叫將雙行元素 (TLE) 集合暫時性資料上傳至。此暫時性資料將用於取代衛星的預設暫時性資料 (請參閱 [預設暫時性資料](#))。此範例示範如何使用 [AWS 適用於 Python 的 SDK \(Boto3\)](#) 執行此操作。

TLE 集是 JSON 格式的物件，可將一或多個 TLEs 串連在一起，以建構連續軌跡。TLEs 集合中的 TLE 必須形成連續集合，我們可以用來建構軌跡 (即 TLE 集合中 TLEs 之間的時間差距)。範例 TLE 集如下所示：

```
[
  {
    "tleLine1": "1 25994U 99068A 20318.54719794 .00000075 00000-0 26688-4 0
9997",
    "tleLine2": "2 25994 98.2007 30.6589 0001234 89.2782 18.9934
14.57114995111906",
    "validTimeRange": {
      "startTime": 12345,
      "endTime": 12346
    }
  },
  {
    "tleLine1": "1 25994U 99068A 20318.54719794 .00000075 00000-0 26688-4 0
9997",
    "tleLine2": "2 25994 98.2007 30.6589 0001234 89.2782 18.9934
14.57114995111906",
    "validTimeRange": {
      "startTime": 12346,
      "endTime": 12347
    }
  }
]
```

Note

TLEs 集中 TLE 的時間範圍必須完全相符，才能成為有效的連續軌跡。

您可以透過 boto3 AWS Ground Station 用戶端上傳 TLE 集，如下所示：

```
import boto3
from datetime import datetime, timedelta, timezone

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Create TLE ephemeris
tle_ephemeris = ground_station_client.create_ephemeris(
    name="Example Ephemeris",
    satelliteId="2e925701-9485-4644-b031-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=3),
```

```

    priority=2,
    ephemeris={
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A   20318.54719794   .000000075   00000-0
26688-4 0 9997",
                    "tleLine2": "2 25994   98.2007   30.6589 0001234   89.2782   18.9934
14.571149951111906",
                    "validTimeRange": {
                        "startTime": datetime.now(timezone.utc),
                        "endTime": datetime.now(timezone.utc) + timedelta(days=7),
                    },
                }
            ]
        }
    },
)

print(f"Created TLE ephemeris with ID: {tle_ephemeris['ephemerisId']}")

```

此呼叫將傳回 `ephemerisId`，可用於在未來參考 ephemeris。例如，我們可以使用上述呼叫中提供的 `ephemerisId` 來查詢 ephemeris 狀態：

```

import boto3
from datetime import datetime, timedelta, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# First, create a TLE ephemeris
print("Creating TLE ephemeris...")

tle_ephemeris = ground_station_client.create_ephemeris(
    name="Example TLE Ephemeris for Description",
    satelliteId="2e925701-9485-4644-b031-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=3),
    priority=2,
    ephemeris={
        "tle": {
            "tleData": [

```

```

        {
            "tleLine1": "1 25994U 99068A   20318.54719794   .00000075   00000-0
26688-4 0 9997",
            "tleLine2": "2 25994   98.2007   30.6589 0001234   89.2782   18.9934
14.57114995111906",
            "validTimeRange": {
                "startTime": datetime.now(timezone.utc),
                "endTime": datetime.now(timezone.utc) + timedelta(days=7),
            },
        }
    ]
}
},
)

ephemeris_id = tle_ephemeris["ephemerisId"]
print(f"Created TLE ephemeris with ID: {ephemeris_id}")

# Describe the ephemeris immediately to check initial status
print("Describing ephemeris...")

response = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)

print(f"Ephemeris ID: {response['ephemerisId']}")
print(f"Name: {response['name']}")
print(f"Status: {response['status']}")

```

以下提供 [DescribeEphemeris](#) 動作的範例回應

```

{
  "creationTime": 1620254718.765,
  "enabled": true,
  "name": "Example Ephemeris",
  "ephemerisId": "fde41049-14f7-413e-bd7b-EXAMPLE01",
  "priority": 2,
  "status": "VALIDATING",
  "suppliedData": {
    "tle": {
      "ephemerisData": "[{\"tleLine1\": \"1 25994U 99068A   20318.54719794   .00000075
00000-0 26688-4 0 9997\", \"tleLine2\": \"2 25994   98.2007   30.6589 0001234   89.2782
18.9934 14.57114995111906\", \"validTimeRange\": {\"startTime\": 1620254712000,
\"endTime\": 1620859512000}}]"
    }
  }
}

```

```
}  
}
```

建議您輪詢 [DescribeEphemeris](#) 路由或使用 CloudWatch 事件來追蹤上傳的暫時性事件的狀態，因為它必須先經過非同步驗證工作流程，才能設定為 ENABLED 並可用於排程和執行聯絡人。

請注意，在上述 25994 範例中，TLE 集中所有 TLEs 中的 NORAD ID 必須符合您的衛星在 [Space-Track](#) 資料庫中指派的 NORAD ID。

範例：從 S3 儲存貯體上傳 TLE ephemeris 資料

您也可以透過指向儲存貯體和物件金鑰，直接從 S3 儲存貯體上傳 TLE ephemeris 檔案。AWS Ground Station 將代表您擷取物件。有關 中靜態資料加密的詳細資訊 AWS Ground Station，請參閱：[的靜態資料加密 AWS Ground Station](#)。

以下是從 S3 儲存貯體上傳 TLE ephemeris 檔案的範例

```
import boto3  
from datetime import datetime, timedelta, timezone  
import json  
  
# Create AWS clients  
s3_client = boto3.client("s3")  
ground_station_client = boto3.client("groundstation")  
  
# Define S3 bucket and key  
bucket_name = "ephemeris-bucket"  
object_key = "test_data.tle"  
  
# Create sample TLE set data  
# Note: For actual satellites, use real TLE data from sources like Space-Track  
tle_set_data = [  
    {  
        "tleLine1": "1 25994U 99068A 20318.54719794 .00000075 00000-0 26688-4 0  
9997",  
        "tleLine2": "2 25994 98.2007 30.6589 0001234 89.2782 18.9934  
14.57114995111906",  
        "validTimeRange": {  
            "startTime": datetime.now(timezone.utc),  
            "endTime": datetime.now(timezone.utc) + timedelta(days=3),  
        },  
    },  
    {
```

```

        "tleLine1": "1 25994U 99068A 20321.54719794 .00000075 00000-0 26688-4 0
9998",
        "tleLine2": "2 25994 98.2007 33.6589 0001234 89.2782 18.9934
14.57114995112342",
        "validTimeRange": {
            "startTime": datetime.now(timezone.utc) + timedelta(days=3),
            "endTime": datetime.now(timezone.utc) + timedelta(days=7),
        },
    },
]

# Convert to JSON string for upload
tle_json = json.dumps(tle_set_data, indent=2)

# Upload sample TLE data to S3
print(f"Uploading TLE set data to s3://{bucket_name}/{object_key}")

s3_client.put_object(
    Bucket=bucket_name, Key=object_key, Body=tle_json, ContentType="application/json"
)
print("TLE set data uploaded successfully to S3")
print(f"Uploaded {len(tle_set_data)} TLE entries covering 7 days")

# Create TLE ephemeris from S3
print("Creating TLE ephemeris from S3...")

s3_tle_ephemeris = ground_station_client.create_ephemeris(
    name="2022-11-05 S3 TLE Upload",
    satelliteId="fde41049-14f7-413e-bd7b-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=5),
    priority=2,
    ephemeris={"tle": {"s3object": {"bucket": bucket_name, "key": object_key}}},
)

print(f"Created TLE ephemeris with ID: {s3_tle_ephemeris['ephemerisId']}")

```

提供 OEM ephemeris 資料

Important

ephemeris API 目前處於預覽狀態

僅在需要時提供對 Ephemeris API 的存取。如果您需要上傳自訂暫時性資料的功能，請透過 開立 AWS 支援 票證 [AWS Support Center Console](#)。您的團隊將與您合作，為您的特定需求啟用此功能。

概觀

Orbit Ephemeris Message (OEM) 是一種標準化格式，用於表示太空船軌跡資料。Ephemeris API 允許將 OEM 暫時化上傳到 AWS Ground Station，以便與衛星搭配使用。這些暫時性物質會覆寫來自 [Space-Track](#) 的預設暫時性物質（請參閱：[預設暫時性資料](#)）。

AWS Ground Station 將 ephemerides 視為 [個人化用量資料](#)。如果您使用此選用功能，AWS 將使用 ephemeris 資料來提供故障診斷支援。

上傳自訂 OEM 暫時化可以改善追蹤品質、處理沒有可用 [Space-Track](#) 暫時化的早期操作 AWS Ground Station，以及計算操作。

Note

在為衛星指派衛星目錄號碼之前提供自訂 ephemeris 時，您可以將 satelliteId 用於 OEM OBJECT_ID 的部分。

如需 OEMs 格式的詳細資訊，請參閱 [OEM ephemeris 格式](#)。

OEM ephemeris 格式

AWS Ground Station 根據 [CCSDS 標準](#) 處理 OEM 客戶提供的 Ephemerides，但有一些額外的限制。OEM 檔案應為 KVN 格式。下表概述 OEM 中的不同欄位，以及與 CCSDS 標準 AWS Ground Station 的差異。

區段	欄位	需要 CCSDS	AWS Ground Station 必要	備註
標頭	CCSDS_OEM_VERS	是	是	必要值：2.0
	COMMENT	否	否	
	分類	否	否	
	CREATION_DATE	是	是	

區段	欄位	需要 CCSDS	AWS Ground Station 必要	備註
中繼資料	原始伺服器	是	是	
	MESSAGE_ID	否	否	
	META_START	是	是	
	COMMENT	否	否	
	OBJECT_NAME	是	是	
	OBJECT_ID	是	是	
	CENTER_NAME	是	是	必要值：地球
	REF_FRAME	是	是	接受的 值：EME2 000、ITRF2000
	REF_FRAME _EPOCH	否	不支援*	不需要，因 為接受的 REF_FRAMEs 具有隱含 epoch
	TIME_SYSTEM	是	是	必要值：UTC
	START_TIME	是	是	
	USEABLE_S TART_TIME	否	否	
	USEABLE_S TOP_TIME	否	否	
	STOP_TIME	是	是	

區段	欄位	需要 CCSDS	AWS Ground Station 必要	備註
	插補	否	是	必要，因此 AWS Ground Station 可以為聯絡人產生準確的指向角度。
	INTERPOLATION_DEGREE	否	是	必要，因此 AWS Ground Station 可以為聯絡人產生準確的指向角度。如果可能，將使用指定的度數，但如果區段中沒有足夠的資料，則會使用較低的度數。
	META_STOP	是	是	
資料	X	是	是	在中代表 km
	Y	是	是	在中代表 km
	Z	是	是	在中代表 km
	X_DOT	是	是	在中代表 km/s
	Y_DOT	是	是	在中代表 km/s
	Z_DOT	是	是	在中代表 km/s
	X_DDOT	否	否	在中代表 km/s ²
	Y_DDOT	否	否	在中代表 km/s ²

區段	欄位	需要 CCSDS	AWS Ground Station 必要	備註
	Z_DDOT	否	否	在中代表 km/s ²
共變數矩陣	轉移_開始	否	否	
	EPOCH	否	否	
	DISC_REF_FRAME	否	否	
	樞紐分析_停止	否	否	

* 如果提供 OEM 中 AWS Ground Station 包含不支援的任何資料列，OEM 將失敗驗證。

與的 CCSDS 標準的重要偏差 AWS Ground Station 為：

- CCSDS_OEM_VERS 必須是 2.0。
- REF_FRAME 必須是 EME2000 或 ITRF2000。
- REF_FRAME_EPOCH 不支援 AWS Ground Station。
- CENTER_NAME 必須是 Earth。
- TIME_SYSTEM 必須是 UTC。
- INTERPOLATION 和 INTERPOLATION_DEGREE 都是 AWS Ground Station 客戶提供的 ephemeris 的必要項目。
- AWS Ground Station 允許不包含足夠暫時性資料記錄的 OEM 資料區塊在指定的執行插補，以偏離 CCSDS 5.2.4 INTERPOLATION_DEGREE.7。在此情況下，AWS Ground Station 將使用小於或等於指定的最高插補程度 INTERPOLATION_DEGREE。

KVN 格式的範例 OEM ephemeris

以下是 JPSS-1 公有廣播者衛星的 KVN 格式 OEM ephemeris 截斷範例。

```
CCSDS_OEM_VERS = 2.0
```

```
COMMENT Orbit data are consistent with planetary ephemeris DE-430
```

```

CREATION_DATE = 2024-07-22T05:20:59
ORIGINATOR    = Raytheon-JPSS/CGS

```

```

META_START
OBJECT_NAME      = J1
OBJECT_ID        = 2017-073A
CENTER_NAME      = Earth
REF_FRAME        = EME2000
TIME_SYSTEM      = UTC
START_TIME       = 2024-07-22T00:00:00.000000
STOP_TIME        = 2024-07-22T00:06:00.000000
INTERPOLATION    = Lagrange
INTERPOLATION_DEGREE = 5
META_STOP

```

```

2024-07-22T00:00:00.000000  5.905147360000000e+02  -1.860082793999999e+03
-6.944807075000000e+03  -5.784245796000000e+00  4.347501391999999e+00
-1.657256863000000e+00
2024-07-22T00:01:00.000000  2.425572045154201e+02  -1.595860765983339e+03
-7.030938457373539e+03  -5.810660250794190e+00  4.457103652219009e+00
-1.212889340333023e+00
2024-07-22T00:02:00.000000  -1.063224256538050e+02  -1.325569732497146e+03
-7.090262617183503e+03  -5.814973972202444e+00  4.549739160042560e+00
-7.639633689161465e-01
2024-07-22T00:03:00.000000  -4.547973959231161e+02  -1.050238305712201e+03
-7.122556683227951e+03  -5.797176562437553e+00  4.625064829516728e+00
-3.121687831090774e-01
2024-07-22T00:04:00.000000  -8.015427368657785e+02  -7.709137891269565e+02
-7.127699477194810e+03  -5.757338007808417e+00  4.682800822515077e+00
1.407953645161997e-01
2024-07-22T00:05:00.000000  -1.145240083085062e+03  -4.886583601179489e+02
-7.105671911254255e+03  -5.695608435738609e+00  4.722731329786999e+00
5.932259682105052e-01
2024-07-22T00:06:00.000000  -1.484582479061495e+03  -2.045451985605701e+02
-7.056557069672793e+03  -5.612218005854990e+00  4.744705579872771e+00
1.043421397392599e+00

```

建立 OEM ephemeris

OEM ephemeris 可以使用 API 中的 AWS Ground Station [CreateEphemeris](#) 動作建立。此動作將使用請求內文中的資料或從指定的 S3 儲存貯體上傳暫時性資料。

請務必注意，上傳 ephemeris 會將 ephemeris 設定為 `VALIDATING` 並啟動非同步工作流程，以驗證和產生來自 ephemeris 的潛在聯絡人。只有在 ephemeris 通過此工作流程後，`ENABLED` 才會將其用於聯絡人。您應該輪詢 [DescribeEphemeris](#) 的暫時性狀態，或使用 CloudWatch 事件來追蹤暫時性狀態變更。

若要對無效的暫時性事件進行故障診斷，請參閱：[故障診斷無效的短暫性事件](#)

範例：從 S3 儲存貯體上傳 OEM ephemeris 資料

您也可以透過指向儲存貯體和物件金鑰，直接從 S3 儲存貯體上傳 OEM ephemeris 檔案。AWS Ground Station 將代表您擷取物件。有關 中靜態資料加密的詳細資訊 AWS Ground Station ，請參閱：[的靜態資料加密 AWS Ground Station](#)。

以下是從 S3 儲存貯體上傳 OEM ephemeris 檔案的範例

```
import boto3
from datetime import datetime, timedelta, timezone

# Create AWS clients
s3_client = boto3.client("s3")
ground_station_client = boto3.client("groundstation")

# Define S3 bucket and key
bucket_name = "ephemeris-bucket"
object_key = "test_data.oem"

# Create sample OEM data in KVN format
oem_data = """CCSDS_OEM_VERS = 2.0

COMMENT Orbit data are consistent with planetary ephemeris DE-430

CREATION_DATE   = 2024-07-22T05:20:59
ORIGINATOR      = Raytheon-JPSS/CGS

META_START
OBJECT_NAME     = J1
OBJECT_ID       = 2017-073A
CENTER_NAME     = Earth
REF_FRAME       = EME2000
TIME_SYSTEM     = UTC
START_TIME      = 2024-07-22T00:00:00.000000
STOP_TIME       = 2024-07-22T00:06:00.000000
INTERPOLATION   = Lagrange
```

```

INTERPOLATION_DEGREE = 5
META_STOP

2024-07-22T00:00:00.000000  5.905147360000000e+02  -1.860082793999999e+03
-6.944807075000000e+03  -5.784245796000000e+00  4.347501391999999e+00
-1.657256863000000e+00
2024-07-22T00:01:00.000000  2.425572045154201e+02  -1.595860765983339e+03
-7.030938457373539e+03  -5.810660250794190e+00  4.457103652219009e+00
-1.212889340333023e+00
2024-07-22T00:02:00.000000  -1.063224256538050e+02  -1.325569732497146e+03
-7.090262617183503e+03  -5.814973972202444e+00  4.549739160042560e+00
-7.639633689161465e-01
2024-07-22T00:03:00.000000  -4.547973959231161e+02  -1.050238305712201e+03
-7.122556683227951e+03  -5.797176562437553e+00  4.625064829516728e+00
-3.121687831090774e-01
2024-07-22T00:04:00.000000  -8.015427368657785e+02  -7.709137891269565e+02
-7.127699477194810e+03  -5.757338007808417e+00  4.682800822515077e+00
1.407953645161997e-01
2024-07-22T00:05:00.000000  -1.145240083085062e+03  -4.886583601179489e+02
-7.105671911254255e+03  -5.695608435738609e+00  4.722731329786999e+00
5.932259682105052e-01
2024-07-22T00:06:00.000000  -1.484582479061495e+03  -2.045451985605701e+02
-7.056557069672793e+03  -5.612218005854990e+00  4.744705579872771e+00
1.043421397392599e+00
""""

# Upload sample OEM data to S3
print(f"Uploading OEM data to s3://{bucket_name}/{object_key}")

s3_client.put_object(
    Bucket=bucket_name, Key=object_key, Body=oem_data, ContentType="text/plain"
)

print("OEM data uploaded successfully to S3")

# Create OEM ephemeris from S3
print("Creating OEM ephemeris from S3...")

s3_oem_ephemeris = ground_station_client.create_ephemeris(
    name="2024-07-22 S3 OEM Upload",
    satelliteId="fde41049-14f7-413e-bd7b-EXAMPLE01",
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=5),
    priority=2,

```

```
    ephemeris={"oem": {"s3object": {"bucket": bucket_name, "key": object_key}}},
)

print(f"Created OEM ephemeris with ID: {s3_oem_ephemeris['ephemerisId']}")
```

以下是從 [DescribeEphemeris](#) 動作傳回的資料範例，該動作會針對上一個範例程式碼區塊中上傳的 OEM ephemeris 呼叫。

```
{
  "creationTime": 1620254718.765,
  "enabled": true,
  "name": "Example Ephemeris",
  "ephemerisId": "fde41049-14f7-413e-bd7b-EXAMPLE02",
  "priority": 2,
  "status": "VALIDATING",
  "suppliedData": {
    "oem": {
      "sourceS3object": {
        "bucket": "ephemeris-bucket-for-testing",
        "key": "test_data.oem"
      }
    }
  }
}
```

提供方位提升暫時性資料

Important

方位提升暫時性特徵目前處於預覽狀態，需要明確加入。

對於有限數量的預先確定、專門的使用案例，方位提升暫時性功能受到嚴格的存取控制。存取比標準客戶提供的暫時性功能更嚴格。如需已核准使用案例和存取請求程序的詳細資訊，請透過 [開立 AWS 支援 票證](#) [AWS Support Center Console](#)。您的團隊將引導您完成特殊使用案例的核准程序。

概觀

方位角抬高暫時性提供一個直接指定天線指向方向的方法，而無需提供衛星軌道資訊。您不用上傳描述衛星軌道的暫時性資料，而是提供標記時間的方位角和高度角度，告訴天線在整個聯絡中的位置。

AWS Ground Station 將 ephemerides 視為 [個人化用量資料](#)。如果您使用此選用功能，AWS 將使用 ephemeris 資料來提供故障診斷支援。

此方法在下列案例中特別有用：

- 早期操作支援：在啟動和早期軌道階段 (LEOP) 期間，當無法使用精確的軌道資料，或軌道參數快速變更時。
- 自訂指標模式：實作特定指標序列進行天線測試或非標準操作。

Note

使用方位提升暫時性時，可能會從聯絡人保留請求中省略衛星 ARN。如果未省略衛星 ARN，它仍會包含在聯絡資料中，但方位角抬高暫時性，將用於天線指向，而不是執行暫時性優先順序解析。方位角海拔高度與特定地面站相關聯，並定義該位置的天線指向方向。

方位海拔暫時性資料格式

方位海拔暫時性資料由時間標記方位和海拔值組成，組織成區段。每個區段都包含涵蓋特定時間範圍的一系列方位角和高度角。

方位海拔暫時性資料的關鍵元件為：

- Ground Station：將使用此方位角海拔面積的特定地面工作站。
- 角度單位：角度的測量單位 (DEGREE_ANGLE 或 RADIAN)。
- 區段：一個或多個時間限制的方位角和高度角度集合。
- 時間標記角度：具有相關聯時間戳記的個別方位和高度值。

每個區段都需要：

- 參考 epoch（區段的基本時間）
- 有效的時間範圍（區段的開始和結束時間）
- 至少 5 個時間標記方位/上升對

方位提升限制條件：

- 方位度：-180° 到 360°

- 弧度的方位： $-\pi$ 到 2π
- 上升度數： -90° 到 90°
- 弧度的上升： $-\pi/2$ 到 $\pi/2$
- 時間值必須在每個區段內以遞增順序排列
- 區段不得在時間上重疊

如需詳細資訊，請參閱 [CreateEphemeris](#) API 文件和 [TimeAzEl](#) 資料類型。

建立方位提升暫時性

方位浮水印是使用相同的 [CreateEphemeris](#) API 動作建立，但使用 `ephemeris azEl` 類型。與 TLE 和 OEM 暫時性差異的主要差異為：

- 您必須指定 `groundStation` 參數
- 必須從請求中省略 `satelliteId` 參數
- 優先順序設定不適用（每個方位方位海拔暫時性都專屬於地面站）
- 每個區段必須至少包含 5 個方位/上升點，以支援第 4 個順序的延遲插補
- [CreateEphemeris](#) API 文件中詳細說明了其他限制和要求

請務必注意，上傳 `ephemeris` 會將 `ephemeris` 設定為 `VALIDATING` 並啟動非同步工作流程，以驗證和產生來自 `ephemeris` 的潛在聯絡人。只有在聯絡人通過此工作流程且其狀態變為 `ENABLED` 後，才會使用暫時性聯絡人 `ENABLED`。您應該輪詢 [DescribeEphemeris](#) 的暫時性狀態，或使用 `CloudWatch` 事件來追蹤暫時性狀態變更。

若要對無效的暫時性事件進行故障診斷，請參閱：[故障診斷無效的短暫性事件](#)

範例：透過 API 建立方位角海拔面積

下列範例示範如何使用適用於 Python 的 AWS SDK (Boto3) 建立方位方位海拔凹凸：

```
import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Create azimuth elevation ephemeris
```

```

azimuth_elevation_ephemeris = ground_station_client.create_ephemeris(
    name="Azimuth Elevation for Ohio Ground Station",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": "2024-03-15T10:00:00Z",
                            "validTimeRange": {
                                "startTime": "2024-03-15T10:00:00Z",
                                "endTime": "2024-03-15T10:15:00Z",
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 45.0, "el": 10.0},
                                {"dt": 180.0, "az": 50.0, "el": 15.0},
                                {"dt": 360.0, "az": 55.0, "el": 20.0},
                                {"dt": 540.0, "az": 60.0, "el": 25.0},
                                {"dt": 720.0, "az": 65.0, "el": 30.0},
                                {"dt": 900.0, "az": 70.0, "el": 35.0},
                            ],
                        },
                    ],
                },
            },
        },
    },
)

print(f"Created ephemeris with ID: {azimuth_elevation_ephemeris['ephemerisId']}")

```

在此範例中：

- 方位海拔資料與「俄亥俄 1」地面站相關聯
- 角度以度為單位指定
- 此區段涵蓋 15 分鐘的期間
- 這些dt值是與參考 epoch 的原子秒位移
- 提供六個方位/上升對（最小值為 5）

範例：從 S3 上傳方位提升資料

對於較大的資料集，您可以從 S3 儲存貯體上傳方位提升資料：

```
import boto3
import json

# Create AWS clients
s3_client = boto3.client("s3")
ground_station_client = boto3.client("groundstation")

# Define S3 bucket and key
bucket_name = "azimuth-elevation-bucket"
object_key = "singapore-azimuth-elevation.json"

# Create sample azimuth elevation data
azimuth_elevation_data = {
    "angleUnit": "DEGREE_ANGLE",
    "azElSegmentList": [
        {
            "referenceEpoch": "2024-03-15T10:00:00Z",
            "validTimeRange": {
                "startTime": "2024-03-15T10:00:00Z",
                "endTime": "2024-03-15T10:15:00Z",
            },
            "azElList": [
                {"dt": 0.0, "az": 45.0, "el": 10.0},
                {"dt": 180.0, "az": 50.0, "el": 15.0},
                {"dt": 360.0, "az": 55.0, "el": 20.0},
                {"dt": 540.0, "az": 60.0, "el": 25.0},
                {"dt": 720.0, "az": 65.0, "el": 30.0},
                {"dt": 900.0, "az": 70.0, "el": 35.0},
            ],
        },
        {
            "referenceEpoch": "2024-03-15T10:15:00Z",
            "validTimeRange": {
                "startTime": "2024-03-15T10:15:00Z",
                "endTime": "2024-03-15T10:30:00Z",
            },
            "azElList": [
                {"dt": 0.0, "az": 70.0, "el": 35.0},
                {"dt": 180.0, "az": 75.0, "el": 40.0},
                {"dt": 360.0, "az": 80.0, "el": 45.0},
            ],
        },
    ],
}
```

```

        {"dt": 540.0, "az": 85.0, "el": 50.0},
        {"dt": 720.0, "az": 90.0, "el": 55.0},
        {"dt": 900.0, "az": 95.0, "el": 50.0},
    ],
},
],
}

# Upload sample data to S3
print(f"Uploading azimuth elevation data to s3://{bucket_name}/{object_key}")

s3_client.put_object(
    Bucket=bucket_name,
    Key=object_key,
    Body=json.dumps(azimuth_elevation_data, indent=2),
    ContentType="application/json",
)
print("Sample data uploaded successfully to S3")

# Create azimuth elevation ephemeris from S3
print("Creating azimuth elevation ephemeris from S3...")

s3_azimuth_elevation_ephemeris = ground_station_client.create_ephemeris(
    name="Large Azimuth Elevation Dataset",
    ephemeris={
        "azEl": {
            "groundStation": "Singapore 1",
            "data": {"s3object": {"bucket": bucket_name, "key": object_key}},
        }
    },
)

print(f"Created ephemeris with ID: {s3_azimuth_elevation_ephemeris['ephemerisId']}")

```

S3 物件應包含 JSON 結構，其中包含與直接上傳範例中所示相同格式的方位角提升資料。

保留具有方位海拔暫時性的聯絡人

使用方位角海拔暫時性符號來保留聯絡人時，程序會與 TLE 和 OEM 暫時性符號不同：

1. 使用 [CreateEphemeris](#) 建立方位角海拔面積
2. 等待 ephemeris 達到ENABLED狀態
3. 使用 [ReserveContact](#) 搭配追蹤覆寫來預留聯絡人

預留與方位海拔暫時性接觸的範例：

```
import boto3
from datetime import datetime
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# First, create an azimuth elevation ephemeris
print("Creating azimuth elevation ephemeris...")

create_ephemeris_response = ground_station_client.create_ephemeris(
    name="Azimuth Elevation for Contact Reservation",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": "2024-03-15T10:00:00Z",
                            "validTimeRange": {
                                "startTime": "2024-03-15T10:00:00Z",
                                "endTime": "2024-03-15T10:15:00Z",
                            },
                        },
                    ],
                    "azElList": [
                        {"dt": 0.0, "az": 45.0, "el": 10.0},
                        {"dt": 180.0, "az": 50.0, "el": 15.0},
                        {"dt": 360.0, "az": 55.0, "el": 20.0},
                        {"dt": 540.0, "az": 60.0, "el": 25.0},
                        {"dt": 720.0, "az": 65.0, "el": 30.0},
                        {"dt": 900.0, "az": 70.0, "el": 35.0},
                    ],
                },
            },
        },
    },
)

ephemeris_id = create_ephemeris_response["ephemerisId"]
```

```
print(f"Created ephemeris with ID: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
print("Waiting for ephemeris to become ENABLED...")

while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# Reserve contact with azimuth elevation ephemeris
print("Reserving contact...")

contact = ground_station_client.reserve_contact(
    # Note: satelliteArn is omitted when using azimuth elevation ephemeris
    missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-profile/
example-mission-profile",
    groundStation="Ohio 1",
    startTime=datetime(2024, 3, 15, 10, 0, 0),
    endTime=datetime(2024, 3, 15, 10, 15, 0),
    trackingOverrides={"programTrackSettings": {"azEl": {"ephemerisId":
ephemeris_id}}},
)

print(f"Reserved contact with ID: {contact['contactId']}")
```

Note

保留與方位海拔暫時性的聯絡人時，可能會省略 `satelliteArn` 參數。天線會在聯絡期間遵循指定的方位角和高度角度。

列出可用的聯絡人

使用方位提升暫時性時，[ListContacts](#) API 需要特定參數：

- 請求中可能會省略 `satelliteArn` 參數
- 您必須提供具有方位角海拔 ephemeris ID 的 `ephemeris` 參數，以指定要使用的 ephemeris
- 可用的聯絡時段會顯示提供的方位角和高度角度何時高於所請求地面站的 [站點遮罩](#)
- 您仍然必須提供 `groundStation` 和 `missionProfileArn`

建立方位角升高暫時性並列出可用聯絡人的範例：

```
import boto3
from datetime import datetime, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Step 1: Create azimuth elevation ephemeris
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Stockholm AzEl Ephemeris",
    ephemeris={
        "azEl": {
            "groundStation": "Stockholm 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": "2024-04-01T12:00:00Z",
                            "validTimeRange": {
                                "startTime": "2024-04-01T12:00:00Z",
                                "endTime": "2024-04-01T12:30:00Z",
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 30.0, "el": 15.0},
                                {"dt": 360.0, "az": 45.0, "el": 30.0},
                                {"dt": 720.0, "az": 60.0, "el": 45.0},
                                {"dt": 1080.0, "az": 75.0, "el": 35.0},
                                {"dt": 1440.0, "az": 90.0, "el": 20.0},
                                {"dt": 1800.0, "az": 105.0, "el": 10.0},
                            ],
                        },
                    ],
                },
            },
        },
    ],
),
```

```
        }
    },
}
),
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Step 2: Wait for ephemeris to become ENABLED
print("Waiting for ephemeris to become ENABLED...")
while True:
    describe_response = ground_station_client.describe_ephemeris(
        ephemerisId=ephemeris_id
    )
    status = describe_response["status"]

    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        # Check for validation errors
        if "invalidReason" in describe_response:
            print(f"Ephemeris validation failed: {describe_response['invalidReason']}")
            raise RuntimeError(f"Ephemeris failed with status: {status}")

    print(f"Current status: {status}, waiting...")
    time.sleep(5)

# Step 3: List available contacts using the azimuth elevation ephemeris
print("Listing available contacts with azimuth elevation ephemeris...")

# Convert epoch timestamps to datetime objects
start_time = datetime.fromtimestamp(1760710513, tz=timezone.utc)
end_time = datetime.fromtimestamp(1760883313, tz=timezone.utc)

contacts_response = ground_station_client.list_contacts(
    startTime=start_time,
    endTime=end_time,
    groundStation="Stockholm 1",
    statusList=["AVAILABLE"],
    ephemeris={"azEl": {"id": ephemeris_id}},
    # satelliteArn is optional
```

```
satelliteArn="arn:aws:groundstation::111122223333:satellite/a88611b0-f755-404e-
b60d-57d8aEXAMPLE",
  missionProfileArn="arn:aws:groundstation:eu-north-1:111122223333:mission-
profile/966b72f6-6d82-4e7e-b072-f8240EXAMPLE",
)

# Process the results
if contacts_response["contactList"]:
    print(f"Found {len(contacts_response['contactList'])} available contacts:")
    for contact in contacts_response["contactList"]:
        print(f"  - Contact from {contact['startTime']} to {contact['endTime']}")
        print(
            f"    Max elevation: {contact.get('maximumElevation', {}).get('value', 'N/
A')}}°"
        )
    else:
        print("No available contacts found for the specified azimuth elevation ephemeris")
```

Note

列出聯絡人時，必須提供具有方位提升 ID 的 ephemeris 參數，以指定應使用哪個方位提升暫時性來判斷聯絡時段。如果 satelliteArn 包含 `azimuthElevationEphemerisId`，則會與聯絡人資料建立關聯，但方位海拔暫時性符號將用於天線指向，而不是執行暫時性優先順序解析。

使用自訂 ephemeris 預留聯絡人

概觀

使用自訂 ephemeris (TLE、OEM 或方位提升) 時，您可以使用 [ReserveContact](#) API 保留聯絡人。本節說明兩種保留聯絡人的常見工作流程，以及確保成功安排聯絡人的重要考量事項。

AWS Ground Station 天線是多個客戶的共用資源。這表示即使當您列出聯絡人時出現聯絡時段，其他客戶仍可能會在您之前保留該時段。因此，請務必確認您的聯絡人在保留後達到 SCHEDULED 狀態，並針對聯絡狀態變更實作適當的監控。

⚠ Important

對於方位提升暫時性，可能會從ReserveContact請求中省略 satelliteArn 參數，而且您必須trackingOverrides提供暫時性 ID。對於 TLE 和 OEM ephemeris，您仍然需要提供 satelliteArn。

聯絡預留工作流程

使用自訂 ephemeris 保留聯絡人有兩個主要工作流程：

1. List-then-reserve 工作流程：先使用 [ListContacts](#) 列出可用的聯絡時段，然後選取並保留特定時段。當您想要在選擇之前查看所有可用的機會時，此方法非常有用。
2. 直接保留工作流程：直接保留特定時段的聯絡人，而不先列出可用的聯絡人。當您已經知道所需的聯絡時間或正在使用預定排程時，此方法非常有用。

這兩個工作流程都是有效的，選擇取決於您的操作需求。下列各節提供每種方法的範例。

工作流程 1：列出可用的聯絡人，然後保留

此工作流程會先查詢可用的聯絡時段，然後保留特定的時段。當您想要在選擇之前查看所有可用的機會時，這非常有用。

範例：列出並預留方位角海拔高度

```
import boto3
from datetime import datetime, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# Create azimuth elevation ephemeris
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="AzEl Ephemeris for Contact",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
```

```

        "azElData": {
            "angleUnit": "DEGREE_ANGLE",
            "azElSegmentList": [
                {
                    "referenceEpoch": "2024-03-15T10:00:00Z",
                    "validTimeRange": {
                        "startTime": "2024-03-15T10:00:00Z",
                        "endTime": "2024-03-15T10:15:00Z",
                    },
                    "azElList": [
                        {"dt": 0.0, "az": 45.0, "el": 10.0},
                        {"dt": 180.0, "az": 50.0, "el": 15.0},
                        {"dt": 360.0, "az": 55.0, "el": 20.0},
                        {"dt": 540.0, "az": 60.0, "el": 25.0},
                        {"dt": 720.0, "az": 65.0, "el": 30.0},
                        {"dt": 900.0, "az": 70.0, "el": 35.0},
                    ],
                },
            ],
        },
    ],
},
),
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# List available contacts
print("Listing available contacts...")
contacts = ground_station_client.list_contacts(
    # Note: satelliteArn is omitted for azimuth elevation ephemeris

```

```

    groundStation="Ohio 1",
    missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-profile/
example-profile",
    startTime=datetime(2024, 3, 15, 10, 0, 0, tzinfo=timezone.utc),
    endTime=datetime(2024, 3, 15, 10, 15, 0, tzinfo=timezone.utc),
    statusList=["AVAILABLE"],
    ephemeris={"azEl": {"id": ephemeris_id}},
)

if contacts["contactList"]:
    # Reserve the first available contact
    contact = contacts["contactList"][0]
    print(f"Reserving contact from {contact['startTime']} to {contact['endTime']}...")

    reservation = ground_station_client.reserve_contact(
        # Note: satelliteArn is omitted when using azimuth elevation ephemeris
        missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-
profile/example-profile",
        groundStation="Ohio 1",
        startTime=contact["startTime"],
        endTime=contact["endTime"],
        trackingOverrides={
            "programTrackSettings": {"azEl": {"ephemerisId": ephemeris_id}}
        },
    )

    print(f"Reserved contact: {reservation['contactId']}")
else:
    print("No available contacts found")

```

範例：使用 TLE ephemeris 列出和保留

```

import boto3
from datetime import datetime, timedelta, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

satellite_id = "12345678-1234-1234-1234-123456789012"
satellite_arn = f"arn:aws:groundstation::111122223333:satellite/{satellite_id}"

# Create TLE ephemeris

```

```

print("Creating TLE ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="TLE Ephemeris for Contact",
    satelliteId=satellite_id,
    enabled=True,
    expirationTime=datetime.now(timezone.utc) + timedelta(days=7),
    priority=1, # Higher priority than default ephemeris
    ephemeris={
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A 24075.54719794 .00000075 00000-0
26688-4 0 9997",
                    "tleLine2": "2 25994 98.2007 30.6589 0001234 89.2782 18.9934
14.57114995111906",
                    "validTimeRange": {
                        "startTime": datetime.now(timezone.utc),
                        "endTime": datetime.now(timezone.utc) + timedelta(days=7),
                    },
                }
            ]
        }
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# List available contacts
print("Listing available contacts...")
start_time = datetime.now(timezone.utc) + timedelta(hours=1)
end_time = start_time + timedelta(days=1)

```

```
contacts = ground_station_client.list_contacts(
    satelliteArn=satellite_arn, # Required for TLE/OEM ephemeris
    groundStation="Hawaii 1",
    missionProfileArn="arn:aws:groundstation:us-west-2:111122223333:mission-profile/
example-profile",
    startTime=start_time,
    endTime=end_time,
    statusList=["AVAILABLE"],
)

if contacts["contactList"]:
    # Reserve the first available contact
    contact = contacts["contactList"][0]
    print(f"Reserving contact from {contact['startTime']} to {contact['endTime']}...")

    reservation = ground_station_client.reserve_contact(
        satelliteArn=satellite_arn, # Required for TLE/OEM ephemeris
        missionProfileArn="arn:aws:groundstation:us-west-2:111122223333:mission-
profile/example-profile",
        groundStation="Hawaii 1",
        startTime=contact["startTime"],
        endTime=contact["endTime"],
        # Note: trackingOverrides is optional for TLE/OEM
        # The system will use the highest priority ephemeris automatically
    )

    print(f"Reserved contact: {reservation['contactId']}")
else:
    print("No available contacts found")
```

工作流程 2：直接聯絡預留

此工作流程會直接保留聯絡人，而不會先列出可用的時段。當您已經知道所需的聯絡時間或正在實作自動化排程時，此方法很有用。

範例：使用方位角海拔暫時性的直接保留

```
import boto3
from datetime import datetime, timezone
import time

# Create AWS Ground Station client
```

```

ground_station_client = boto3.client("groundstation")

# Define contact window
contact_start = datetime(2024, 3, 20, 14, 0, 0, tzinfo=timezone.utc)
contact_end = datetime(2024, 3, 20, 14, 15, 0, tzinfo=timezone.utc)

# Create azimuth elevation ephemeris for the specific contact time
print("Creating azimuth elevation ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Direct Contact AzEl Ephemeris",
    ephemeris={
        "azEl": {
            "groundStation": "Ohio 1",
            "data": {
                "azElData": {
                    "angleUnit": "DEGREE_ANGLE",
                    "azElSegmentList": [
                        {
                            "referenceEpoch": contact_start.isoformat(),
                            "validTimeRange": {
                                "startTime": contact_start.isoformat(),
                                "endTime": contact_end.isoformat(),
                            },
                            "azElList": [
                                {"dt": 0.0, "az": 45.0, "el": 10.0},
                                {"dt": 180.0, "az": 50.0, "el": 15.0},
                                {"dt": 360.0, "az": 55.0, "el": 20.0},
                                {"dt": 540.0, "az": 60.0, "el": 25.0},
                                {"dt": 720.0, "az": 65.0, "el": 30.0},
                                {"dt": 900.0, "az": 70.0, "el": 35.0},
                            ],
                        },
                    ],
                },
            },
        },
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:

```

```

status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
    "status"
]
if status == "ENABLED":
    print("Ephemeris is ENABLED")
    break
elif status in ["INVALID", "ERROR"]:
    raise RuntimeError(f"Ephemeris failed: {status}")
time.sleep(5)

# Directly reserve the contact
print(f"Reserving contact from {contact_start} to {contact_end}...")

reservation = ground_station_client.reserve_contact(
    # Note: satelliteArn is omitted for azimuth elevation
    missionProfileArn="arn:aws:groundstation:us-east-2:111122223333:mission-profile/
example-profile",
    groundStation="Ohio 1",
    startTime=contact_start,
    endTime=contact_end,
    trackingOverrides={"programTrackSettings": {"azEl": {"ephemerisId":
ephemeris_id}}},
)

print(f"Reserved contact: {reservation['contactId']}")

```

範例：使用 TLE ephemeris 直接保留

```

import boto3
from datetime import datetime, timedelta, timezone
import time

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

satellite_id = "12345678-1234-1234-1234-123456789012"
satellite_arn = f"arn:aws:groundstation::111122223333:satellite/{satellite_id}"

# Define contact window (based on predicted pass)
contact_start = datetime(2024, 3, 21, 10, 30, 0, tzinfo=timezone.utc)
contact_end = datetime(2024, 3, 21, 10, 42, 0, tzinfo=timezone.utc)

# Create TLE ephemeris

```

```
print("Creating TLE ephemeris...")
ephemeris_response = ground_station_client.create_ephemeris(
    name="Direct Contact TLE Ephemeris",
    satelliteId=satellite_id,
    enabled=True,
    expirationTime=contact_end + timedelta(days=1),
    priority=1,
    ephemeris={
        "tle": {
            "tleData": [
                {
                    "tleLine1": "1 25994U 99068A 24080.50000000 .00000075 00000-0
26688-4 0 9999",
                    "tleLine2": "2 25994 98.2007 35.6589 0001234 89.2782 18.9934
14.57114995112000",
                    "validTimeRange": {
                        "startTime": (contact_start - timedelta(hours=1)).isoformat(),
                        "endTime": (contact_end + timedelta(hours=1)).isoformat(),
                    },
                }
            ]
        }
    },
)

ephemeris_id = ephemeris_response["ephemerisId"]
print(f"Created ephemeris: {ephemeris_id}")

# Wait for ephemeris to become ENABLED
while True:
    status = ground_station_client.describe_ephemeris(ephemerisId=ephemeris_id)[
        "status"
    ]
    if status == "ENABLED":
        print("Ephemeris is ENABLED")
        break
    elif status in ["INVALID", "ERROR"]:
        raise RuntimeError(f"Ephemeris failed: {status}")
    time.sleep(5)

# Directly reserve the contact
print(f"Reserving contact from {contact_start} to {contact_end}...")

reservation = ground_station_client.reserve_contact(
```

```
satelliteArn=satellite_arn, # Required for TLE ephemeris
missionProfileArn="arn:aws:groundstation:us-west-2:111122223333:mission-profile/
example-profile",
groundStation="Hawaii 1",
startTime=contact_start,
endTime=contact_end,
# Note: trackingOverrides is optional for TLE
# The system will use the highest priority ephemeris automatically
)

print(f"Reserved contact: {reservation['contactId']}")
```

監控聯絡狀態變更

保留聯絡人之後，請務必監控其狀態，以確保其成功轉換為 `SCHEDULED` 並在發生任何問題時收到通知。會向 Amazon EventBridge AWS Ground Station 發出所有聯絡人狀態變更的事件。

聯絡狀態遵循此生命週期：

- `SCHEDULING` - 正在處理聯絡人以進行排程
- `SCHEDULED` - 已成功排程聯絡人並將執行
- `FAILED_TO_SCHEDULE` - 無法排程聯絡人（終端狀態）

如需聯絡狀態和生命週期的詳細資訊，請參閱 [了解聯絡生命週期](#)。

使用 EventBridge 實作聯絡狀態監控

若要即時監控聯絡狀態變更，您可以設定每當 Ground Station 聯絡狀態變更時觸發 Lambda 函數的 Amazon EventBridge 規則。相較於輪詢聯絡狀態，此方法更有效率且更具可擴展性。

實作步驟

1. 建立 Lambda 函數來處理聯絡狀態變更事件
2. 建立符合 Ground Station 聯絡狀態變更事件的 EventBridge 規則
3. 新增 Lambda 函數做為規則的目標

Lambda 函數處理常式範例

如需處理聯絡狀態變更事件的 Lambda 函數完整範例，請參閱 `AquaSnppJpssTerraDigIF.yml` CloudFormation 範本中的 `GroundStationCloudWatchEventHandlerLambda` 資源。此範本可在

AWS Ground Station 客戶加入 Amazon S3 儲存貯體中取得。如需存取此範本的說明，請參閱資料流程端點範例的 [將它放在一起](#) 一節。

EventBridge 規則組態

EventBridge 規則應使用下列事件模式，以符合所有 Ground Station 聯絡狀態變更：

```
{
  "source": ["aws.groundstation"],
  "detail-type": ["Ground Station Contact State Change"]
}
```

若要篩選特定狀態（例如失敗），您可以新增詳細資訊篩選條件：

```
{
  "source": ["aws.groundstation"],
  "detail-type": ["Ground Station Contact State Change"],
  "detail": {
    "contactStatus": [
      "FAILED_TO_SCHEDULE",
      "FAILED",
      "AWS_FAILED",
      "AWS_CANCELLED"
    ]
  }
}
```

如需使用 Lambda 目標建立 EventBridge 規則的詳細說明，請參閱《Amazon EventBridge 使用者指南》中的 [建立對事件做出反應的規則](#)。

設定用於自動化的 EventBridge 規則

您可以建立 EventBridge 規則，以自動回應聯絡狀態變更。例如：

- 當聯絡人無法排程時傳送通知
- 觸發 Lambda 函數，以在聯絡進入時準備資源 PREPASS
- 記錄聯絡案例完成以進行稽核

如需設定 AWS Ground Station 事件的 EventBridge 規則的詳細資訊，請參閱 [AWS Ground Station 使用事件自動化](#)。

最佳實務和考量

處理排程衝突

由於 AWS Ground Station 天線是共用資源，因此在 中出現的聯絡時段ListContacts可能由其他客戶保留，然後才能進行保留。若要處理此問題：

1. 保留後一律檢查聯絡狀態
2. 使用替代時段實作重試邏輯
3. 可能的話，請考慮提前預留聯絡人
4. 使用 EventBridge 事件來監控FAILED_TO_SCHEDULE狀態

Ephemeris 驗證時間

請記住，ephemeris 必須處於 ENABLED 狀態，才能使用它來保留聯絡人。根據暫時性的類型和大小，驗證程序通常需要幾秒鐘到幾分鐘的時間。在嘗試保留聯絡人之前，請務必驗證暫時性狀態。

聯絡時間考量

使用自訂 ephemeris 時：

- 確保您的暫時性接觸涵蓋整個聯絡持續時間
- 對於方位角升高暫時性，請確認在整個接觸過程中，角度將天線保持在[站點遮罩](#)上方
- 安排未來的聯絡時，請考慮暫時性過期時間

依暫時性差異類型的 API 差異

ReserveContact API 的行為因暫時性變化類型而異：

Ephemeris 類型	satelliteArn 必要	trackingOverrides 必要
TLE	是	否 (選用)
OEM	是	否 (選用)
方位上升	否 (選用)	是

了解使用哪些 ephemeris

Ephemerides 具有優先順序、過期時間和啟用的旗標。這些會一起決定在聯絡期間用於追蹤的 ephemeris。

TLE 和 OEM 暫時化

對於 OEM 和 TLE 暫時性資料，每個衛星只能有一個暫時性資料處於作用中狀態。將使用的最高優先順序 ephemeris，其過期時間是未來的 ephemeris。優先順序值越大表示優先順序越高。[ListContacts](#) 傳回的可用聯絡時間是以此暫時性為基礎。如果多個 ENABLED ephemeride 具有相同的優先順序，則會使用最近建立或更新的 ephemeris。

Note

AWS Ground Station 對每個衛星ENABLED客戶提供的短暫性數量具有服務配額（請參閱：[Service Quotas](#)）。若要在達到此配額後上傳暫時性資料，請刪除（使用 [DeleteEphemeris](#)）或停用（使用 [UpdateEphemeris](#)）最低優先順序/最早建立的客戶提供暫時性資料。

如果尚未建立暫時性資料，或者如果沒有暫時性資料具有ENABLED狀態，則如果可用，AWS Ground Station 將使用衛星的預設暫時性資料（來自 [Space-Track](#)）。此預設 ephemeris 的優先順序為 0。

方位海拔半合子

方位角海拔半合子的運作方式與 OEM 和 TLE 半合子不同。每個方位角海拔高度均與特定地面站相關聯，且沒有優先順序。當您預留與方位角升高暫時性的接觸時，您會明確指定要透過 `trackingOverrides` 參數使用的方位角升高暫時性。

方位角海拔翅目的主要差異：

- 無優先順序系統 - 您明確選取每個聯絡人的暫時性
- 特定地面站 - 每個臨時工都與特定地面站相關聯
- 沒有自動備用 - 如果指定的短暫性不可用，聯絡將會失敗

Note

方位角海拔半合子不會與 OEM 和 TLE 半合子競爭。保留聯絡人時，系統會明確選取這些聯絡人，且只會在指定追蹤覆寫時使用。

新暫時性差異對先前排程聯絡人的影響

使用 [DescribeContact API](#) 透過傳回作用中的可見性時間，來檢視新短暫性對先前排程聯絡人的影響。

對於 OEM 和 TLE 暫時性資料，上傳新暫時性資料之前排程的聯絡案例將保留最初排程的聯絡時間，而天線追蹤將使用作用中暫時性資料。如果太空船的位置根據作用中的暫時性差異很大，則由於太空船在傳輸/接收網站遮罩之外操作，這可能會導致與天線的衛星接觸時間縮短。因此，我們建議您在上傳與先前 ephemeris 有很大差異的新 ephemeris 之後，取消和重新排程未來的聯絡人。

使用 [DescribeContact API](#)，您可以比較您的排程聯絡人和傳回的 `visibilityStartTime` `startTimeendTime` 和 `visibilityEndTime`，以判斷因傳輸/接收網站遮罩外操作的太空船而無法使用的未來聯絡人部分 `visibilityEndTime`。如果您選擇取消和重新排定未來的聯絡時間，聯絡時間範圍不能超過可見性時間範圍（超過 30 秒）。取消太接近聯絡時間時，取消聯絡可能會產生費用。如需取消聯絡人的詳細資訊，請參閱：[Ground Station FAQs](#)。

對於方位角海拔半合子，排程聯絡人將使用保留聯絡人時選取的特定半合子。如果您需要更新排程聯絡人的方位提升資料，您可以取消和重新排程具有新暫時性的聯絡人。

取得衛星的目前暫時性變化

透過呼叫 [GetSatellite](#) 或 [ListSatellites](#) 動作，可以擷取目前由 AWS Ground Station 用於特定衛星的暫時性。這兩種方法都會傳回目前使用中 ephemeris 的中繼資料。此暫時性中繼資料對於上傳至的自訂暫時性中繼資料 AWS Ground Station 和預設暫時性中繼資料不同。

Note

方位浮水印不會與衛星相關聯，因此 [GetSatellite](#) 或 [ListSatellites](#) 不會傳回。若要擷取方位角海拔半合子的相關資訊，請使用 [DescribeEphemeris](#) API 搭配特定半合子 ID，或使用 [ListEphemerides](#) 查看您帳戶的所有可用半合子。

預設對等詞只會包含 `source` 和 `epoch` 欄位。`epoch` 是從 [Space-Track](#) 提取的 [雙行元素集](#) 的 `epoch`，目前正在用於運算衛星的軌跡。

自訂 `ephemeris source` 的值為 `CUSTOMER_PROVIDED` 並在 `ephemerisId` 欄位中包含唯一識別符。此唯一識別符可用於透過 [DescribeEphemeris](#) 動作查詢 `ephemeris`。如果在 AWS Ground Station 透過 [CreateEphemeris](#) 動作上傳至期間為 `ephemeris` 指派名稱，則會傳回選用 `name` 欄位。

請務必注意，暫時性資料是由動態更新，AWS Ground Station 因此傳回的資料只是呼叫 API 時正在使用暫時性資料的快照。

使用預設 `ephemeris` 的衛星 [GetSatellite](#) 傳回範例

```
{
  "satelliteId": "e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
  "satelliteArn": "arn:aws:groundstation::111122223333:satellite/e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
  "noradSatelliteID": 25994,
  "groundStations": [
    "Ohio 1",
    "Oregon 1"
  ],
  "currentEphemeris": {
    "source": "SPACE_TRACK",
    "epoch": 1528245583.619
  }
}
```

使用自訂 `ephemeris` 的衛星 [GetSatellite](#) 範例

```
{
  "satelliteId": "e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
  "satelliteArn": "arn:aws:groundstation::111122223333:satellite/e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
  "noradSatelliteID": 25994,
  "groundStations": [
    "Ohio 1",
    "Oregon 1"
  ],
  "currentEphemeris": {
    "source": "CUSTOMER_PROVIDED",
    "ephemerisId": "e1cfe0c7-67f9-4d98-bad2-EXAMPLE",
    "name": "My Ephemeris"
  }
}
```

列出方位角升高環羈化物

由於方位角海拔半合子與衛星無關，因此您需要使用不同的 APIs 來探索和擷取其相關資訊：

1. 使用 [ListEphemerides](#) 列出您 帳戶中的所有半合子，包括方位角海拔半合子。您可以依狀態和暫時性類型進行篩選。
2. 使用 [DescribeEphemeris](#) 搭配特定 ephemeris ID 來取得方位海拔 ephemeris 的詳細資訊。
3. 使用 [DescribeContact](#) 搭配特定聯絡人 ID，以取得用於聯絡人的暫時性詳細資訊。

[ListEphemerides](#) 回應範例，包括方位角升高暫時性：

```
{
  "ephemerides": [
    {
      "ephemerisId": "abc12345-6789-def0-1234-5678EXAMPLE",
      "ephemerisType": "AZ_EL",
      "name": "Azimuth Elevation for Ohio Ground Station",
      "status": "ENABLED",
      "creationTime": 1620254718.765
    },
    {
      "ephemerisId": "def45678-9012-abc3-4567-8901EXAMPLE",
      "ephemerisType": "TLE",
      "name": "TLE for Satellite 12345",
      "status": "ENABLED",
      "creationTime": 1620254700.123
    }
  ]
}
```

Note

在 [ListEphemerides](#) 回應中，方位角海拔半合子將具有 `groundStation` 欄位，而不是 `satelliteId` 欄位，使其易於識別。

還原至預設 ephemeris 資料

當您上傳自訂 ephemeris 資料時，它將覆寫該特定衛星 AWS Ground Station 使用的預設 ephemerides。AWS Ground Station 不會再次使用預設 ephemerides，直到目前沒有啟用、未過期的客戶提供的 ephemerides 可供使用為止。AWS Ground Station 也不會列出目前客戶提供的 ephemeris 過期時間之後的聯絡人，即使該過期時間之後仍有預設 ephemeris 可用。

Note

方位角海拔半合子沒有預設值，也不會覆寫衛星半合子。使用 `trackingOverrides` 參數保留聯絡人時，系統會明確選取這些項目。如果您不想再使用方位浮水印，只需保留聯絡人而不指定追蹤覆寫，系統會改用作用中的衛星浮水印。

還原 TLE 和 OEM 暫時化

若要還原至衛星的預設 [Space-Track](#) 暫時化，您需要執行下列其中一項操作：

- 刪除（使用 [DeleteEphemeris](#)）或停用（使用 [UpdateEphemeris](#)）所有已啟用客戶提供的暫時性資料。您可以使用 [ListEphemerides](#) 列出客戶提供的衛星短暫性。
- 等待所有現有客戶提供的暫時性資料過期。

您可以呼叫 [GetSatellite](#) 並確認目前衛星的 ephemeris 為 `SPACE_TRACK`，以確認 source 正在使用預設 ephemerisSPACE_TRACK。如需預設 ephemerides 的詳細資訊 [預設暫時性資料](#)，請參閱。

管理方位角升高環羈化物

由於針對每個聯絡人明確選取方位角海拔短暫性，且未與衛星建立關聯，因此沒有「還原」為預設值的概念。反之，您可以管理方位角海拔環羈化物，如下所示：

- 若要停止使用方位提升暫時性：只需保留新的聯絡人，而不指定 `trackingOverrides`。`satelliteArn` 聯絡人將改為為指定的衛星使用作用中的 ephemeris。
- 若要移除未使用的方位角海拔半合子：使用 [DeleteEphemeris](#) 刪除不再需要的方位角海拔半合子。請注意，您無法刪除排程聯絡人目前正在使用的暫時性出血。

若要列出您帳戶中的所有方位方位海拔半合子，請使用 [ListEphemerides](#)。方位海拔半合子可以由 `ephemerisType` 欄位識別，也可以透過 `groundStation` 回應中欄位的存在來識別，而不是 `satelliteId` 欄位。

使用資料流程

AWS Ground Station 使用節點和邊緣關係來建構資料流程，以啟用資料的串流處理。每個節點都由描述其預期處理的組態表示。若要說明此概念，請考慮將的資料流程antenna-downlink傳送至 s3-recording。antenna-downlink 節點代表根據組態上定義的參數，無線電頻譜的類比到數位轉型。s3-recording 代表將接收傳入資料的運算節點，並將其存放在 S3 儲存貯體中。產生的資料流程是根據您的規格，將數位化 RF 資料非同步交付至 S3 儲存貯體的資料。

在任務描述檔中，您可以建立許多資料流程來滿足您的需求。下列各節說明如何設定要與 搭配使用的其他 AWS 資源，AWS Ground Station 並提供建構資料流程的建議。如需每個節點行為的詳細資訊，包括是否被視為來源或目的地節點，請參閱 [使用 AWS Ground Station 組態](#)。

主題

- [AWS Ground Station 資料平面界面](#)
- [使用跨區域資料交付](#)
- [設定 Amazon S3](#)
- [設定 Amazon VPC](#)
- [設定 Amazon EC2](#)

AWS Ground Station 資料平面界面

所選資料流程產生的資料結構取決於資料流程的來源。這些格式的詳細資訊會在衛星加入期間提供給您。以下摘要說明每種資料流程類型所使用的格式。

- antenna-downlink
 - (頻寬less-than-or-equal-to 40MHz) 資料會以 [VITA-49 訊號資料/IP](#) 格式封包傳送。
 - (大於 40MHz 的頻寬) 資料會以 AWS Ground Station Class 2 封包的形式交付。
- antenna-downlink-demod-decode
 - 資料會以解調/解碼資料/IP 格式封包的形式交付。
- antenna-uplink
 - 資料必須以 [VITA-49 訊號資料/IP](#) 格式封包傳送。
- antenna-uplink-echo
 - 資料會以 [VITA-49 訊號資料/IP](#) 格式封包的形式交付。

使用跨區域資料交付

AWS Ground Station 跨區域資料交付功能可讓您靈活地將資料從天線傳送至任何 AWS Ground Station 支援 AWS 的區域。這表示您可以在單一 AWS 區域中維護基礎設施，並在 [AWS Ground Station 位置](#) 您加入的任何 上排程聯絡人。

在 Amazon S3 儲存貯體中接收您的聯絡資料時，AWS Ground Station 會為您管理所有交付層面。

若要使用跨區域資料交付至 Amazon EC2 執行個體（使用 AWS Ground Station 代理程式或資料流程端點），必須在您目前的 AWS 區域中建立資料流程端點，且 dataflow-endpoint-config 必須指定相同的區域。AWS Ground Station 會為您管理跨區域交付資料。

設定 Amazon S3

您可以使用 Amazon S3 儲存貯體來接收使用的下行訊號 AWS Ground Station。若要建立目的地 s3-recording-config，您必須能夠指定 Amazon S3 儲存貯體和 IAM 角色，授權 AWS Ground Station 將檔案寫入儲存貯體。

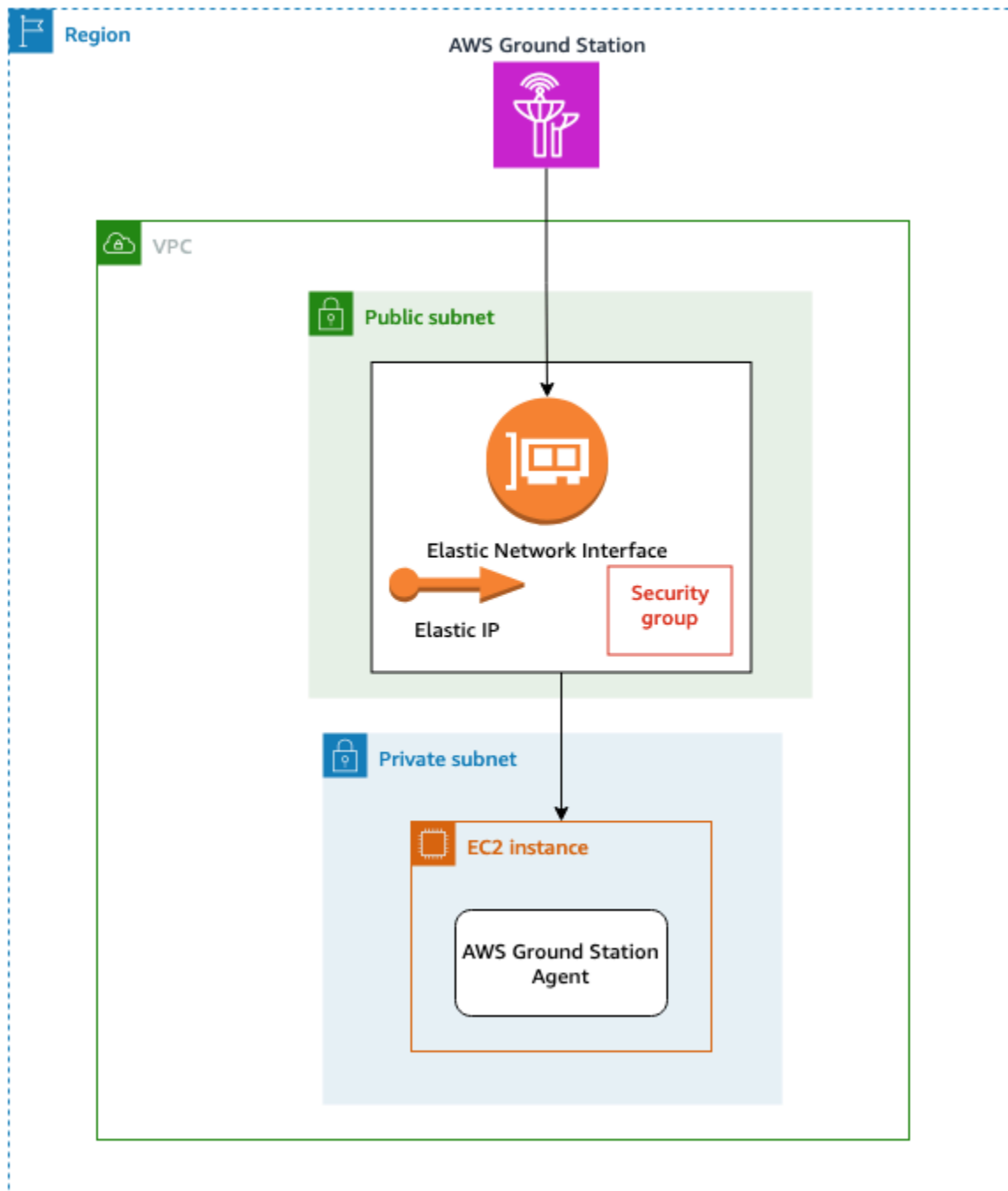
[Amazon S3 錄製組態](#) 如需 Amazon S3 儲存貯體、IAM 角色或組態建立的限制，請參閱 AWS Ground Station。

設定 Amazon VPC

設定 VPC 的完整指南超出本指南的範圍。如需深入了解，請參閱 [Amazon VPC 使用者指南](#)。

在本節中，會說明您的 Amazon EC2 和資料流程端點如何在 VPC 內存在。AWS Ground Station 不支援特定資料流程的多個交付點 - 預期每個資料流程都會終止為單一 EC2 接收器。由於我們預期單一 EC2 接收器，因此組態不是多可用區域備援。如需將使用您的 VPC 的完整範例，請參閱 [範例任務設定檔組態](#)。

使用 AWS Ground Station 代理程式的 VPC 組態



您的衛星資料會提供給靠近天線的 AWS Ground Station 代理程式執行個體。AWS Ground Station 代理程式會分割資料，然後使用您提供的 AWS KMS 金鑰加密資料。每個條紋都會從 AWS 網路骨幹的來源天線傳送到您的 [Amazon EC2 彈性 IP \(EIP\)](#)。資料會透過連接的 Amazon EC2 彈性網路界面 (ENI) 送達您的 EC2 執行個體。 [Amazon EC2](#) 在您的 EC2 執行個體上，已安裝的 AWS Ground

Station 代理程式會解密您的資料並執行轉送錯誤修正 (FEC) 以復原任何捨棄的資料，然後將其轉送至您在設定中指定的 IP 和連接埠。

以下清單會在設定 VPC 進行 AWS Ground Station 客服人員交付時，指出唯一的設定考量。

安全群組 - 建議您設定僅限 AWS Ground Station 流量的安全群組。此安全群組應允許 UDP 輸入流量位於您在 Dataflow 端點群組中指定的相同連接埠範圍。AWS Ground Station 會維護 AWS 受管字首清單，將您的許可限制為僅限 AWS Ground Station IP 地址。如需如何取代部署區域的 PrefixListId 的詳細資訊，請參閱 [AWS 受管字首清單](#)。

彈性網路界面 (ENI) - 您需要將上述安全群組與此 ENI 建立關聯，並將其放置在公有子網路中。

Note

每個 ENI 連接的安全群組數量預設配額為 5。這是最多 16 個可調整的限制，請參閱 [Amazon VPC Quotas](#)。

下列 CloudFormation 範本示範如何建立本節所述的基礎設施。

ReceiveInstanceEIP:

```
Type: AWS::EC2::EIP
Properties:
  Domain: 'vpc'
```

InstanceSecurityGroup:

```
Type: AWS::EC2::SecurityGroup
Properties:
  GroupDescription: AWS Ground Station receiver instance security group.
  VpcId: YourVpcId
  SecurityGroupIngress:
    # Add additional items here.
    - IpProtocol: udp
      FromPort: your-port-start-range
      ToPort: your-port-end-range
      PrefixListIds:
        - PrefixListId: com.amazonaws.global.groundstation
  Description: "Allow AWS Ground Station Downlink ingress."
```

InstanceNetworkInterface:

```
Type: AWS::EC2::NetworkInterface
Properties:
```

Description: *ENI for AWS Ground Station to connect to.*

GroupSet:

- !Ref *InstanceSecurityGroup*

SubnetId: *A Public Subnet*

ReceiveInstanceEIPAllocation:

Type: AWS::EC2::EIPAssociation

Properties:

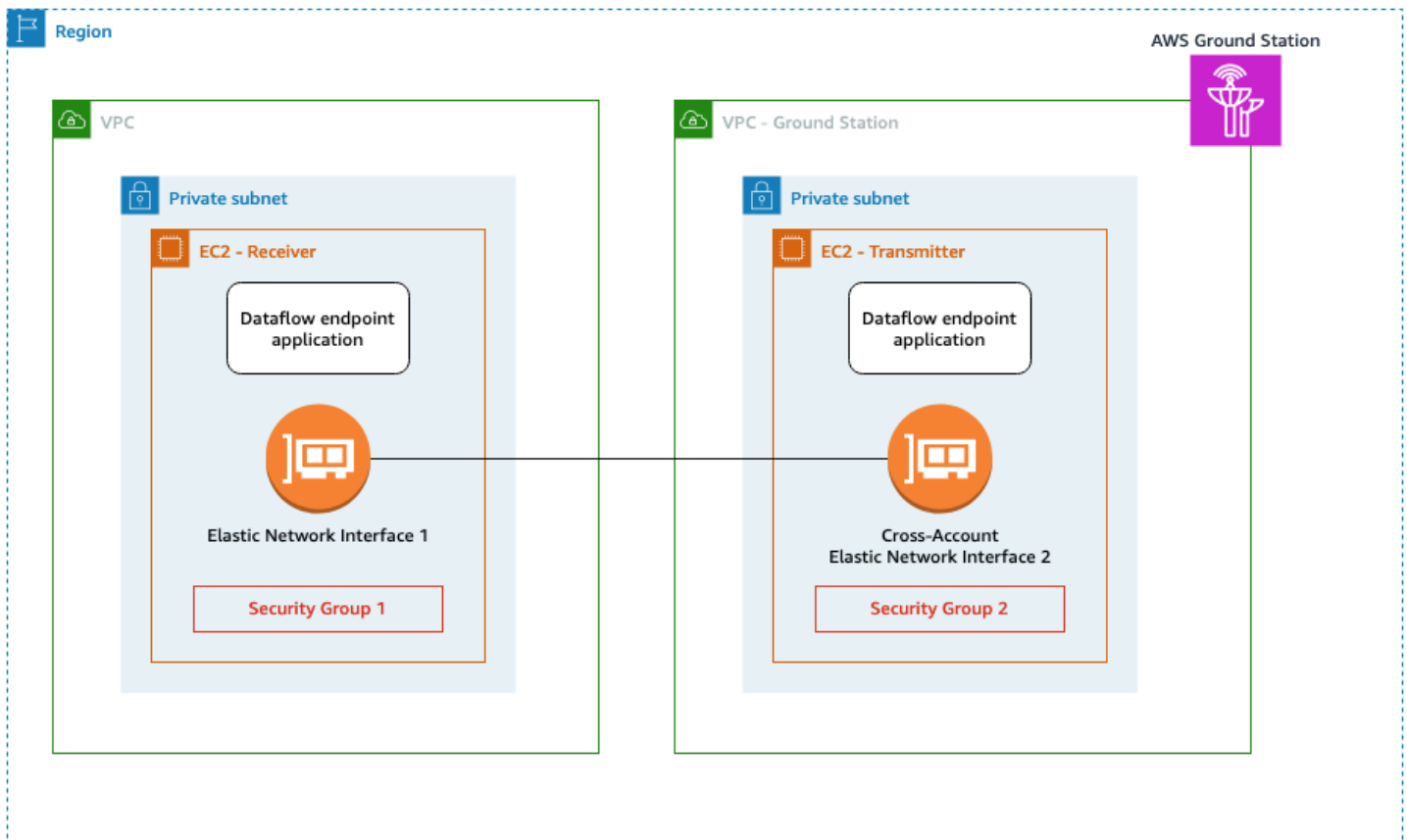
AllocationId:

Fn::GetAtt: [*ReceiveInstanceEIP*, AllocationId]

NetworkInterfaceId:

Ref: *InstanceNetworkInterface*

具有資料流程端點的 VPC 組態



您的衛星資料會提供給靠近天線的資料流程端點應用程式執行個體。然後，資料會透過跨帳戶 [Amazon EC2 彈性網路界面 \(ENI\)](#) 從擁有的 VPC 傳送 AWS Ground Station。然後，資料會透過連接到 Amazon EC2 執行個體的 ENI 到達 EC2 執行個體。Amazon EC2 然後，安裝的資料流程端點應用程式會將其轉送到您在設定中指定的 IP 和連接埠。上行連線會發生此流程的反向。

以下清單在設定 VPC 進行資料流程端點交付時，會發出唯一的設定考量。

Note

每個 ENI 連接的安全群組數量預設配額為 5。這是最多 16 個可調整的限制，請參閱 [Amazon VPC Quotas](#)。

IAM 角色 - IAM 角色是 Dataflow 端點的一部分，不會顯示在圖表中。用於建立跨帳戶 ENI 並將其連接至 AWS Ground Station Amazon EC2 執行個體的 IAM 角色。

安全群組 1 - 此安全群組會連接到 ENI，該 ENI 將與您帳戶中的 Amazon EC2 執行個體相關聯。它需要在 dataflow-endpoint-group 群組中指定的連接埠上允許來自安全群組 2 的 UDP 流量。

彈性網路界面 (ENI) 1 - 您需要將安全群組 1 與此 ENI 建立關聯，並將其放置在子網路中。

子網路 - 您需要確保帳戶中 Amazon EC2 執行個體的每個資料流程至少有一個可用的 IP 地址。如需子網路大小的詳細資訊，請參閱 [子網路 CIDR 區塊](#)

安全群組 2 - 在 Dataflow 端點中參考此安全群組。此安全群組將連接到 ENI，該 ENI AWS Ground Station 將用來將資料放入您的帳戶。

區域 - 如需跨區域連線支援區域的詳細資訊，請參閱 [使用跨區域資料交付](#)。

下列 CloudFormation 範本示範如何建立本節所述的基礎設施。

DataflowEndpointSecurityGroup:

Type: AWS::EC2::SecurityGroup

Properties:

GroupDescription: Security Group for AWS Ground Station registration of Dataflow Endpoint Groups

VpcId: *YourVpcId*

AWSGroundStationSecurityGroupEgress:

Type: AWS::EC2::SecurityGroupEgress

Properties:

GroupId: !Ref: *DataflowEndpointSecurityGroup*

IpProtocol: udp

FromPort: *5555*

ToPort: *5555*

CidrIp: *10.0.0.0/8*

Description: *"Allow AWS Ground Station to send UDP traffic on port 55555 to the 10/8 range."*

InstanceSecurityGroup:

Type: AWS::EC2::SecurityGroup

Properties:

GroupDescription: *AWS Ground Station receiver instance security group.*

VpcId: *YourVpcId*

SecurityGroupIngress:

- IpProtocol: *udp*

FromPort: *55555*

ToPort: *55555*

SourceSecurityGroupId: *!Ref DataflowEndpointSecurityGroup*

Description: *"Allow AWS Ground Station Ingress from DataflowEndpointSecurityGroup"*

ReceiverSubnet:

Type: AWS::EC2::Subnet

Properties:

Ensure your CidrBlock will always have at least one available IP address per dataflow endpoint.

See <https://docs.aws.amazon.com/vpc/latest/userguide/subnet-sizing.html> for subnet sizing guidelines.

CidrBlock: *"10.0.0.0/24"*

Tags:

- Key: *"Name"*

Value: *"AWS Ground Station - Dataflow endpoint Example Subnet"*

- Key: *"Description"*

Value: *"Subnet for EC2 instance receiving AWS Ground Station data"*

VpcId: *!Ref ReceiverVPC*

設定 Amazon EC2

同步交付 VITA-49 Signal/IP 資料或 VITA-49 Extension data/IP 需要正確設定 Amazon EC2 執行個體，才能透過 AWS Ground Station 代理程式或資料流程端點交付。根據您的特定需求，您可以直接在同一個執行個體上執行前端 (FE) 處理器或軟體定義無線電 (SDR)，或者您可能需要使用其他 EC2 執行個體。選擇和安裝您的 FE 或 SDR 超出本使用者指南的範圍。如需特定資料格式的詳細資訊，請參閱 [AWS Ground Station 資料平面界面](#)。

如需服務條款的相關資訊，請參閱 [AWS 服務條款](#)。

提供的常見軟體

AWS Ground Station 提供常見的軟體，以簡化 Amazon EC2 執行個體的設定。

AWS Ground Station 代理程式

AWS Ground Station 代理程式會接收數位中繼頻率 (DigIF) 下行資料，並輸出解密的資料以啟用下列項目：

- 從 40 MHz 到 400 MHz 頻寬的 DigIF 下行功能。 MHz MHz
- 將高速率、低抖動的 DigIF 資料交付至 AWS 網路上的任何公有 IP (AWS 彈性 IP)。
- 使用轉送錯誤校正 (FEC) 提供可靠的資料。
- 使用客戶受管 AWS KMS 金鑰進行加密，以保護資料交付。

如需詳細資訊，請參閱[AWS Ground Station 客服人員使用者指南](#)。

Dataflow 端點應用程式

用於 AWS Ground Station 在 AWS Ground Station 天線位置與 Amazon EC2 執行個體之間傳送和接收資料的聯網應用程式。它可用於資料的上行和下行。

軟體定義的無線電 (SDR)

軟體定義的無線電 (SDR)，可用於調節/取消調節用於與衛星通訊的訊號。

AWS Ground Station Amazon Machine Image AMIs)

為了減少這些安裝的建置和組態時間，AWS Ground Station 也提供預先設定的 AMIs。加入完成後，您的帳戶即可使用具有資料流程端點聯網應用程式的 AMIs 和軟體定義的無線電 (SDR)。您可以在 Amazon EC2 主控台中搜尋私有 [Amazon Machine Image \(AMIs\)](#) 中的地面站，以找到它們。具有 AWS Ground Station 代理 AMIs 是公有的，可以透過在公有 Amazon Machine Image (AMI) 中搜尋地面站，在 Amazon EC2 主控台中找到。 [AMIs](#)

使用遙測

AWS Ground Station 遙測在衛星聯絡期間從 AWS Ground Station 天線提供近乎即時的指標。您可以使用遙測資料來監控聯絡效能、偵測異常狀況，並針對衛星通訊做出明智的決策。

主題

- [遙測的運作方式](#)
- [可用的遙測類型](#)
- [區域可用性](#)
- [設定遙測](#)
- [了解遙測資料](#)

遙測的運作方式

若要使用遙測，您可以設定 `TelemetrySinkConfig`，指定 AWS Ground Station 應在何處傳送遙測資料。然後，您可以使用 `telemetrySinkConfigArn` 欄位將此組態新增至您的任務描述檔。在使用啟用遙測功能任務設定檔的聯絡人期間，會將遙測資料 AWS Ground Station 串流至您的帳戶。

遙測交付程序的運作方式如下：

1. 您可以在 AWS 帳戶中建立 Kinesis Data Streams 串流，以接收遙測資料。串流必須在您排程聯絡人的相同帳戶和區域中建立。
2. 您可以建立 IAM 角色，授予將資料寫入串流的 AWS Ground Station 許可。
3. 您可以建立參考串流和 IAM 角色的 `TelemetrySinkConfig`。
4. 您可以將 `TelemetrySinkConfig` 新增至任務描述檔。
5. 您可以使用啟用遙測功能的新任務設定檔來列出和保留聯絡人。
6. 在使用此任務描述檔的聯絡期間，會以近乎即時的方式將遙測資料 AWS Ground Station 串流到您的 Kinesis Data Streams 串流。
7. 您可以使用 AWS 服務或您自己的應用程式，從串流取用和處理遙測資料。

可用的遙測類型

AWS Ground Station 在聯絡期間提供下列遙測類型：

Note

AWS Ground Station 正在擴展支援的遙測類型數量

指向遙測

提供衛星接觸期間天線指向方向的相關資訊。此遙測類型一律會在聯絡期間傳送，並包含實際和命令方位角和高度角度。如需詳細資訊，請參閱[指向遙測](#)。

追蹤遙測

提供天線追蹤狀態和追蹤錯誤的相關資訊。在追蹤組態中啟用自動追蹤時，會傳送此遙測類型。如需詳細資訊，請參閱[追蹤遙測](#)。

區域可用性

AWS Ground Station 運作的所有 AWS 區域皆可使用遙測。在聯絡執行期間，遙測會從 AWS Ground Station 天線傳送到您排程聯絡的區域，提供跨區域支援。

如需 AWS Ground Station 區域和地面站點位置的完整清單，請參閱[AWS Ground Station 位置](#)。

設定遙測

請依照下列步驟為您的 AWS Ground Station 聯絡人設定遙測。完成此設定後，遙測資料會在使用啟用遙測功能的任務設定檔的聯絡期間交付到您的 Kinesis Data Streams 串流。如需 Kinesis Data Streams 的深入了解，請參閱[Kinesis Data Streams 使用者指南](#)。

步驟 1：建立先決條件 AWS 資源

下列程式碼 CloudFormation 片段示範如何建立遙測交付的先決條件 AWS 資源。此程式碼片段會建立 Kinesis Data Streams 串流和 IAM 角色，授予將遙測資料寫入串流的 AWS Ground Station 許可。

```
TelemetryStream:
  Type: AWS::Kinesis::Stream
  Properties:
    Name: GroundStationTelemetryStream
    StreamModeDetails:
      StreamMode: ON_DEMAND
```

RetentionPeriodHours: *24*

TelemetryRole:

Type: AWS::IAM::Role

Properties:

RoleName: *GroundStationTelemetryRole*

AssumeRolePolicyDocument:

Version: '2012-10-17'

Statement:

- Effect: Allow
- Principal:
 - Service: groundstation.amazonaws.com
- Action: sts:AssumeRole

Policies:

- PolicyName: *KinesisWritePolicy*

PolicyDocument:

Version: '2012-10-17'

Statement:

- Effect: Allow
- Action:
 - kinesis:DescribeStream
 - kinesis:PutRecord
 - kinesis:PutRecords
- Resource: !GetAtt *TelemetryStream*.Arn

下列清單會在設定遙測交付時，指出唯一的設定考量 AWS Ground Station。

Kinesis Data Streams 串流 - 串流使用隨需容量模式，可根據輸送量自動擴展。對於大多數使用案例，建議使用此選項。串流設定為保留資料 24 小時。根據預設，串流會使用 AWS 受管加密。若要搭配使用客戶受管加密 AWS Key Management Service，請新增 StreamEncryption 屬性並更新 IAM 角色政策以包含 kms:GenerateDataKey 許可。如需詳細資訊，請參閱 [Amazon Kinesis Data Streams 中的資料保護](#)。

IAM 角色 - IAM 角色允許 groundstation.amazonaws.com 服務主體擔任該角色，並將遙測資料寫入您的 Kinesis Data Streams 串流。角色政策會授予串流上 kinesis:DescribeStream、kinesis:PutRecord 和 kinesis:PutRecords 動作的許可。[遙測接收器組態](#) 如需設定信任政策和角色政策的指引，請參閱。

其他組態 - 將 iam:PassRole 許可新增至您用於 AWS Ground Station API 呼叫的 IAM 使用者或角色。這可讓您在建立 TelemetrySinkConfig AWS Ground Station 時將遙測角色傳遞至。

TelemetrySinkConfig

PassRole 政策範例

如需如何更新或連接角色政策的詳細資訊，請參閱 [《IAM 使用者指南》中的管理 IAM 政策](#)。如需 iam:PassRole 許可的詳細資訊，請參閱 [授予使用者將角色傳遞至 AWS 服務的許可](#)

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::999999999999:role/your-telemetry-delivery-role-name"
    }
  ]
}
```

步驟 2：建立 TelemetrySinkConfig

建立 TelemetrySinkConfig，定義 AWS Ground Station 如何將遙測資料交付至 Kinesis Data Streams 串流。使用步驟 1 中 CloudFormation 堆疊輸出的串流 ARN 和角色 ARN。

Note

當您建立 TelemetrySinkConfig 時，AWS Ground Station 將透過提供分割區索引鍵為 的空白測試記錄，來驗證對 Kinesis Data Streams 串流的存取 test。

如需建立 TelemetrySinkConfig 的詳細資訊，請參閱 [遙測接收器組態](#)。

步驟 3：將遙測新增至您的任務設定檔

建立任務描述檔。如需建立任務描述檔的詳細資訊，請參閱 [使用 AWS Ground Station 任務設定檔](#)。將 telemetrySinkConfigArn 新增至您的任務設定檔，以在聯絡期間啟用遙測交付。使用在步驟 2 中建立的 TelemetrySinkConfig 的 ARN。

步驟 4：排程聯絡

使用啟用遙測功能的任務設定檔來排程聯絡人。在聯絡期間，AWS Ground Station 會將遙測資料串流到您的 Kinesis Data Streams 串流。

聯絡期間預期會發生的情況

- 遙測開始 - 資料會在聯絡人開始時開始串流。
- 近乎即時的交付 - 遙測會以近乎即時的方式送達 Kinesis Data Streams 串流。
- 聯絡持續時間 - 資料在整個聯絡過程中持續存在。
- 自動停止 - 遙測會在聯絡結束時停止串流。

監控交付

您可以使用下列方式監控遙測交付：

- Kinesis Data Streams 串流指標 - 檢查 CloudWatch 中的傳入記錄。如需詳細資訊，請參閱[監控 Amazon Kinesis Data Streams](#)。
- 應用程式日誌 - 驗證應用程式中從串流取用的資料處理。
- Kinesis Data Viewer - 使用 Kinesis Data Streams 串流主控台檢視來自串流的範例記錄。

後續步驟

完成設定後，您可以：

- 了解遙測資料格式和可用的遙測類型。請參閱 [了解遙測資料](#)。
- 建置應用程式來處理來自 Kinesis Data Streams 串流的遙測資料。如需詳細資訊，請參閱 [為 Amazon Kinesis Data Streams 建置消費者](#)。
- 使用 CloudWatch 和其他 AWS 服務建立儀表板和提醒。
- 如果您遇到問題，請檢閱疑難排解指引。請參閱 [對遙測進行故障診斷](#)。

了解遙測資料

遙測資料會以 Base64-encoded 的 JSON 記錄形式交付到您的 Kinesis Data Streams 串流。每個記錄都包含在衛星聯絡期間收集的資訊，包括有關聯絡和取樣遙測測量的中繼資料。

資料格式概觀

每個遙測記錄都包含下列元件：

遙測類型和版本

識別特定類型的遙測資料及其結構描述版本。這可讓您適當地剖析不同的遙測類型。如需結構描述版本控制的詳細資訊，請參閱 [結構描述版本控制和演變](#)。

範圍 ID

遙測範圍的唯一識別符。這可讓您將遙測資料與特定聯絡人建立關聯。

中繼資料

遙測的相關內容資訊。

資料

特定於遙測類型的取樣遙測測量。

分割區索引鍵

遙測記錄會以下列格式使用分割區索引鍵傳送到 Kinesis Data Streams 串流：

```
SCOPE#scopeId#TELEMETRY_ID#telemetryId#TELEMETRY_VERSION#telemetryVersion
```

此分割區索引鍵可確保單一聯絡的所有指定類型的遙測都會交付至 Kinesis Data Streams 串流中的相同碎片，為該聯絡的遙測串流提供最大努力的排序。

指向遙測

指向遙測提供衛星接觸期間天線指向方向的相關資訊。此遙測類型一律會在聯絡期間傳送。

資料欄位

sampleTimestamp

取樣遙測資料的時間，採用 ISO-8601 格式的 UTC，精確度為毫秒。

方位角

天線的實際方位角，以度為單位。

海拔

天線的實際高度角度，以度為單位。

commandedAzimuth

以度為單位的命令方位角。這是天線嘗試達到的目標方位角。

commandedElevation

命令抬高角度，以度為單位。這是天線嘗試達到的目標高度角度。

Note

由於實體限制或聯絡期間的機械延遲，實際天線位置可能與命令位置不同。

中繼資料欄位

groundStation

地面站點的名稱（例如，「俄亥俄 1」）。

satelliteId

衛星資源的識別符 AWS Ground Station。

contactId

聯絡人的識別符。

範例 JSON

```
{
  "telemetryTypeAndVersion": "POINTING#1.0.0",
  "telemetryType": "POINTING",
  "telemetryVersion": "1.0.0",
  "scopeId": "12345678-1234-1234-1234-123456789012",
  "metadata": {
    "groundStation": "Ohio 1",
    "satelliteId": "87654321-4321-4321-4321-210987654321",
    "contactId": "12345678-1234-1234-1234-123456789012"
  }
}
```

```
},  
"data": {  
  "sampleTimestamp": "2025-12-08T12:00:00.123Z",  
  "azimuth": 180.5,  
  "elevation": 45.2,  
  "commandedAzimuth": 180.0,  
  "commandedElevation": 45.0  
}  
}
```

追蹤遙測

追蹤遙測提供天線追蹤狀態和追蹤錯誤的相關資訊。此遙測類型會在追蹤組態中啟用自動追蹤，以及天線主動使用自動追蹤時傳送。

Note

如果 TrackingConfig 中的 autotrack 參數設定為 REMOVED，則不會傳送追蹤遙測。如需追蹤組態的詳細資訊，請參閱 [追蹤組態](#)。

資料欄位

sampleTimestamp

取樣遙測資料的時間，採用 ISO-8601 格式的 UTC，精確度為毫秒。

trackingStatus

天線的目前追蹤狀態。可能值為：

- TRACKING — 天線已成功鎖定在符合任務描述檔的訊號上，並主動跨天空跟隨它。這是聯絡期間的名目操作狀態。
- ACQUIRING — 天線正在定位和鎖定訊號。系統目前正在使用程式設計追蹤，根據暫時性資料指向。
- MASKED — 衛星的預測位置位於自動追蹤遮罩後方，這表示天線無法可靠地在該特定指向方向利用自動追蹤。這通常發生在高 RF 干擾的區域，例如低海拔。

trackingErrorAzimuth

方位軸中的追蹤錯誤，以度為單位。

trackingErrorElevation

高度軸中的追蹤錯誤，以度為單位。

Note

追蹤誤差值代表來自暫時性程式軌跡的調整，在自動追蹤期間 AWS Ground Station 套用以最大化訊號強度。

中繼資料欄位

追蹤遙測包含與指向遙測相同的中繼資料欄位：satelliteId、groundStation和 contactId。

範例 JSON

```
{
  "telemetryTypeAndVersion": "TRACKING#1.0.0",
  "telemetryType": "TRACKING",
  "telemetryVersion": "1.0.0",
  "scopeId": "12345678-1234-1234-1234-123456789012",
  "metadata": {
    "groundStation": "Ohio 1",
    "satelliteId": "87654321-4321-4321-4321-210987654321",
    "contactId": "12345678-1234-1234-1234-123456789012"
  },
  "data": {
    "sampleTimestamp": "2025-12-08T12:00:00.123Z",
    "trackingStatus": "TRACKING",
    "trackingErrorAzimuth": 0.2,
    "trackingErrorElevation": 0.1
  }
}
```

從 Kinesis Data Streams 串流讀取資料

遙測資料會傳送到 Kinesis Data Streams 串流，並且可以使用標準串流耗用模式來取用。從串流讀取資料時，請謹記下列考量。

Base64 解碼

Kinesis Data Streams 串流中的資料是 Base64-encoded。您必須先解碼資料，才能將其剖析為 JSON。如需詳細資訊，請參閱[使用 Amazon Kinesis Data Streams](#)。

使用 Kinesis Data Viewer

為了快速存取遙測資料，Kinesis Data Streams 串流主控台提供 Data Viewer 功能。使用此功能時：

- 遙測交付可能發生在您串流中的任何碎片。
- 預設開始位置會從碎片中的最新記錄讀取。
- 您可能需要調整選取的碎片，並使用「時間戳記」開始位置來檢視收到的記錄。

使用 Kinesis 用戶端程式庫

Kinesis Client Library (KCL) 可管理從 Kinesis Data Streams 串流耗用資料的許多相關複雜性，包括碎片管理、檢查點和負載平衡。我們建議將 KCL 用於生產遙測消耗應用程式。

如需詳細資訊，請參閱[使用 Kinesis Client Library 開發消費者](#)。

消耗的最佳實務

- 將延遲降至最低 - 使用增強廣發功能從 Kinesis Data Streams 串流讀取，具有專用輸送量，相較於輪詢，延遲更低。如需詳細資訊，請參閱[開發增強型廣發消費者](#)。
- 專用串流 - 使用專用 Kinesis Data Streams 串流進行 AWS Ground Station 遙測整合。與其他應用程式共用串流可能會導致寫入輸送量飽和和遙測交付失敗。
- 隨需容量 - 在隨需佈建模式下部署 Kinesis Data Streams 串流，以根據輸送量自動擴展碎片。
- 監控輸送量 - 使用 CloudWatch 指標監控串流的限流。如需詳細資訊，請參閱[監控 Amazon Kinesis Data Streams](#)。

結構描述版本控制和演變

遙測結構描述已進行版本控制，以支援隨著時間的推移而演進。每個記錄中 telemetryVersion 的欄位都會指出結構描述版本。

處理結構描述變更

- 未來可能會引進新的遙測類型。
- 現有的遙測類型可能會收到具有重大變更的新版本。
- 您的應用程式應可容忍未知的遙測類型和版本。

- 剖析 `telemetryTypeAndVersion`、`telemetryType`和 `telemetryVersion` 欄位，以決定如何處理每個記錄。

我們建議實作可正常處理多個結構描述版本的版本感知承載序列化，讓您的應用程式在推出新版本時繼續運作。

使用聯絡人

您可以使用 AWS Ground Station 主控台或您選擇的語言 AWS SDK，輸入衛星資料、識別天線位置 AWS CLI、通訊和排程所選衛星的天線時間。您可以在聯絡開始之前最多 15 分鐘檢閱、取消和重新排程聯絡保留*。您也可以更新聯絡人以指定暫時性覆寫，包括方位角/上升、OEM 或 TLE 追蹤資料，或變更目標衛星。如需詳細資訊，請參閱[更新聯絡人和聯絡人版本控制](#)。此外，如果您正在使用 AWS Ground Station 保留分鐘定價模型，您可以檢視您的保留分鐘定價計劃詳細資訊。

AWS Ground Station 支援跨區域資料交付。資料流程端點組態為您選取任務設定檔的一部分，會決定資料傳送到哪個區域。如需使用跨區域資料交付的詳細資訊，請參閱[使用跨區域資料交付](#)。

若要排程接收，必須設定您的資源。如果您尚未設定資源，請參閱[開始使用](#)。呼叫 [ReserveContact](#) 時，會 AWS Ground Station 拍攝任務設定檔的快照，並設定資源以在整個聯絡生命週期中使用。使用 [UpdateMissionProfile](#) 和 [UpdateConfig](#) APIs 對這些資源的變更不會反映在更新之前預留的聯絡人中。如果您需要將資源變更套用至已排程的聯絡人，您必須先使用 [CancelContact](#) 取消聯絡人，然後使用 [ReserveContact](#) 重新排程聯絡人。

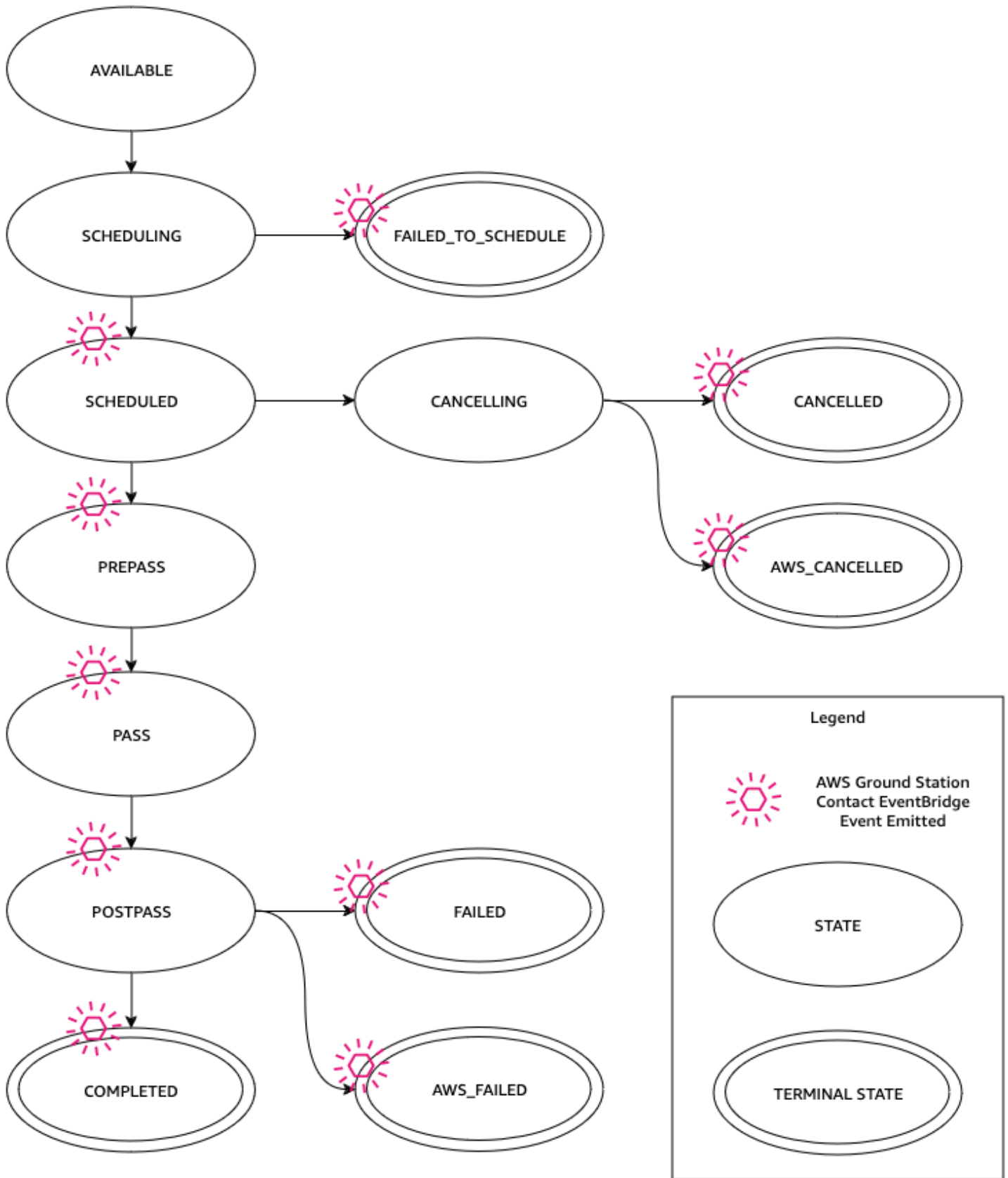
* 取消太接近聯絡時間時，取消聯絡可能會產生費用。如需取消聯絡人的詳細資訊，請參閱：[Ground Station FAQs](#)。

主題

- [了解聯絡生命週期](#)
- [了解聯絡帳單](#)
- [更新聯絡人和聯絡人版本控制](#)

了解聯絡生命週期

了解聯絡生命週期可協助您在使用時自動化和疑難排解各種問題 AWS Ground Station。下圖顯示 AWS Ground Station 聯絡生命週期以及生命週期期間發出的事件橋接事件。請務必注意，COMPLETED、FAILED、FAILED_TO_SCHEDULE、CANCELLED、AWS_CANCELLED 和 AWS_FAILED 是終端狀態。聯絡人不會從終端狀態轉換。如需每個狀態指示的內容，以及使用 [CancelContact](#) 可停止或可取消 [AWS Ground Station 聯絡狀態](#) 的詳細資訊，請參閱。



AWS Ground Station 聯絡狀態

AWS Ground Station 聯絡人的狀態可讓您深入了解該聯絡人在特定時間發生的情況。

聯絡狀態

下表說明聯絡人可以擁有的狀態：

狀態	Description	終端機	可取消	可停止
AVAILABLE	聯絡人可供保留。	否	N/A	N/A
排程	聯絡人正在進行排程。	否	是	否
已排程	已成功排程聯絡人。	否	是	否
FAILED_TO_SCHEDULE	聯絡人無法排程。	是	否	否
預先通過	聯絡人即將開始，並且正在準備資源。	否	是	否
PASS	聯絡人目前正在執行中，且衛星正在與之通訊。	否	否	是
POSTPASS	通訊已完成，且正在使用的資源正在清除。	否	否	否
COMPLETED (已完成)	聯絡人已完成，沒有錯誤。	是	否	否
失敗	由於資源組態發生問題，聯絡人失敗。	是	否	否
AWS_FAILED	由於 AWS Ground Station 服務發生問題，聯絡人失敗。	是	否	否
CANCEL	聯絡人正在取消。	否	否	否
AWS_CANCELED	聯絡人已由 AWS Ground Station 服務取消。天線或網站維護，以及	是	否	否

狀態	Description	終端機	可取消	可停止
	暫時性漂移是可能發生這種情況的範例。			
CANCELLED (已取消)	聯絡人已由您取消。	是	否	否

Note

如需已取消或已停止聯絡人的帳單影響資訊，請參閱 [了解聯絡帳單](#)。

聯絡人資料保留

AWS Ground Station 在提出 [ReserveContact](#) 請求以保留聯絡人後，會保留聯絡人資料 1 年。在 1 年期間之後，會刪除聯絡人資料。

如果您需要保留聯絡資料超過一年，建議您在保留期間到期之前匯出資料。如需如何存取和匯出聯絡人資料的詳細資訊，請參閱：

- [AWS Ground Station API 參考](#)
- [AWS Ground Station CLI 命令參考](#)

了解聯絡帳單

使用時 AWS Ground Station，您只需按每分鐘使用。AWS Ground Station meters 聯絡用量支付天線時間的費用。對於每個聯絡，服務會計算從開始到結束時間的聯絡持續時間，並四捨五入至最接近的分鐘。此計量持續時間會決定該聯絡人的費用。

您的速率取決於兩個主要因素：

- 頻寬 – 為聯絡人預留的頻寬量（窄頻帶或寬頻）
- 地面站位置 – 費率因地面站位置而異

頻寬定義

AWS Ground Station 根據即時頻寬將聯絡人分類為兩個頻寬層：

- 窄頻帶 – 瞬間頻寬小於或等於 40 MHz 的任何聯絡人
- 寬頻 – 瞬間頻寬大於 40 MHz 的任何聯絡人

排程模式

AWS Ground Station 提供兩種排程模式：

- 隨需 – 支付無需長期承諾的天線存取費用
- 預留 – 與隨需相比，提供折扣費率和改善排程，並提供每月承諾。預留分鐘定價適用於承諾在一段時間內每月使用的客戶。

如需帳戶的特定定價資訊或進一步了解預留排程模式，請聯絡您的 AWS 代表。

CancelContact

使用 [CancelContact](#) API 會根據您呼叫時的聯絡狀態而有所不同：

- 聯絡開始之前 - 完全取消聯絡
- 聯絡開始之後和聯絡結束之前 - 停止進行中的聯絡

當您取消聯絡人時，計費取決於您的排程模式和取消時間。如需詳細資訊，請聯絡您的 AWS 代表。

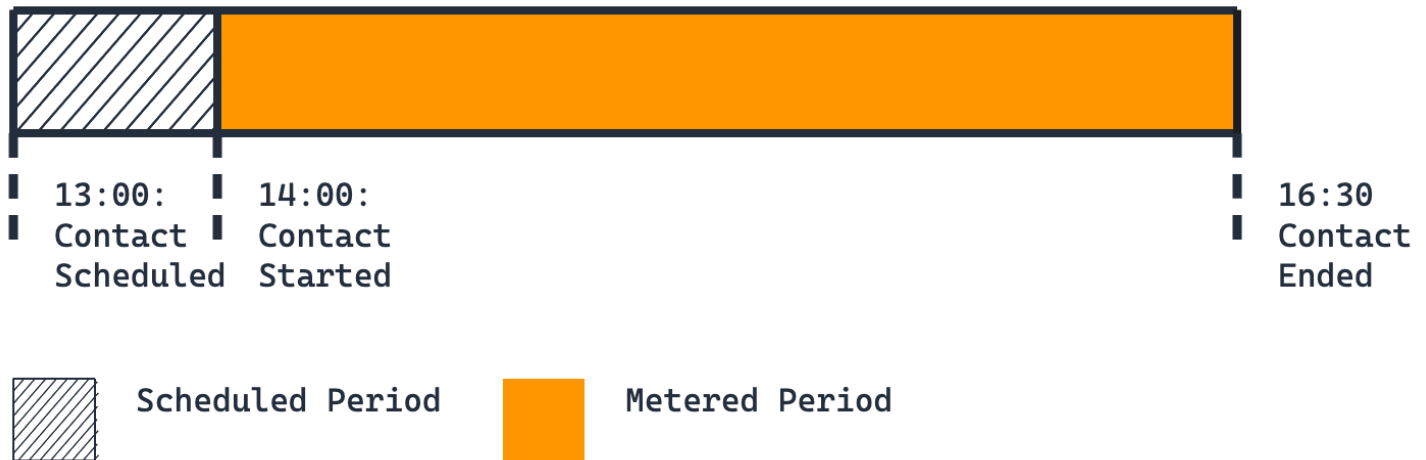
當您停止聯絡人時，您需要支付所執行的聯絡人部分，以及重複聯絡人未涵蓋的剩餘時間的費用。在此內容中重複的聯絡人為：

- 排程在與原始停止聯絡案例相同的地面工作站
- 依與原始停止聯絡人相同的 AWS 帳戶 ID 排程
- 在發出命令後預留，以停止原始聯絡人

下列案例示範此計量的實際運作方式。

案例 1：單一聯絡人

您可以在 Ground Station Anytown 1 上排定 150 分鐘的聯絡，從 14：00 開始，到 16：30 結束。



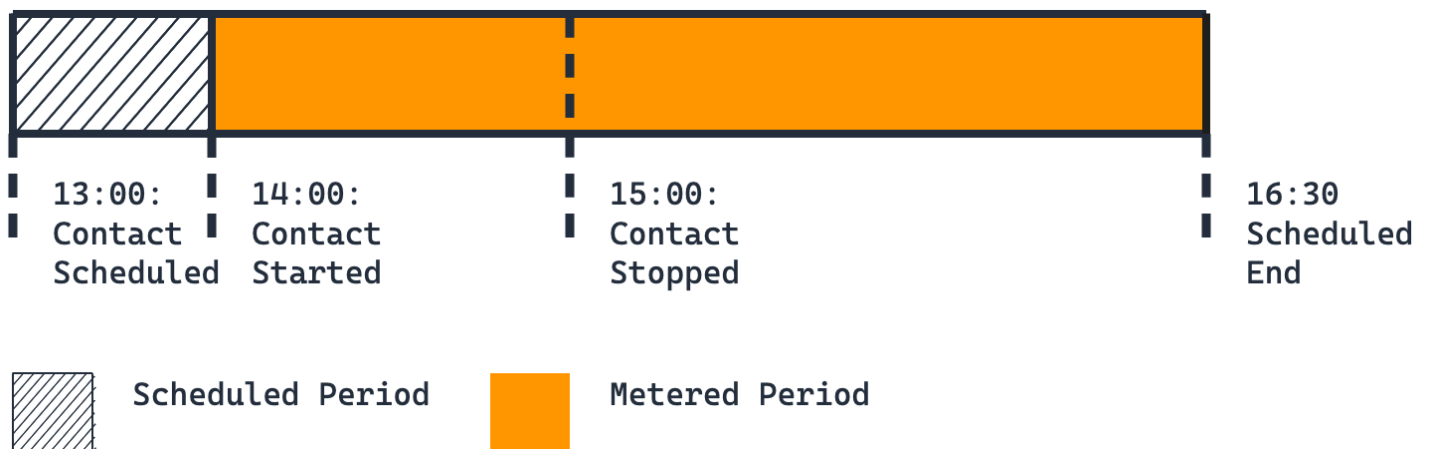
帳單明細：

- 第一次聯絡：150 分鐘（完整持續時間）

您需要支付 150 分鐘的費用。這是基準案例，其中聯絡人執行到排定的完成，而沒有任何停止或取消。

案例 2：單一已停止的聯絡

您可以在 Ground Station Anytown 1 上排定 150 分鐘的聯絡，從 14：00 開始，到 16：30 結束。在 15：00，您可以呼叫 CancelContact API 來停止您的聯絡。



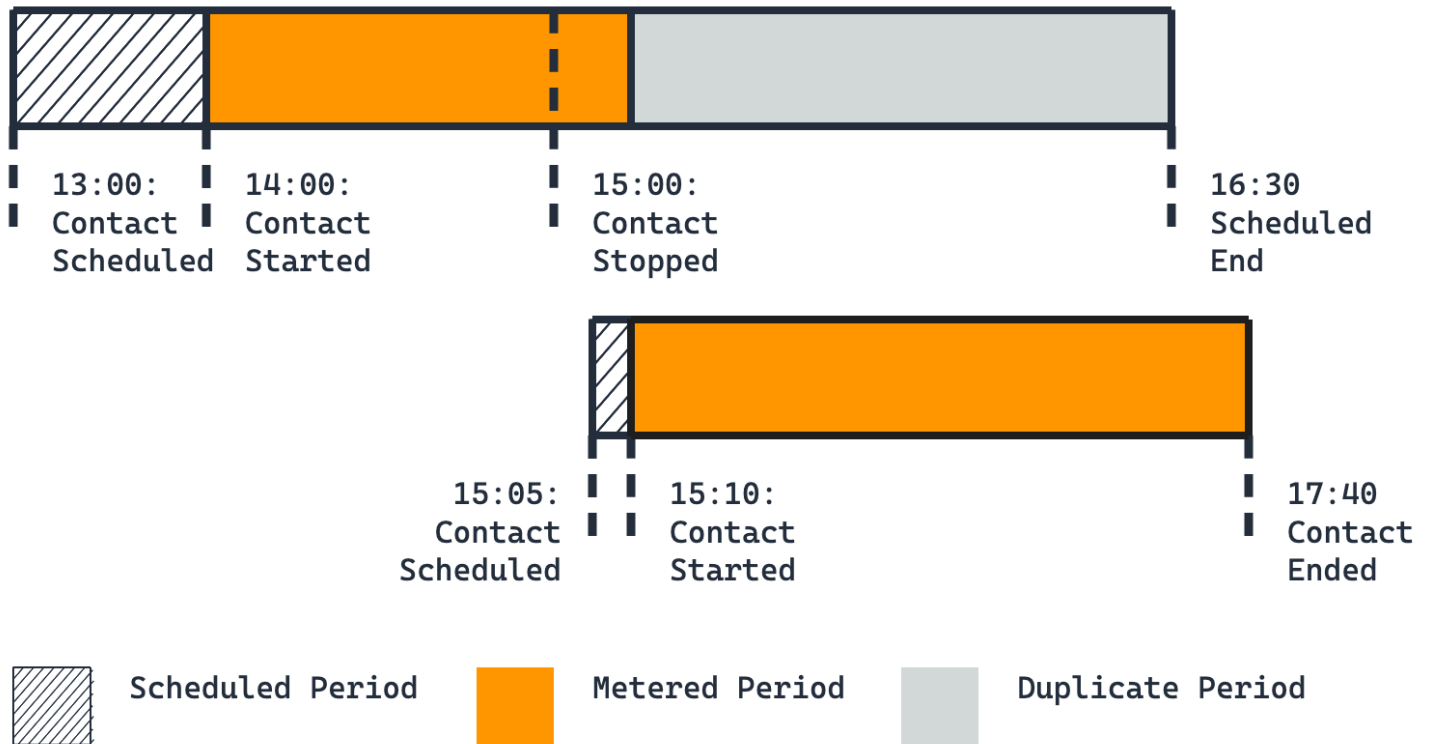
帳單明細：

- 第一次聯絡：150 分鐘（完整原始持續時間）

因為您已停止聯絡人，但未排定任何重複的聯絡人來涵蓋剩餘時間 (15:00-16:30)，所以需支付完整的 150 分鐘費用。當您在沒有排程重複項目的情況下停止聯絡時，您仍需負責整個最初排程的持續時間。

案例 3：單一重複

您可以在 Ground Station Anytown 1 上排定 150 分鐘的聯絡，從 14:00 開始，到 16:30 結束。在 15:00，您可以呼叫 CancelContact API 來停止您的第一個聯絡人。呼叫 CancelContact 後，您可以在同一個 Ground Station 上安排另一個聯絡人，從 15:10 開始，持續 150 分鐘。



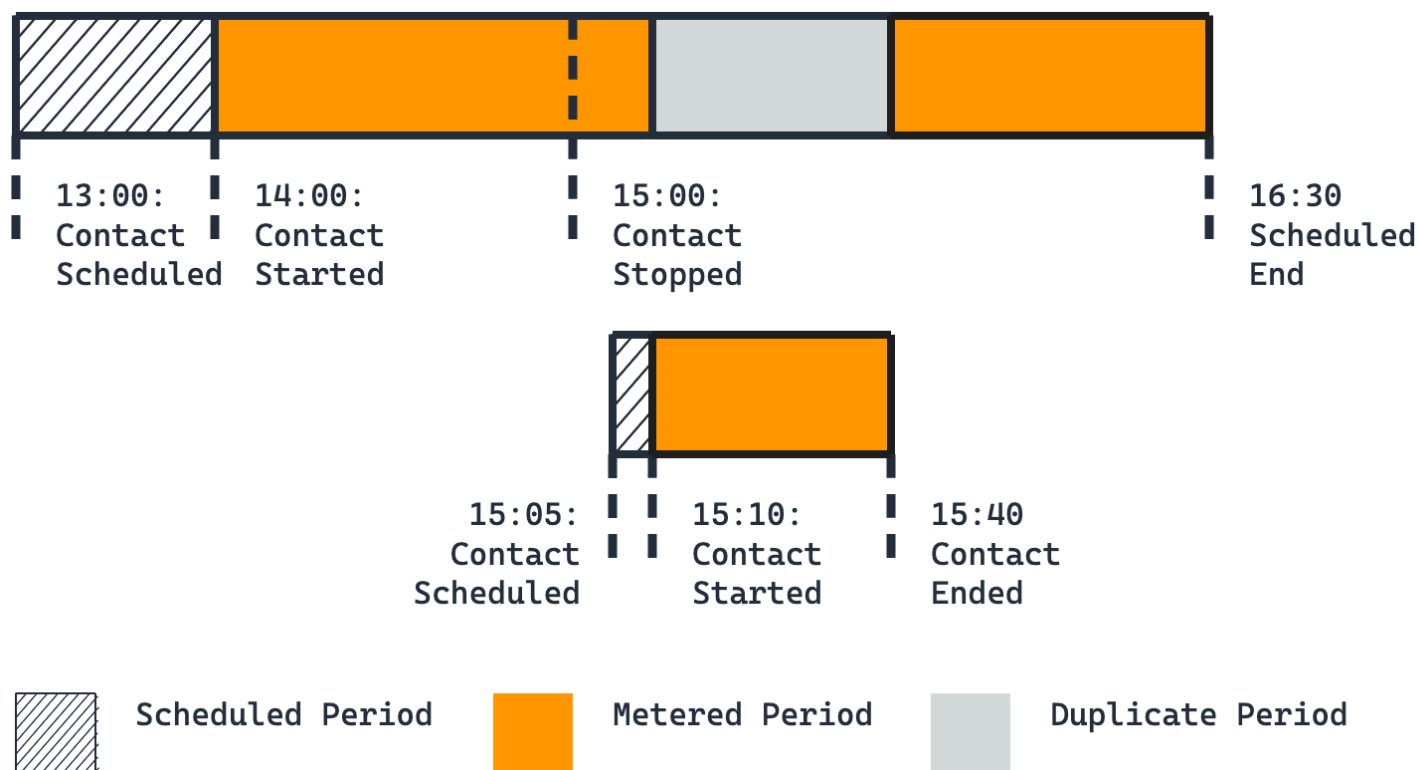
帳單明細：

- 第一次聯絡：70 分鐘（執行 60 分鐘 + 第二次聯絡開始前 10 分鐘的停機時間）
- 第二次聯絡：150 分鐘（完整持續時間）

第二個聯絡案例是重複的，因為您在停止第一個聯絡案例之後已排程。複本涵蓋從 15 : 10 到 16 : 30 的剩餘時間，因此您只需支付第一次聯絡實際執行的時間，加上停止和重新啟動之間的 10 分鐘間隔的費用。

案例 4：簡短重複

您可以在 Ground Station Anytown 1 上排定 150 分鐘的聯絡，從 14 : 00 開始，到 16 : 30 結束。在 15 : 00，您可以呼叫 CancelContact API 來停止您的第一個聯絡人。呼叫 CancelContact 後，您可以在相同的 Ground Station 上安排 30 分鐘的聯絡，時間從 15 : 10 開始。



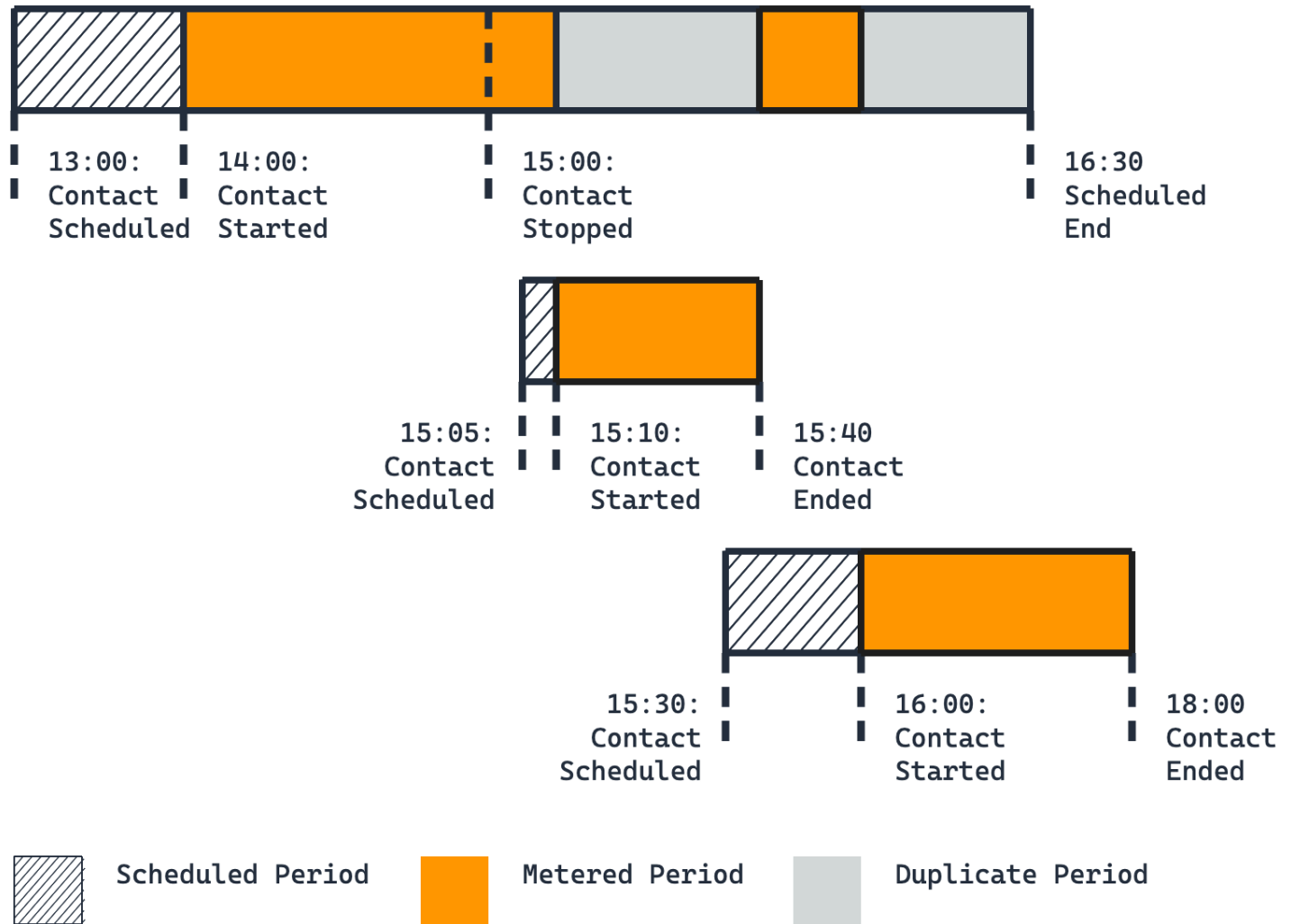
帳單明細：

- 第一次聯絡：120 分鐘（執行 60 分鐘 + 第二次聯絡開始前 10 分鐘的停機時間 + 重複未涵蓋的剩餘 50 分鐘時間）
- 第二次聯絡：30 分鐘（完整持續時間）

重複的聯絡人僅涵蓋您停止第一次聯絡後剩餘 90 分鐘的 30 分鐘 (15 : 10-15 : 40)。在重複項目開始之前的 10 分鐘間隙和重複項目結束之後的 50 分鐘未覆蓋時間 (15 : 40-16 : 30)，都會向您收取費用。

案例 5：多個重複項目

您可以在 Ground Station Anytown 1 上排定 150 分鐘的聯絡，從 14：00 開始，到 16：30 結束。在 15：00，您可以呼叫 CancelContact API 來停止您的第一個聯絡人。呼叫 CancelContact 後，您可以在相同的 Ground Station 上安排 30 分鐘的聯絡，時間從 15：10 開始。稍後，您會在 15：30 排定從 16：00 開始的另一個聯絡人 120 分鐘。



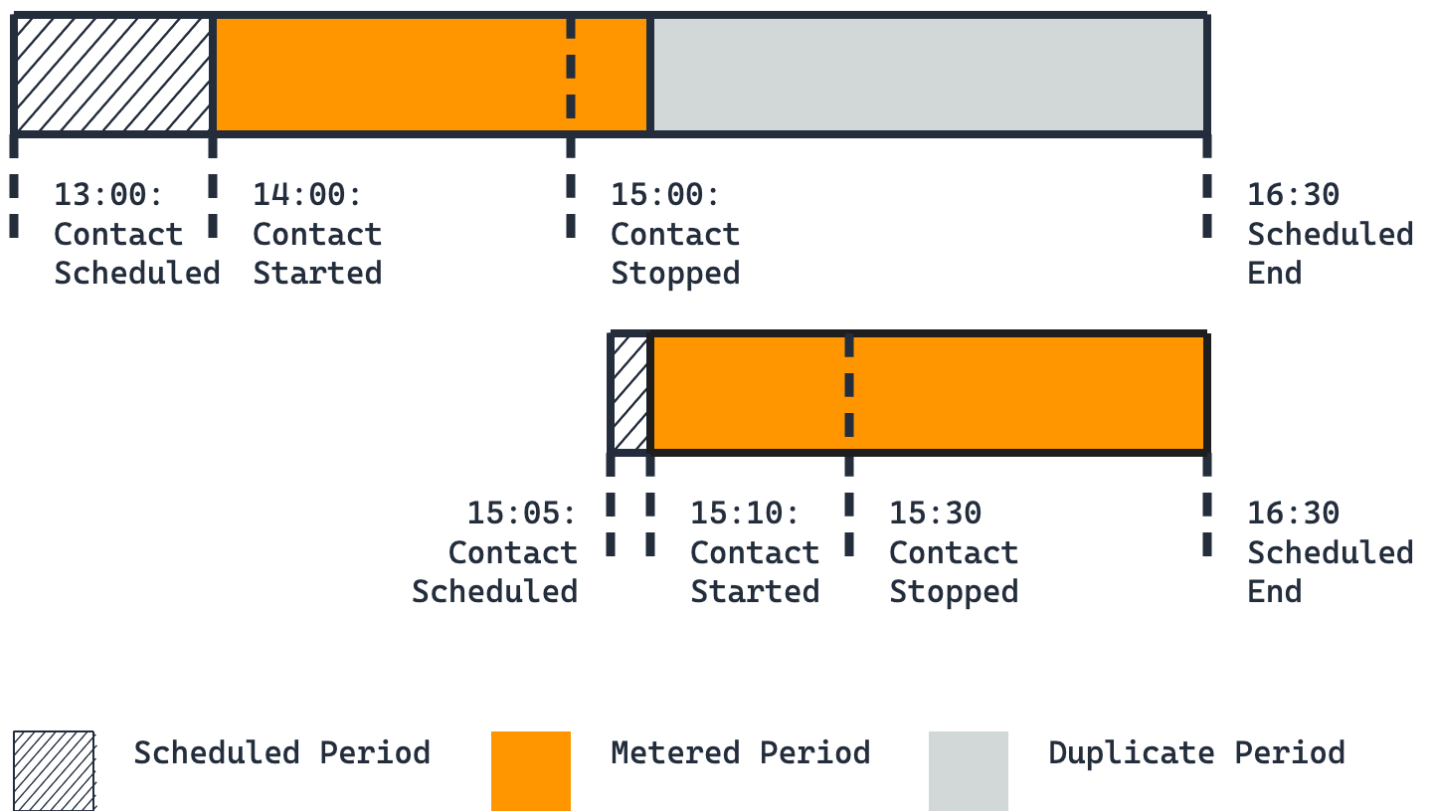
帳單明細：

- 第一次聯絡：90 分鐘（執行 60 分鐘 + 第二次聯絡開始之前 10 分鐘的停機時間 + 第二次與第三次聯絡之間 20 分鐘的停機時間）
- 第二次聯絡：30 分鐘（完整持續時間）
- 第三次聯絡：120 分鐘（完整持續時間）

第二個和第三個聯絡案例都會計為重複，因為您在停止第一個聯絡案例後排定了這些聯絡案例。不過，仍會向您收取聯絡之間的差距：第一個停止 (15 : 00) 和第二個開始 (15 : 10) 之間的 10 分鐘，以及第二個結束 (15 : 40) 和第三個開始 (16 : 00) 之間的 20 分鐘。

案例 6：多次停止

您可以在 Ground Station Anytown 1 上排定 150 分鐘的聯絡，從 14 : 00 開始，到 16 : 30 結束。在 15 : 00，您可以呼叫 CancelContact API 來停止您的第一個聯絡人。呼叫 CancelContact 後，您可以在 Ground Station Anytown 1 上安排 80 分鐘的聯絡，時間從 15 : 10 開始，到 16 : 30 結束。在 15 : 30，您再次呼叫 CancelContact API，停止重複的聯絡。



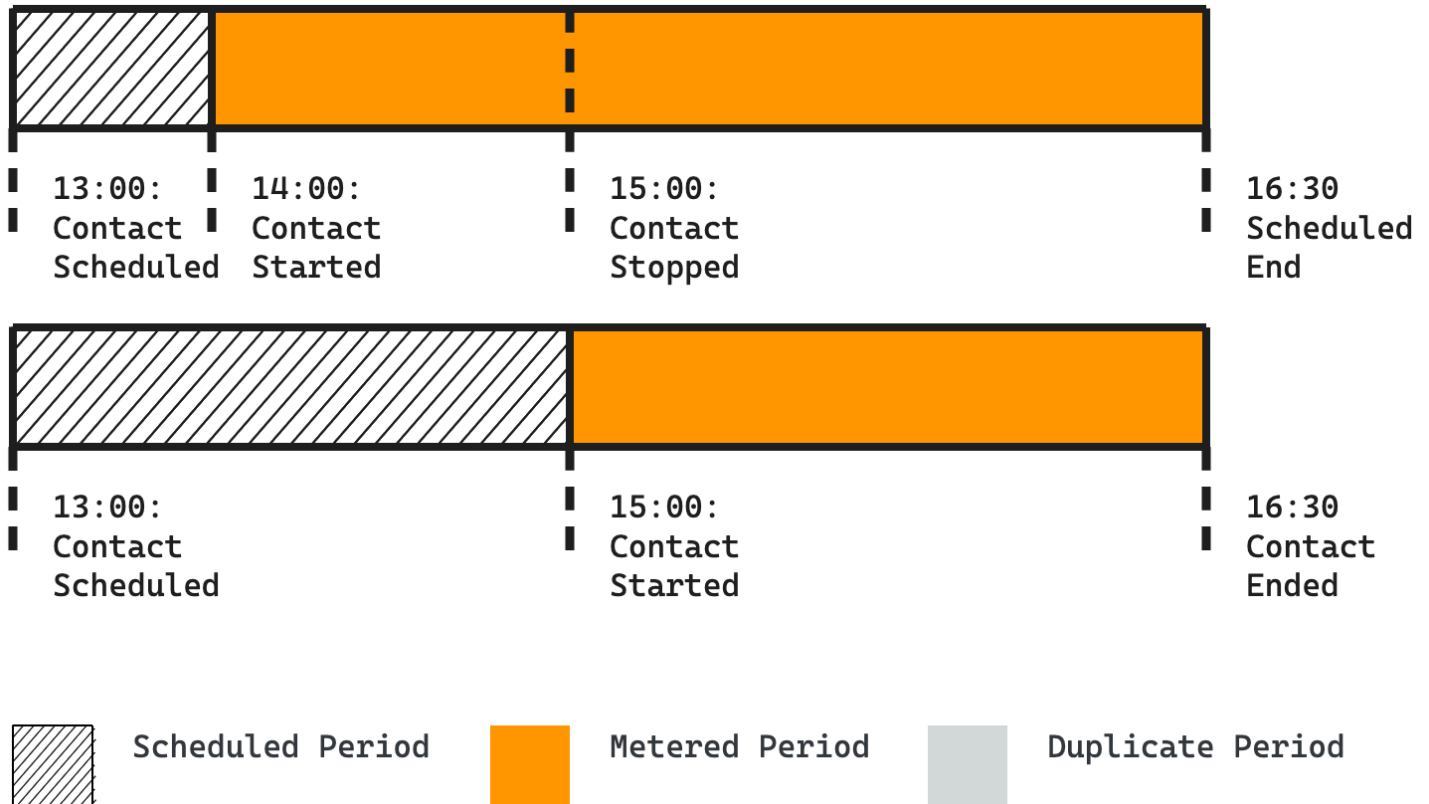
帳單明細：

- 第一次聯絡：70 分鐘（執行 60 分鐘 + 第二次聯絡開始前 10 分鐘的停機時間）
- 第二次聯絡：80 分鐘（完整原始持續時間）

因為您在 15 : 30 停止聯絡，導致 60 分鐘的原始排程時間 (15 : 30-16 : 30) 未填滿，所以會針對第二個聯絡案例的完整 80 分鐘持續時間計費。除非您安排另一個重複的聯絡案例來涵蓋剩餘的時間，否則您必須負責任何已停止聯絡案例的整個持續時間。

案例 7：沒有重複的多天線地面站

在 13:00，您可以在 Ground Station Anytown 1 上排程兩個聯絡人。第一個是 150 分鐘的聯絡，從 14:00 開始，到 16:30 結束。第二個是 90 分鐘的聯絡，從 15:00 開始，到 16:30 結束。在 15:00，您可以呼叫 CancelContact API 來停止您的第一個聯絡人。Ground Station Anytown 1 是一個多天線地面站，允許兩個聯絡人同時執行。



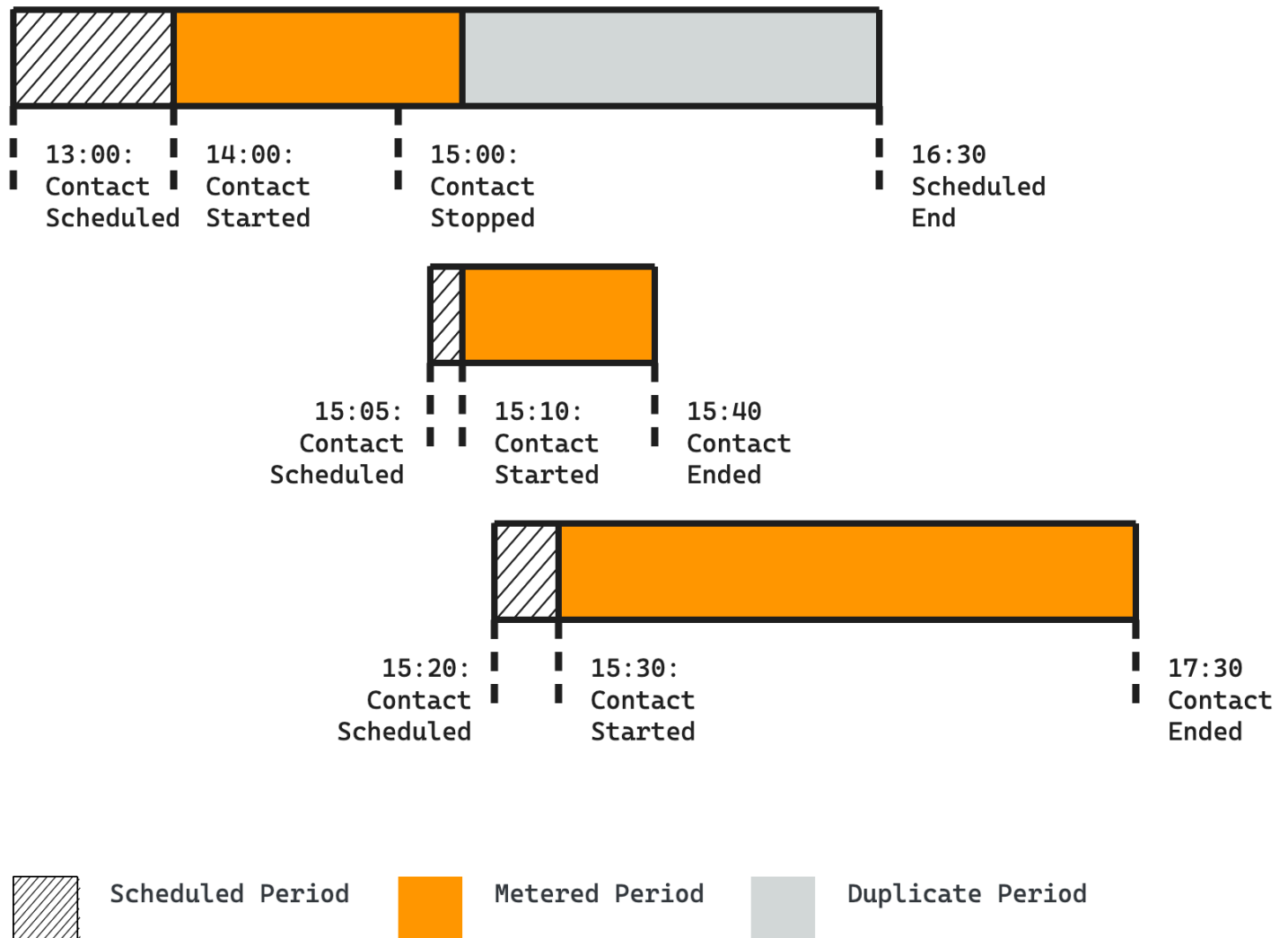
帳單明細：

- 第一次聯絡：150 分鐘（完整原始持續時間）
- 第二次聯絡：90 分鐘（完整持續時間）

雖然第二個聯絡案例與第一個聯絡案例的已停止部分重疊，但不會視為重複。第二個聯絡案例不符合重複項目的第一個條件：已排定在 13:00，再於 15:00 停止第一個聯絡案例。由於這不是重複的，無論您何時停止，都會向您收取第一次聯絡的完整原始持續時間的費用。

案例 8：具有重複聯絡人的多天線地面站

您可以在 Ground Station Anytown 1 上排定 150 分鐘的聯絡，從 14：00 開始，到 16：30 結束。在 15：00，您可以呼叫 CancelContact API 來停止您的第一個聯絡人。呼叫 CancelContact 之後，您可以在 Ground Station Anytown 1 上安排 30 分鐘的聯絡，從 15：10 開始，到 15：40 結束。稍後，您會在 Ground Station Anytown 1 上安排另外 90 分鐘的聯絡，從 15：30 開始，到 17：00 結束。Ground Station Anytown 1 是一個多天線地面站，允許兩個重複的聯絡人同時執行並重疊時間。



帳單明細：

- 第一次聯絡：70 分鐘（執行 60 分鐘 + 第二次聯絡開始前 10 分鐘的停機時間）
- 第二次聯絡：30 分鐘（完整持續時間）
- 第三次聯絡：90 分鐘（完整持續時間）

第二個和第三個聯絡案例都會計為重複，因為您在停止第一個聯絡案例後排定了這些聯絡案例。停止第一個聯絡 (15 : 00) 和開始第二個聯絡 (15 : 10) 之間的 10 分鐘間隔代表針對原始聯絡向您收費的停機時間。

更新聯絡人和聯絡人版本控制

AWS Ground Station 支援更新具有 SCHEDULED、PREPASS 或 聯絡人狀態的 PASS 聯絡人。???您可以使用 [UpdateContact](#) API 來指定聯絡人的暫時性覆寫，包括方位/提升、OEM 或 TLE 追蹤資料，而無需取消和重新排程。這適用於需要在聯絡期間將天線重新任務到不同衛星的地理同步 (GEO) 衛星操作，或需要調整方向的啟動和早期操作 (LEOPs)。

每次您預留或更新聯絡人時，AWS Ground Station 都會建立新的聯絡人版本。聯絡版本提供對聯絡所做的變更歷史記錄，並可讓您追蹤每次更新的狀態。

聯絡版本控制的運作方式

當您呼叫 [ReserveContact](#) 時，會 AWS Ground Station 建立第一個版本的聯絡人 (第 1 版)，並在回應 `versionId` 中傳回。每次後續呼叫 [UpdateContact](#) 時，都會以遞增的版本編號建立新的版本。

[DescribeContact](#) API 會傳回目前的 ACTIVE [聯絡版本](#)，包括回應 `version` 欄位中的版本資訊。[ListContacts](#) API 也包含每個聯絡人的版本資訊。

若要檢視特定版本的聯絡人，請使用 [DescribeContactVersion](#) API。若要列出聯絡人的所有版本，請使用 [ListContactVersions](#) API。

更新聯絡人

當聯絡人處於排程、PREPASS 或 PASS 狀態時，您可以呼叫 [UpdateContact](#)。API 接受下列參數：

- `contactId` — 要更新的聯絡人識別符。
- `clientToken` — 冪等字符，可確保僅處理請求一次。如果您使用相同的用戶端字符重試請求，會 AWS Ground Station 傳回原始回應，而不會再次執行更新。如果未提供用戶端字符，則 AWS SDKs 會自動為您產生用戶端字符。
- `trackingOverrides` — 聯絡人的新追蹤組態。這包括程式追蹤設定 (方位/上升、TLE 或 OEM ephemeris)。
- `satelliteArn` — 聯絡人的衛星 ARN。變更目標衛星與程式軌跡設定時，請提供新衛星的 ARN。只有核准方位角/上升指向角的客戶，才能將此值設定為 `null`。所有其他客戶必須包含聯絡人的衛星 ARN。

⚠ Important

UpdateContact API 會套用請求中的所有參數。省略或明確設定為 null 的任何參數都會視為清除該值的請求，而不會將其保持不變。例如，如果您提供 trackingOverrides 但省略 satelliteArn，則會清除衛星 ARN。請務必在每個更新請求中包含所有所需的值。

您可以在聯絡期間透過提供新的 satelliteArn 以及對應的 trackingOverrides 來變更目標衛星 trackingOverrides。新的衛星必須在聯絡期間從地面工作站可見，因為聯絡開始和結束時間不會因此 API 而變更。新衛星也必須加入地面站點，並擁有任務設定檔所需的授權。聯絡人的任務描述檔無法變更，因此只有在兩個衛星都使用相同的任務描述檔時，切換衛星才適用。

⚠ Important

UpdateContact API 不支援變更聯絡人的開始時間、結束時間或任務設定檔。若要變更這些值，請取消聯絡人並保留新的聯絡人。UpdateContact API 旨在重新排序天線指向組態，例如在衛星之間切換或更新暫時性資料。

⚠ Important

UpdateContact API 不支援具有使用 [天線下行解調解碼組態](#) 組態之任務描述檔的聯絡人。若要變更這些聯絡人的組態，請取消聯絡人並保留新的聯絡人。

UpdateContact API 會傳回 contactId 和新的 versionId。更新會以非同步方式處理。使用 [DescribeContactVersion](#) 檢查更新的狀態。AWS SDKs 和 AWS Command Line Interface 提供輪詢的 ContactUpdated 等待程式，直到版本達到 ACTIVE 或 FAILED_TO_UPDATE 狀態為止。

ℹ Note

一次只能進行一個更新。如果最新的聯絡版本處於 UPDATING 狀態，API 會傳回 ConflictException。等待目前的更新達到 ACTIVE 或 FAILED_TO_UPDATE 狀態，再提交另一個更新。

聯絡版本狀態

每個聯絡版本都有下列其中一種狀態：

狀態	Description
UPDATING	正在將 版本套用至聯絡人。更新已提交並由 處理 AWS Ground Station。
ACTIVE	版本是聯絡人目前作用中的組態。地面工作站正在使用此版本的設定。
SUPERSEDED (已取代)	版本先前已啟用，但已被較新版本取代。
FAILED_TO_UPDATE	無法套用更新。聯絡人會還原至先前作用中的版本。如需詳細資訊，請參閱 <code>failureCodes</code> 和 <code>failureMessage</code> 欄位。

程式碼範例

下列範例示範如何搭配適用於 Python APIs。AWS

範例：更新聯絡人

下列範例會使用新的追蹤覆寫更新聯絡人，並等待使用 Boto3 `ContactUpdated` 等待程式完成更新。

```
import boto3
import uuid

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

# The contact ID of an existing scheduled contact to update
contact_id = "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"

# Generate a unique client token for idempotency.
# If you retry the same request with the same client token,
# the API returns the same response without creating a duplicate version.
client_token = str(uuid.uuid4())

# Update the contact to use a different TLE ephemeris for tracking.
# The UpdateContact API applies all parameters in the request.
```

```
# Any parameter set to null is treated as a request to clear that value,
# not to leave it unchanged. Include all desired values in each request.
print(f"Updating contact {contact_id}...")

update_response = ground_station_client.update_contact(
    contactId=contact_id,
    clientToken=client_token,
    satelliteArn="arn:aws:groundstation::111122223333:satellite/a88611b0-f755-404e-
b60d-57d8aEXAMPLE",
    trackingOverrides={
        "programTrackSettings": {
            "tle": {"ephemerisId": "b2c3d4e5-6789-01ab-cdef-EXAMPLE22222"}
        }
    },
)

contact_id = update_response["contactId"]
version_id = update_response["versionId"]
print(f"Update submitted. Contact: {contact_id}, Version: {version_id}")

# Wait for the update to complete using the ContactUpdated waiter.
# The waiter polls DescribeContactVersion until the version reaches
# ACTIVE (success) or FAILED_TO_UPDATE (failure) status.
# The waiter raises WaiterError if the version reaches FAILED_TO_UPDATE
# or if MaxAttempts is exceeded, so we use try/except to handle both cases.
print("Waiting for update to complete...")

from botocore.exceptions import WaiterError

waiter = ground_station_client.get_waiter("contact_updated")

try:
    waiter.wait(
        contactId=contact_id,
        versionId=version_id,
        WaiterConfig={
            "Delay": 5,
            "MaxAttempts": 180,
        },
    )
    print(f"Contact updated successfully. Version {version_id} is now active.")
except WaiterError as e:
    # WaiterError is raised when the version reaches FAILED_TO_UPDATE
    # or when MaxAttempts is exceeded. Retrieve the current version to inspect.
```

```
version_response = ground_station_client.describe_contact_version(
    contactId=contact_id,
    versionId=version_id,
)
version_status = version_response["version"]["status"]
if version_status == "FAILED_TO_UPDATE":
    failure_codes = version_response["version"].get("failureCodes", [])
    failure_message = version_response["version"].get("failureMessage", "")
    print(f"Update failed. Codes: {failure_codes}, Message: {failure_message}")
else:
    print(f"Waiter timed out. Current version status: {version_status}. Error:
{e}")
```

範例：描述聯絡版本

下列範例會擷取特定聯絡版本的詳細資訊，包括其狀態、組態和任何失敗資訊。

```
import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

contact_id = "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"
version_id = 2

# Describe a specific version of a contact.
# Use this API to check the status of an update or to view the
# configuration that was active at a specific point in time.
print(f"Describing version {version_id} of contact {contact_id}...")

response = ground_station_client.describe_contact_version(
    contactId=contact_id,
    versionId=version_id,
)

# Display version details
version = response["version"]
print(f"Version ID: {version['versionId']}")
print(f"Status: {version['status']}")
print(f"Created: {version.get('created', 'N/A')}")

if version.get("activated"):
    print(f"Activated: {version['activated']}")
```

```

if version.get("superseded"):
    print(f"Superseded: {version['superseded']}")

# Display contact details for this version
print(f"\nContact ID: {response['contactId']}")
print(f"Contact Status: {response['contactStatus']}")
print(f"Ground Station: {response['groundStation']}")
print(f"Start Time: {response['startTime']}")
print(f"End Time: {response['endTime']}")

if response.get("satelliteArn"):
    print(f"Satellite ARN: {response['satelliteArn']}")

if response.get("trackingOverrides"):
    print(f"Tracking Overrides: {response['trackingOverrides']}")

# Check for failure details if the version failed to update
if version["status"] == "FAILED_TO_UPDATE":
    failure_codes = version.get("failureCodes", [])
    failure_message = version.get("failureMessage", "")
    print(f"\nFailure Codes: {failure_codes}")
    print(f"Failure Message: {failure_message}")

```

範例：列出聯絡人版本

下列範例列出聯絡人的所有版本，以檢視變更的完整歷史記錄，使用分頁處理大型結果集。

```

import boto3

# Create AWS Ground Station client
ground_station_client = boto3.client("groundstation")

contact_id = "a1b2c3d4-5678-90ab-cdef-EXAMPLE11111"

# List all versions of a contact to view the full history of changes.
# Results are paginated. Use the nextToken to retrieve additional pages.
print(f"Listing versions for contact {contact_id}...")

paginator = ground_station_client.get_paginator("list_contact_versions")
page_iterator = paginator.paginate(
    contactId=contact_id,
    PaginationConfig={

```

```
        "MaxItems": 100,
        "PageSize": 20,
    },
)

for page in page_iterator:
    for version in page["contactVersionsList"]:
        version_id = version["versionId"]
        status = version["status"]
        created = version.get("created", "N/A")

        print(f" Version {version_id}: status={status}, created={created}")

        if version.get("activated"):
            print(f"    Activated: {version['activated']}")

        if version.get("superseded"):
            print(f"    Superseded: {version['superseded']}")

        if status == "FAILED_TO_UPDATE":
            failure_codes = version.get("failureCodes", [])
            failure_message = version.get("failureMessage", "")
            print(f"    Failure Codes: {failure_codes}")
            print(f"    Failure Message: {failure_message}")

        if status == "UPDATING":
            print(f"    Update is currently in progress.")
```

考量事項

- 在引入聯絡版本控制功能之前建立的聯絡沒有版本資訊。為這些聯絡人呼叫 [DescribeContactVersion](#) 或 [ListContactVersions](#) 會傳回 `ResourceNotFoundException`。
- 當您更新進行中聯絡的追蹤覆寫時，天線會調整為新的指向組態時，會有短暫的轉換期間。在此期間，訊號接收或傳輸可能會中斷。
- 無法刪除聯絡版本。使用 [ListContactVersions](#) 來檢視聯絡人變更的完整歷史記錄。
- `UpdateContact` API 只能從聯絡人的排程區域呼叫。

使用 AWS Ground Station 數位分身功能

的數位分身功能 AWS Ground Station 為您提供環境，您可以在其中測試和整合衛星任務管理和命令和控制軟體。數位分身功能可讓您測試排程、驗證組態和適當的錯誤處理，而無需使用生產天線容量。測試與數位分身功能的 AWS Ground Station 整合可讓您對系統能夠順暢地管理衛星操作更具信心。它還可讓您測試 AWS Ground Station APIs 而無需使用生產容量或需要頻譜授權。

若要開始使用，請遵循 [加入衛星](#)，請求加入數位分身功能。一旦您的衛星加入數位分身功能，您就可以針對數位分身地面站點排程聯絡。您可以透過 AWS SDK [ListGroundStations](#) 回應擷取您有權存取的地面工作站清單。數位分身地面站是 中列出的地面站的確切副本 [AWS Ground Station 位置](#)，字首為「數位分身」的地面站名稱。這包括其天線功能和中繼資料，包括但不限於網站遮罩和實際 GPS 座標。目前，數位分身功能不支援 中所述的資料交付 [使用資料流程](#)。

加入後，數位分身功能會發出與生產服務相同的 Amazon EventBridge 事件和 API 回應，如 中所述 [AWS Ground Station 使用 事件自動化](#)。這些事件可讓您微調組態和資料流程端點群組。

AWS Ground Station 專用天線

AWS Ground Station 專用天線是可代表您管理的自訂天線系統 AWS。與其他客戶共用天線時間的公有 AWS Ground Station 天線不同，專用天線可讓您專用存取根據規格建置的天線。您可以有一或多個專用天線。專用天線已連線至全球 AWS 網路，您可以使用與公有天線搭配使用 AWS Ground Station APIs 和工作流程與其互動，並增強天線使用率的可見性。

什麼是專用天線

專用天線是一種實體天線系統，專為您的組織量身打造並由完整管理 AWS。您可以有一或多個專用天線。專用天線與公有 AWS Ground Station 天線的不同之處如下：

- 自訂建置 — 專用天線是根據您的規格建置。專用天線的功能不限於中所述的公有天線功能 [AWS Ground Station 站點功能](#)。
- 彈性位置 — 專用天線不限於現有的 AWS Ground Station 天線位置。專用天線可以連接到任何現有 AWS 區域，包括 AWS Ground Station 目前不可用的區域。您可以使用 AWS 來判斷符合您需求的位置和區域。
- 專用存取 — 您可以專用存取天線，而不是透過每次聯絡與其他 AWS Ground Station 客戶共用天線時間。

包括專用天線在內的所有 AWS Ground Station 天線都由完全管理 AWS。這包括維護和全球 AWS 網路的連線。您可以使用與公有天線搭配使用 AWS Ground Station APIs 和工作流程，與專用天線互動。您可以以相同的方式排程聯絡人、設定任務設定檔和交付資料。

專用天線可以由多個 AWS 帳戶共用，其中持有專用天線合約的客戶選擇要加入的帳戶。每個加入的帳戶可以獨立排程天線上的聯絡人，並且可以查看共用天線之所有帳戶的保留。

若要進一步了解專用天線或開始使用，AWS 支援請透過聯絡 [AWS Support Center Console](#)。

增強的保留可見性

當您針對專用天線使用 [ListGroundStationReservations](#) API 時，您會看到公有天線上無法使用的其他資訊。下表摘要說明專用天線的 [ListGroundStationReservations](#) API 與公有 AWS Ground Station 天線相比的行為差異。

Behavior (行為)	專用天線	公有天線
您自己的聯絡人	顯示完整詳細資訊，包括 contactId	顯示完整詳細資訊，包括 contactId
其他帳戶的聯絡人	僅顯示時段，不含 contactId	不可見
維護時段	使用 PLANNED 或 可見 UNPLANNED maintenanceType	不可見

維護時段代表天線無法用於衛星通訊的期間。maintenanceType 欄位指出維護是 PLANNED 還是 UNPLANNED。排定計劃外維護時，可能會取消與維護時段重疊的聯絡人 AWS Ground Station。

當您檢視其他共用專用天線 AWS 帳戶的聯絡人時，保留會包含時段和天線資訊，但不 contactId 包含。

Important

增強的保留可見性僅適用於您的專用天線。當您使用公有 AWS Ground Station 天線時，您擁有與其他客戶相同的可見性。您在公有天線上看不到其他帳戶的維護時段或保留。

如需列出保留的詳細資訊，請參閱 [檢視地面站點保留](#)。

相關資源

- [檢視地面站點保留](#)
- [AWS Ground Station 位置](#)
- [AWS Ground Station 站點功能](#)
- AWS Ground Station API 參考中的 [ListGroundStationReservations](#)

了解使用 監控 AWS Ground Station

監控是維護 AWS Ground Station 可靠性、可用性和效能的重要環節。AWS 提供下列監控工具來監看 AWS Ground Station、回報錯誤，並適時採取自動動作。

- Amazon EventBridge Events 提供近乎即時的系統事件串流，描述 AWS 資源的變更。EventBridge Events 可啟用自動事件驅動運算，因為您可以撰寫規則來監控特定事件，並在發生這些事件時觸發 AWS 其他服務中的自動動作。如需 EventBridge Events 的詳細資訊，請參閱 [《Amazon EventBridge Events 使用者指南》](#)。
- AWS CloudTrail 會擷取由您的帳戶發出或代表 AWS 您的帳戶發出的 API 呼叫和相關事件，並將日誌檔案交付至您指定的 Amazon S3 儲存貯體。您可以識別呼叫的使用者和帳戶 AWS、進行呼叫的來源 IP 地址，以及呼叫的時間。如需的詳細資訊 AWS CloudTrail，請參閱 [AWS CloudTrail 《使用者指南》](#)。
- Amazon CloudWatch Metrics 會在使用時擷取排程聯絡人的指標 AWS Ground Station。CloudWatch 指標可讓您根據頻道、極化和衛星 ID 來分析資料，以識別聯絡人中的訊號強度和錯誤。如需詳細資訊，請參閱 [使用 Amazon CloudWatch 指標](#)。
- [AWS 使用者通知](#) 可用來設定交付管道，以取得 AWS Ground Station 事件的通知。當事件符合您指定的規則時，便會收到通知。您可以透過多個管道接收事件的通知，包括電子郵件、[聊天應用程式中的 Amazon Q Developer](#) 聊天通知或 [AWS Console Mobile Application](#) 推送通知。您也可以可以在 AWS 主控台 [通知中心](#) 查看通知。使用者通知 支援彙總，這可以減少您在特定事件期間收到的通知數量。

使用下列主題來監控 AWS Ground Station。

主題

- [AWS Ground Station 使用 事件自動化](#)
- [使用 記錄 AWS Ground Station API 呼叫 AWS CloudTrail](#)
- [使用 Amazon CloudWatch 檢視指標](#)

AWS Ground Station 使用 事件自動化

Note

本文件全文使用「事件」一詞。CloudWatch Events 和 EventBridge 是相同的基礎服務和 API。您可以使用任一服務建置符合傳入事件並將其路由至目標以進行處理的規則。

事件可讓您自動化 AWS 服務，並自動回應系統事件，例如應用程式可用性問題或資源變更。來自 AWS 服務的事件會以近乎即時的方式交付。您可編寫簡單的規則，來指示您在意的事件，以及當事件符合規則時所要自動執行的動作。可自動觸發的一些動作包括下列項目：

- 叫用 AWS Lambda 函數
- 調用 Amazon EC2 執行命令
- 將事件轉傳至 Amazon Kinesis Data Streams
- 啟用 AWS Step Functions 狀態機器
- 通知 Amazon SNS 主題或 Amazon SQS 佇列

搭配使用事件的一些範例 AWS Ground Station 包括：

- 叫用 Lambda 函數，根據事件狀態自動啟動和停止 Amazon EC2 執行個體。
- 每當聯絡人變更狀態時，發佈至 Amazon SNS 主題。這些主題可設定為在聯絡開始或結束時寄出電子郵件通知。

如需詳細資訊，請參閱 [《Amazon EventBridge Events 使用者指南》](#)。

AWS Ground Station 事件類型

Note

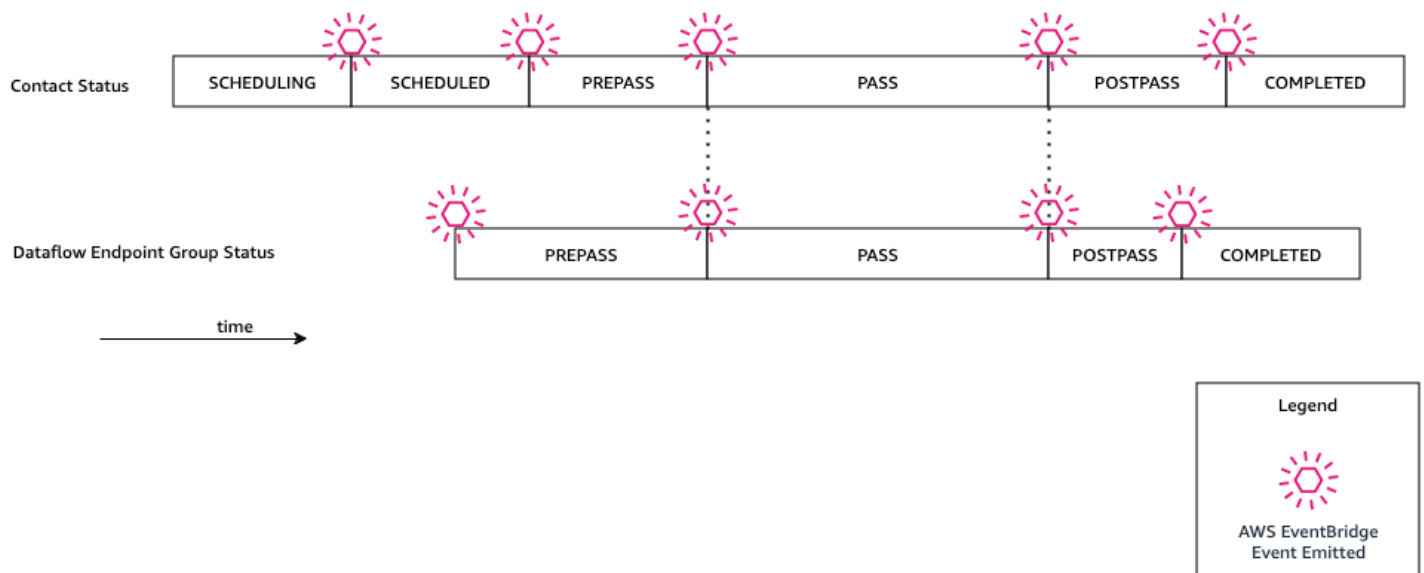
AWS Ground Station 產生的所有事件都有 "aws.groundstation" 做為 "source" 的值。

AWS Ground Station 會發出與狀態變更相關的事件，以支援您自訂自動化的能力。目前，AWS Ground Station 支援聯絡狀態變更事件、資料流程端點群組變更事件和暫時性狀態變更事件。下列各節提供每種類型的詳細資訊。

聯絡事件時間表

AWS Ground Station 當您的聯絡人變更狀態時，會發出事件。如需這些狀態變更以及狀態本身意義的詳細資訊，請參閱 [了解聯絡生命週期](#)。聯絡人中使用的任何資料流程端點群組都有獨立的事件集，也會發出。在相同的時間範圍內，我們也會為您的資料流程端點群組發出事件。當您設定任務描述檔和資料流程端點群組時，您可以設定傳遞前和傳遞後事件的精確時間。

下圖顯示針對名目聯絡人及其相關聯資料流程端點群組發出的狀態和事件。



Ground Station 聯絡狀態變更

如果您想要在即將到來的聯絡變更狀態時執行特定動作，您可以設定規則來自動執行此動作。如果您要接收聯絡狀態變更的相關通知，這種方式就非常有助。如果您想要在收到這些事件時變更，您可以修改任務設定檔的 [contactPrePassDurationSeconds](#) 和 [contactPostPassDurationSeconds](#)。事件會重送至聯絡排程起始的區域。

範例事件提供如下。

```
{
  "version": "0",
  "id": "01234567-0123-0123",
  "account": "123456789012",
  "time": "2019-05-30T17:40:30Z",
  "region": "us-west-2",
  "source": "aws.groundstation",
  "resources": [
    "arn:aws:groundstation:us-west-2:123456789012:contact/11111111-1111-1111-1111-111111111111"
  ],
  "detailType": "Ground Station Contact State Change",
  "detail": {
    "contactId": "11111111-1111-1111-1111-111111111111",
    "groundstationId": "Ground Station 1",
    "missionProfileArn": "arn:aws:groundstation:us-west-2:123456789012:mission-profile/11111111-1111-1111-1111-111111111111",
  }
}
```

```

    "satelliteArn":
      "arn:aws:groundstation::123456789012:satellite/11111111-1111-1111-1111-111111111111",
      "contactStatus": "PASS"
  }
}

```

的可能值 `contactStatus` 在 [中定義 the section called “AWS Ground Station 聯絡狀態”](#)。

Ground Station 資料流程端點群組狀態變更

如果您要在使用資料流程端點群組來接收資料時執行某的動作，您可以設定規則以自動執行此動作。這樣您就可以在回應資料流程端點群組狀態變更時執行不同的動作。如果您想要在收到這些事件時變更，請使用具有不同 [contactPrePassDurationSeconds](#) 和 [contactPostPassDurationSeconds](#) 的資料流程端點群組。此事件將傳送至資料流程端點群組的區域。

以下提供範例。

```

{
  "version": "0",
  "id": "01234567-0123-0123",
  "account": "123456789012",
  "time": "2019-05-30T17:40:30Z",
  "region": "us-west-2",
  "source": "aws.groundstation",
  "resources": [
    "arn:aws:groundstation:us-west-2:123456789012:dataflow-endpoint-group/bad957a8-1d60-4c45-a92a-39febd98921d",
    "arn:aws:groundstation:us-west-2:123456789012:contact/98ddd10f-f2bc-479c-bf7d-55644737fb09",
    "arn:aws:groundstation:us-west-2:123456789012:mission-profile/c513c84c-eb40-4473-88a2-d482648c9234"
  ],
  "detailType": "Ground Station Dataflow Endpoint Group State Change",
  "detail": {
    "dataflowEndpointGroupId": "bad957a8-1d60-4c45-a92a-39febd98921d",
    "groundstationId": "Ground Station 1",
    "contactId": "98ddd10f-f2bc-479c-bf7d-55644737fb09",
    "dataflowEndpointGroupArn": "arn:aws:groundstation:us-west-2:680367718957:dataflow-endpoint-group/bad957a8-1d60-4c45-a92a-39febd98921d",
    "missionProfileArn": "arn:aws:groundstation:us-west-2:123456789012:mission-profile/c513c84c-eb40-4473-88a2-d482648c9234",
    "dataflowEndpointGroupState": "PREPASS"
  }
}

```

```
}
}
```

`dataflowEndpointGroupState` 可能的狀態包括 PREPASS、PASS、POSTPASS 和 COMPLETED。

Ephemeris 事件

Ground Station Ephemeris 狀態變更

如果您想要在暫時性變更狀態時執行動作，您可以設定規則來自動化此動作。這可讓您執行不同的動作，以回應暫時性變化狀態。例如，您可以在 `ephemeris` 已完成驗證且現在為 `時執行 動作ENABLED`。此事件的通知會在 `ephemeris` 上傳時傳送至 區域。

以下提供範例。

```
{
  "id": "7bf73129-1428-4cd3-a780-95db273d1602",
  "detail-type": "Ground Station Ephemeris State Change",
  "source": "aws.groundstation",
  "account": "123456789012",
  "time": "2019-12-03T21:29:54Z",
  "region": "us-west-2",
  "resources": [
    "arn:aws:groundstation::123456789012:satellite/10313191-c9d9-4ecb-a5f2-bc55cab050ec",
    "arn:aws:groundstation::123456789012:ephemeris/111111-cccc-bbbb-a555-bcccca005000"
  ],
  "detail": {
    "ephemerisStatus": "ENABLED",
    "ephemerisId": "111111-cccc-bbbb-a555-bcccca005000",
    "satelliteId": "10313191-c9d9-4ecb-a5f2-bc55cab050ec"
  }
}
```

的可能狀態 `ephemerisStatus` 包括 ENABLED、VALIDATING、INVALID、ERROR、DISABLED、EXPIRED

使用記錄 AWS Ground Station API 呼叫 AWS CloudTrail

AWS Ground Station 已與服務整合 AWS CloudTrail，此服務可提供使用者、角色或 AWS 服務在其中採取之動作的記錄 AWS Ground Station。CloudTrail 會將 AWS Ground Station 的所有 API 呼叫擷取為事件。擷取的呼叫包括來自 AWS Ground Station 主控台的呼叫，以及對 AWS Ground Station API 操作的程式碼呼叫。如果您建立線索，您可以將 CloudTrail 事件持續交付至 Amazon S3 儲存貯體，包括的事件 AWS Ground Station。即使您未設定追蹤，依然可以透過 CloudTrail 主控台中事件歷史記錄檢視最新事件。您可以使用 CloudTrail 所收集的資訊來判斷提出的請求 AWS Ground Station、提出請求的 IP 地址、提出請求的人員、提出請求的時間，以及其他詳細資訊。

若要進一步了解 CloudTrail，請參閱[AWS CloudTrail 《使用者指南》](#)。

AWS Ground Station CloudTrail 中的資訊

當您建立 AWS 帳戶時，會在您的帳戶上啟用 CloudTrail。當活動在中發生時 AWS Ground Station，該活動會與事件歷史記錄中的其他服務 AWS 事件一起記錄在 CloudTrail 事件中。您可以在 AWS 帳戶中檢視、搜尋和下載最近的事件。如需詳細資訊，請參閱《使用 CloudTrail 事件歷史記錄檢視事件》<https://docs.aws.amazon.com/awsccloudtrail/latest/userguide/view-cloudtrail-events.html>。

若要持續記錄您 AWS 帳戶中的事件，包括的事件 AWS Ground Station，請建立追蹤。線索能讓 CloudTrail 將日誌檔案交付至 Amazon S3 儲存貯體。根據預設，當您在主控台建立追蹤記錄時，追蹤記錄會套用到所有 AWS 區域。線索會記錄 AWS 分割區中所有區域的事件，並將日誌檔案交付至您指定的 Amazon S3 儲存貯體。此外，您可以設定其他 AWS 服務，以進一步分析和處理 CloudTrail 日誌中收集的事件資料。如需詳細資訊，請參閱下列內容：

- [建立追蹤的概觀](#)
- [CloudTrail 支援的服務和整合](#)
- [設定 CloudTrail 的 Amazon SNS 通知](#)
- [從多個區域接收 CloudTrail 日誌檔案](#)，以及[從多個帳戶接收 CloudTrail 日誌檔案](#)

CloudTrail 會記錄所有 AWS Ground Station 動作，並記錄在 [AWS Ground Station API 參考](#)中。例如，對 ReserveContact、CancelContact 和 ListConfigs 動作發出的呼叫會在 CloudTrail 記錄檔案中產生專案。

每一筆事件或日誌專案都會包含產生請求者的資訊。身分資訊可協助您判斷下列事項：

- 是否使用根或 AWS Identity and Access Management (IAM) 使用者登入資料提出請求。
- 提出該請求時，是否使用了特定角色或聯合身分使用者的暫時安全憑證。

- 請求是否由其他 AWS 服務提出。

如需詳細資訊，請參閱 [CloudTrail userIdentity 元素](#)。

了解 AWS Ground Station 日誌檔案項目

追蹤是一種組態，能讓事件以日誌檔案的形式交付到您指定的 Amazon S3 儲存貯體。CloudTrail 日誌檔案包含一或多個日誌專案。一個事件為任何來源提出的單一請求，並包含請求動作、請求的日期和時間、請求參數等資訊。CloudTrail 日誌檔並非依公有 API 呼叫的堆疊追蹤排序，因此不會以任何特定順序出現。

以下範例顯示的是展示 ReserveContact 動作的 CloudTrail 日誌項目。

範例：ReserveContact

```
{
  "eventVersion": "1.05",
  "userIdentity": {
    "type": "IAMUser",
    "principalId": "EX_PRINCIPAL_ID",
    "arn": "arn:aws:sts::123456789012:user/Alice",
    "accountId": "123456789012",
    "accessKeyId": "EXAMPLE_KEY_ID",
    "sessionContext": {
      "attributes": {
        "mfaAuthenticated": "false",
        "creationDate": "2019-05-15T21:11:59Z"
      },
      "sessionIssuer": {
        "type": "Role",
        "principalId": "EX_PRINCIPAL_ID",
        "arn": "arn:aws:iam::123456789012:role/Alice",
        "accountId": "123456789012",
        "userName": "Alice"
      }
    }
  },
  "eventTime": "2019-05-15T21:14:37Z",
  "eventSource": "groundstation.amazonaws.com",
  "eventName": "ReserveContact",
  "awsRegion": "us-east-2",
```

```
"sourceIPAddress": "127.0.0.1",
"userAgent": "Mozilla/5.0 Gecko/20100101 Firefox/123.0",
"requestParameters": {
  "satelliteArn":
"arn:aws:groundstation::123456789012:satellite/11111111-2222-3333-4444-555555555555",
  "groundStation": "Ohio 1",
  "startTime": 1558356107,
  "missionProfileArn": "arn:aws:groundstation:us-east-2:123456789012:mission-
profile/11111111-2222-3333-4444-555555555555",
  "endTime": 1558356886
},
"responseElements": {
  "contactId": "11111111-2222-3333-4444-555555555555"
},
"requestID": "11111111-2222-3333-4444-555555555555",
"eventID": "11111111-2222-3333-4444-555555555555",
"readOnly": false,
"eventType": "AwsApiCall",
"recipientAccountId": "11111111-2222-3333-4444-555555555555"
}
```

使用 Amazon CloudWatch 檢視指標

在聯絡期間，AWS Ground Station 會自動擷取資料並將其傳送至 CloudWatch 進行分析。您可以在 Amazon CloudWatch 主控台中檢視您的資料。如需存取和 CloudWatch 指標的詳細資訊，請參閱[使用 Amazon CloudWatch 指標](#)。

AWS Ground Station 遙測功能也可以用來在聯絡期間接收近乎即時的指標。CloudWatch 指標近乎即時地不可用，且交付可能會延遲。CloudWatch 也會彙總一秒期間的指標，進而降低資料精細程度。遙測功能提供個別指標，並以近乎即時的方式直接交付至 AWS 您的帳戶。如需詳細資訊，請參閱[使用遙測](#)。

Important

AWS Ground Station 會將 CloudWatch 指標發送到與聯絡人的地面工作站位置相關聯的 AWS 區域，而不是排程聯絡人 AWS 的區域。若要檢視聯絡人的指標，您必須在地面工作站區域存取 CloudWatch。如需與每個地面站位置相關聯之 AWS 區域的資訊，請參閱[尋找地面站點位置 AWS 的區域](#)。若要在您排程聯絡人的區域中接收遙測資料，您可以使用 AWS Ground Station 遙測功能。如需詳細資訊，請參閱[使用遙測](#)。

AWS Ground Station 指標和維度

目前提供哪些指標？

下列指標可從 取得 AWS Ground Station。

Note

發出的特定指標取決於正在使用 AWS Ground Station 的功能。根據您的組態，可能只會發出以下指標的子集。

指標	指標維度	Description
AzimuthAngle	Satelliteld	天線的方位角。 真正的北向是 0 度，而東向是 90 度。 單位：度
BitErrorRate	頻道、偏光、Satelliteld	指定位元傳輸次數中位元的錯誤率。位元錯誤是由雜訊、失真或干擾所導致 單位：每單位時間的位元錯誤
BlockErrorRate	頻道、偏光、Satelliteld	指定已接收區塊數中的區塊錯誤率。區塊錯誤是由干擾所導致。 單位：錯誤區塊 / 區塊總數

指標	指標維度	Description
CarrierFrequencyRecovery_Cn0	類別、Config、Satelliteld	每個單位頻寬的載波雜訊密度比率。 單位：decibel-Hertz (dB-Hz)
CarrierFrequencyRecovery_Locked	類別、Config、Satelliteld	當解調器電信業者頻率復原迴圈鎖定時，設為 1，解除鎖定時設為 0。 單位：無單位
CarrierFrequencyRecovery_OffsetFrequency_Hz	類別、Config、Satelliteld	估計訊號中心與理想中心頻率之間的位移。這是由於太空船和天線系統之間的都卜勒偏移和本機振盪器偏移所造成。 單位：hertz (Hz)
ElevationAngle	Satelliteld	天線的高度角度。地平線為 0 度，小數為 90 度。 單位：度

指標	指標維度	Description
Es/N0	頻道、偏光、SatelliteId	每個符號的能量與雜訊功率光譜密度的比率。 單位：分貝 (dB)
ReceivedPower	極化、SatelliteId	解調器/解碼器中測量得到的訊號強度。 單位：相對於毫秒 (dBm) 的分貝
SymbolTimingRecovery_ErrorVectorMagnitude	類別、Config、SatelliteId	收到的符號和理想星座點之間的錯誤向量大小。 單位：百分比
SymbolTimingRecovery_Locked	類別、Config、SatelliteId	當解調器符號計時復原迴圈鎖定時，設為 1，解除鎖定時設為 0 單位：無單位
SymbolTimingRecovery_OffsetSymbolRate	類別、Config、SatelliteId	估計符號率與理想訊號符號率之間的位移。這是由於太空船和天線系統之間的多卜勒偏移和本機振盪器偏移所造成。 單位：符號/秒

使用哪些維度 AWS Ground Station ？

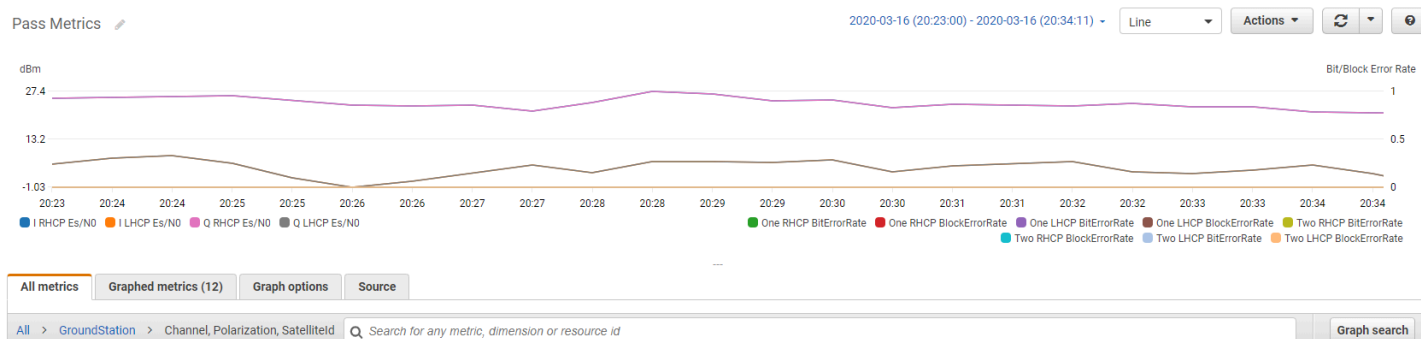
您可以使用下列維度篩選 AWS Ground Station 資料。

維度	Description
Category	解調或解碼。
Channel	每個聯絡的頻道包含一、二、I (同相) 和 Q (正交)。
Config	天線下行示範解碼組態 arn。
Polarization	每個聯絡的極化包含 LHCP (左側圓極化) 或 RHCP (右側圓極化)。
SatelliteId	衛星 ID 包含聯絡衛星的 ARN。

檢視 指標

檢視圖形化指標時，請務必注意彙總視窗會決定如何顯示您的指標。在收到資料之後，聯絡中的每個指標都會持續在 3 小時期間內顯示為每秒資料。CloudWatch Metrics 會將您的資料彙總為經過 3 小時後的每分鐘資料。如果您需要檢視每秒資料量值的指標，建議您在收到資料後 3 小時內檢視您的資料，或將其保留在 CloudWatch 指標之外。如需 CloudWatch 保留的詳細資訊，請參閱 [Amazon CloudWatch 概念 - 指標保留](#)。

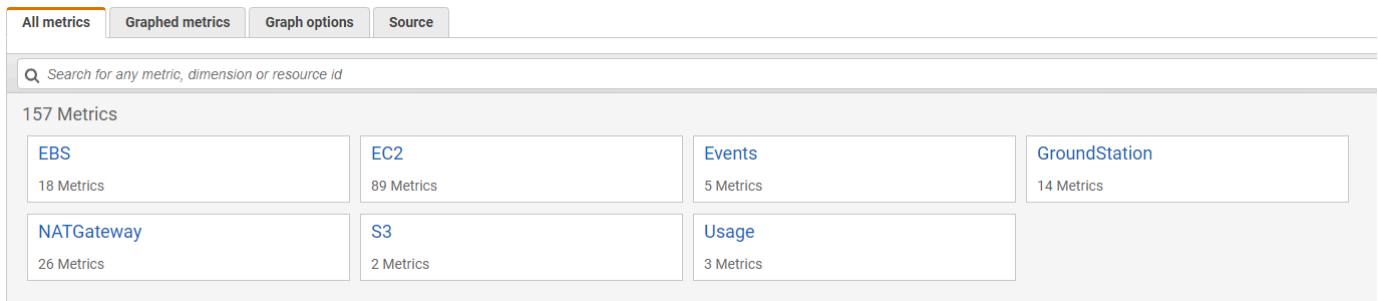
此外，前 60 秒內所擷取的任何資料都不會包含足以產生有意義指標的資訊，而且可能不會顯示。為了檢視有意義的資料，建議您在 60 秒過後再檢視資料。



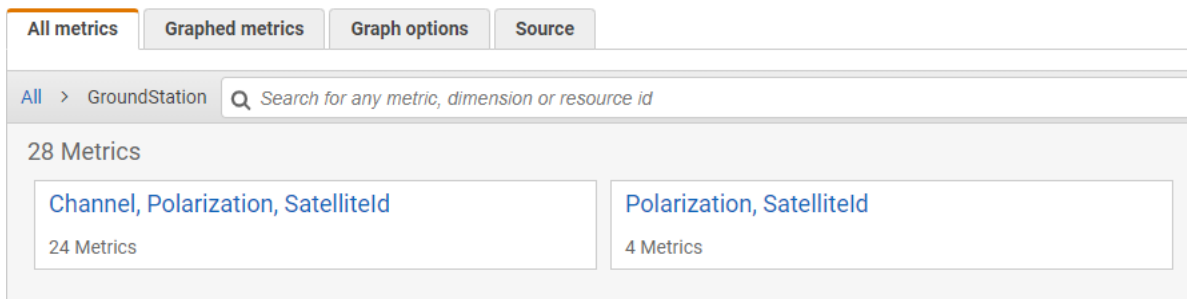
如需在 CloudWatch 中繪製 AWS Ground Station 指標圖形的詳細資訊，請參閱 [繪製指標圖形](#)。

使用 主控台檢視指標

1. 判斷與地面工作站位置相關聯的 AWS 區域。AWS Ground Station 會在與聯絡人地面工作站位置相關聯的區域中發出 CloudWatch 指標。如需地面站點位置及其相關 AWS 區域的清單，請參閱 [尋找地面站點位置 AWS 的區域](#)。
2. 開啟 [CloudWatch 主控台](#)。
3. 在導覽窗格中，選擇 Metrics (指標)。
4. 選取 GroundStation 命名空間。



5. 選取所需的指標維度 (例如，頻道、極化、Satelliteld)。



6. All metrics (所有指標) 索引標籤會顯示命名空間中該維度的所有指標。您可以執行下列動作：
 - a. 若要將資料表排序，請使用直欄標題。
 - b. 若要繪製指標圖形，請選取與指標相關聯的核取方塊。若要選取所有指標，請選取資料表標題列中的核取方塊。
 - c. 若要依資源篩選，請選擇資源 ID，然後選擇 Add to search (新增至搜尋)。
 - d. 若要依指標篩選，請選擇指標名稱，然後選擇 Add to search (新增至搜尋)。

使用 檢視指標 AWS CLI

AWS Ground Station 在與聯絡人的地面工作站位置相關聯的區域中發出 CloudWatch 指標。如需其相關聯 AWS 區域的地面站位置清單，請參閱[尋找地面站點位置 AWS 的區域](#)。將 `ground-station-region-code` 取代為您的地面站位置 AWS 的區域代碼（例如，us-west-2 適用於奧勒岡州 1、夏威夷州 1 或阿拉斯加州 1）。此程序中的所有後續 AWS CLI 命令都必須使用相同的區域。

1. 確定 AWS CLI 已安裝。如需安裝的資訊 AWS CLI，請參閱[安裝 AWS CLI 第 2 版](#)。
2. 識別與地面站位置相關聯的 AWS 區域。
3. 使用 CloudWatch CLI 的 [get-metric-data](#) 方法來產生可修改的檔案，以指定您感興趣的指標，然後用來查詢這些指標。

若要這樣做，請執行下列動作：`aws cloudwatch get-metric-data --region ground-station-region-code --generate-cli-skeleton`。這將產生類似以下的輸出：

```
{
  "MetricDataQueries": [
    {
      "Id": "",
      "MetricStat": {
        "Metric": {
          "Namespace": "",
          "MetricName": "",
          "Dimensions": [
            {
              "Name": "",
              "Value": ""
            }
          ]
        },
        "Period": 0,
        "Stat": "",
        "Unit": "Seconds"
      },
      "Expression": "",
      "Label": "",
      "ReturnData": true,
      "Period": 0,
      "AccountId": ""
    }
  ],
```

```

    "StartTime": "1970-01-01T00:00:00",
    "EndTime": "1970-01-01T00:00:00",
    "NextToken": "",
    "ScanBy": "TimestampDescending",
    "MaxDatapoints": 0,
    "LabelOptions": {
      "Timezone": ""
    }
  }
}

```

4. 透過執行 列出可用的 CloudWatch `aws cloudwatch list-metrics --region ground-station-region-code` 指標。

如果您最近使用過 AWS Ground Station，方法應傳回包含下列項目的輸出：

```

...
{
  "Namespace": "AWS/GroundStation",
  "MetricName": "ReceivedPower",
  "Dimensions": [
    {
      "Name": "Polarization",
      "Value": "LHCP"
    },
    {
      "Name": "SatelliteId",
      "Value": "arn:aws:groundstation::111111111111:satellite/aaaaaaaa-
bbbb-cccc-dddd-eeeeeeeeeeee"
    }
  ]
},
...

```

Note

如果距離上次使用已超過 2 週 AWS Ground Station，則您需要手動檢查 [可用指標的資料表](#)，以尋找指標命名空間中的 `AWS/GroundStation` 指標名稱和維度。如需 CloudWatch 限制的詳細資訊，請參閱：[檢視可用的指標](#)

5. 修改您在步驟 2 中建立的 JSON 檔案，以符合步驟 3 的必要值，例如 `SatelliteId`和`Polarization`您的指標。此外，請務必更新 `StartTime`、和 `EndTime`值，以符合您的聯絡人。例如：

```
{
  "MetricDataQueries": [
    {
      "Id": "receivedPowerExample",
      "MetricStat": {
        "Metric": {
          "Namespace": "AWS/GroundStation",
          "MetricName": "ReceivedPower",
          "Dimensions": [
            {
              "Name": "SatelliteId",
              "Value":
"arn:aws:groundstation::111111111111:satellite/aaaaaaaa-bbbb-cccc-dddd-
eeeeeeeeeeee"
            },
            {
              "Name": "Polarization",
              "Value": "RHCP"
            }
          ]
        },
        "Period": 300,
        "Stat": "Maximum",
        "Unit": "None"
      },
      "Label": "ReceivedPowerExample",
      "ReturnData": true
    }
  ],
  "StartTime": "2024-02-08T00:00:00",
  "EndTime": "2024-04-09T00:00:00"
}
```

Note

AWS Ground Station 根據指標，每 1 到 60 秒發佈一次指標。如果Period欄位的值小於指標的發佈期間，則不會傳回指標。

6. `aws cloudwatch get-metric-data` 使用先前步驟中建立的組態檔案執行。以下提供範例。

```
aws cloudwatch get-metric-data --region ground-station-region-code --cli-input-json
file://<nameOfConfigurationFileCreatedInStep2>.json
```

指標會與您聯絡中的時間戳記一起提供。以下是 AWS Ground Station 指標的輸出範例。

```
{
  "MetricDataResults": [
    {
      "Id": "receivedPowerExample",
      "Label": "ReceivedPowerExample",
      "Timestamps": [
        "2024-04-08T18:35:00+00:00",
        "2024-04-08T18:30:00+00:00",
        "2024-04-08T18:25:00+00:00"
      ],
      "Values": [
        -33.30191555023193,
        -31.46100273132324,
        -32.13915576934814
      ],
      "StatusCode": "Complete"
    }
  ],
  "Messages": []
}
```

中的安全性 AWS Ground Station

的雲端安全性 AWS 是最高優先順序。身為 AWS 客戶，您將受益於資料中心和網路架構，該架構旨在滿足最安全敏感組織的需求。AWS 提供安全特定的工具和功能，協助您達成安全目標。這些工具和功能包含網路安全性、組態管理、存取控制和資料安全性。

使用時 AWS Ground Station，我們建議您遵循業界最佳實務並實作 end-to-end 加密。AWS 提供 API，可讓您整合加密和資料保護。如需 AWS 安全性的詳細資訊，請參閱 [AWS 安全簡介白皮書](#)。

請使用下列主題來了解如何保護您的資源。

主題

- [的 Identity and Access Management AWS Ground Station](#)
- [AWS 的 受管政策 AWS Ground Station](#)
- [使用 Ground Station 的服務連結角色](#)
- [的靜態資料加密 AWS Ground Station](#)
- [傳輸期間的資料加密 AWS Ground Station](#)

的 Identity and Access Management AWS Ground Station

AWS Identity and Access Management (IAM) 是一種 AWS 服務，可協助管理員安全地控制對 AWS 資源的存取。IAM 管理員可控制誰可以進行驗證（登入）和授權（具有許可）來使用 AWS Ground Station 資源。IAM 是您可以免費使用 AWS 服務的。

主題

- [目標對象](#)
- [使用身分驗證](#)
- [使用政策管理存取權](#)
- [AWS Ground Station 如何使用 IAM](#)
- [的身分型政策範例 AWS Ground Station](#)
- [對 AWS Ground Station 身分和存取進行故障診斷](#)

目標對象

使用方式 AWS Identity and Access Management (IAM) 會根據您的角色而有所不同：

- 服務使用者 — 若無法存取某些功能，請向管理員申請所需許可 (請參閱 [對 AWS Ground Station 身分和存取進行故障診斷](#))
- 服務管理員 — 負責設定使用者存取權並提交相關許可請求 (請參閱 [AWS Ground Station 如何使用 IAM](#))
- IAM 管理員 — 撰寫政策以管理存取控制 (請參閱 [的身分型政策範例 AWS Ground Station](#))

使用身分驗證

身分驗證是您 AWS 使用身分憑證登入的方式。您必須以 AWS 帳戶根使用者、IAM 使用者或擔任 IAM 角色身分進行身分驗證。

您可以使用身分來源的登入資料，例如 AWS IAM Identity Center (IAM Identity Center)、單一登入身分驗證或 Google/Facebook 登入資料，以聯合身分的形式登入。如需有關登入的詳細資訊，請參閱《AWS 登入 使用者指南》中的[如何登入您的 AWS 帳戶](#)。

對於程式設計存取，AWS 提供 SDK 和 CLI 以密碼編譯方式簽署請求。如需詳細資訊，請參閱《IAM 使用者指南》中的[API 請求的AWS 第 4 版簽署程序](#)。

AWS 帳戶 根使用者

當您建立時 AWS 帳戶，您會從一個名為 AWS 帳戶 theroot 使用者的登入身分開始，該身分可完整存取所有 AWS 服務和資源。強烈建議不要使用根使用者來執行日常任務。有關需要根使用者憑證的任務，請參閱《IAM 使用者指南》中的[需要根使用者憑證的任務](#)。

聯合身分

最佳實務是要求人類使用者使用聯合身分提供者，以 AWS 服務使用臨時憑證存取。

聯合身分是來自您的企業目錄、Web 身分提供者的使用者，或使用來自身分來源的 AWS 服務憑證存取 Directory Service。聯合身分會擔任角色，而該角色會提供臨時憑證。

若需集中化管理存取權限，建議使用 AWS IAM Identity Center。如需詳細資訊，請參閱 AWS IAM Identity Center 使用者指南中的[什麼是 IAM Identity Center?](#)

IAM 使用者和群組

IAM 使用者https://docs.aws.amazon.com/IAM/latest/UserGuide/id_users.html是一種身分具備單人或應用程式的特定許可權。建議以臨時憑證取代具備長期憑證的 IAM 使用者。如需詳細資訊，請參閱《IAM 使用者指南》中的[要求人類使用者使用聯合身分提供者來 AWS 使用臨時憑證存取](#)。

[IAM 群組](#)會指定 IAM 使用者集合，使管理大量使用者的許可權更加輕鬆。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 使用者的使用案例](#)。

IAM 角色

IAM 角色https://docs.aws.amazon.com/IAM/latest/UserGuide/id_roles.html的身分具有特定許可權，其可以提供臨時憑證。您可以透過[從使用者切換到 IAM 角色（主控台）](#)或呼叫 AWS CLI 或 AWS API 操作來擔任角色。如需詳細資訊，請參閱《IAM 使用者指南》中的[擔任角色的方法](#)。

IAM 角色適用於聯合身分使用者存取、臨時 IAM 使用者許可、跨帳戶存取權與跨服務存取，以及在 Amazon EC2 執行的應用程式。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 中的快帳戶資源存取](#)。

使用政策管理存取權

您可以透過建立政策並將其連接到身分或資源 AWS 來控制 AWS 中的存取。政策定義與身分或資源相關聯的許可。當委託人提出請求時 AWS，會評估這些政策。大多數政策會以 JSON 文件 AWS 形式存放在中。如需進一步了解 JSON 政策文件，請參閱《IAM 使用者指南》中的[JSON 政策概觀](#)。

管理員會使用政策，透過定義哪些主體可在哪些條件下對哪些資源執行動作，以指定可存取的範圍。

預設情況下，使用者和角色沒有許可。IAM 管理員會建立 IAM 政策並將其新增至角色，供使用者後續擔任。IAM 政策定義動作的許可，無論採用何種方式執行。

身分型政策

身分型政策是附加至身分 (使用者、使用者群組或角色) 的 JSON 許可政策文件。這類政策控制身分可對哪些資源執行哪些動作，以及適用的條件。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的[透過客戶管理政策定義自訂 IAM 許可](#)。

身分型政策可分為內嵌政策 (直接內嵌於單一身分) 與受管政策 (可附加至多個身分的獨立政策)。如需了解如何在受管政策及內嵌政策之間做選擇，請參閱《IAM 使用者指南》中的[在受管政策與內嵌政策之間選擇](#)。

資源型政策

資源型政策是附加到資源的 JSON 政策文件。範例包括 IAM 角色信任政策與 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。您必須在資源型政策中[指定主體](#)。

資源型政策是位於該服務中的內嵌政策。您無法在資源型政策中使用來自 IAM 的 AWS 受管政策。

其他政策類型

AWS 支援其他政策類型，可設定更多常見政策類型授予的最大許可：

- 許可界限 — 設定身分型政策可授與 IAM 實體的最大許可。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 實體許可界限](#)。
- 服務控制政策 (SCP) — 為 AWS Organizations 中的組織或組織單位指定最大許可。如需詳細資訊，請參閱《AWS Organizations 使用者指南》中的[服務控制政策](#)。
- 資源控制政策 (RCP) — 設定您帳戶中資源可用許可的上限。如需詳細資訊，請參閱《AWS Organizations 使用者指南》中的[資源控制政策 \(RCP\)](#)。
- 工作階段政策 — 在以程式設計方式為角色或聯合身分使用者建立臨時工作階段時，以參數形式傳遞的進階政策。如需詳細資訊，請參閱《IAM 使用者指南》中的[工作階段政策](#)。

多種政策類型

當多種類型的政策適用於請求時，產生的許可會更複雜而無法理解。若要了解如何 AWS 在涉及多個政策類型時決定是否允許請求，請參閱《IAM 使用者指南》中的[政策評估邏輯](#)。

AWS Ground Station 如何使用 IAM

在您使用 IAM 管理對的存取之前 AWS Ground Station，請先了解可使用哪些 IAM 功能 AWS Ground Station。

您可以搭配使用的 IAM 功能 AWS Ground Station

IAM 功能	AWS Ground Station 支援
身分型政策	是

IAM 功能	AWS Ground Station 支援
資源型政策	否
政策動作	是
政策資源	是
政策條件索引鍵 (服務特定)	是
ACL	否
ABAC (政策中的標籤)	是
臨時憑證	是
主體許可	是
服務角色	否
服務連結角色	是

若要全面了解 AWS Ground Station 和其他 AWS 服務如何與大多數 IAM 功能搭配使用，請參閱 [《AWS IAM 使用者指南》中的與 IAM 搭配使用的服務](#)。

的身分型政策 AWS Ground Station

支援身分型政策：是

身分型政策是可以附加到身分 (例如 IAM 使用者、使用者群組或角色) 的 JSON 許可政策文件。這些政策可控制身分在何種條件下能對哪些資源執行哪些動作。如需了解如何建立身分型政策，請參閱《IAM 使用者指南》中的 [透過客戶管理政策定義自訂 IAM 許可](#)。

使用 IAM 身分型政策，您可以指定允許或拒絕的動作和資源，以及在何種條件下允許或拒絕動作。如要了解您在 JSON 政策中使用的所有元素，請參閱《IAM 使用者指南》中的 [IAM JSON 政策元素參考](#)。

的身分型政策範例 AWS Ground Station

若要檢視 AWS Ground Station 身分型政策的範例，請參閱 [的身分型政策範例 AWS Ground Station](#)。

內的資源型政策 AWS Ground Station

支援資源型政策：否

資源型政策是附加到資源的 JSON 政策文件。資源型政策的最常見範例是 IAM 角色信任政策和 Amazon S3 儲存貯體政策。在支援資源型政策的服務中，服務管理員可以使用它們來控制對特定資源的存取權限。對於附加政策的資源，政策會定義指定的主體可以對該資源執行的動作以及在何種條件下執行的動作。您必須在資源型政策中[指定主體](#)。委託人可以包括帳戶、使用者、角色、聯合身分使用者或 AWS 服務。

如需啟用跨帳戶存取權，您可以在其他帳戶內指定所有帳戶或 IAM 實體作為資源型政策的主體。如需詳細資訊，請參閱《IAM 使用者指南》中的[IAM 中的快帳戶資源存取](#)。

的政策動作 AWS Ground Station

支援政策動作：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

JSON 政策的 Action 元素描述您可以用來允許或拒絕政策中存取的動作。政策會使用動作來授予執行相關聯動作的許可。

若要查看 AWS Ground Station 動作清單，請參閱《服務授權參考》中的[定義的動作 AWS Ground Station](#)。

中的政策動作在動作之前 AWS Ground Station 使用以下字首：

```
groundstation
```

若要在單一陳述式中指定多個動作，請用逗號分隔。

```
"Action": [  
    "groundstation:action1",  
    "groundstation:action2"  
]
```

若要檢視 AWS Ground Station 身分型政策的範例，請參閱[的身分型政策範例 AWS Ground Station](#)。

的政策資源 AWS Ground Station

支援政策資源：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Resource JSON 政策元素可指定要套用動作的物件。最佳實務是使用其 [Amazon Resource Name \(ARN\)](#) 來指定資源。若動作不支援資源層級許可，使用萬用字元 (*) 表示該陳述式適用於所有資源。

```
"Resource": "*"
```

若要查看 AWS Ground Station 資源類型及其 ARNs，請參閱《服務授權參考》中的 [定義的資源 AWS Ground Station](#)。若要了解您可以使用哪些動作指定每個資源的 ARN，請參閱 [AWS Ground Station 定義的動作](#)。

若要檢視 AWS Ground Station 身分型政策的範例，請參閱 [的身分型政策範例 AWS Ground Station](#)。

的政策條件索引鍵 AWS Ground Station

支援服務特定政策條件金鑰：是

管理員可以使用 AWS JSON 政策來指定誰可以存取內容。也就是說，哪個主體在什麼條件下可以對什麼資源執行哪些動作。

Condition 元素會根據定義的條件，指定陳述式的執行時機。您可以建立使用 [條件運算子](#) 的條件運算式 (例如等於或小於)，來比對政策中的條件和請求中的值。若要查看所有 AWS 全域條件索引鍵，請參閱《IAM 使用者指南》中的 [AWS 全域條件內容索引鍵](#)。

若要查看 AWS Ground Station 條件索引鍵的清單，請參閱《服務授權參考》中的 [的條件索引鍵 AWS Ground Station](#)。若要了解您可以使用條件索引鍵的動作和資源，請參閱 [定義的動作 AWS Ground Station](#)。

若要檢視 AWS Ground Station 身分型政策的範例，請參閱 [的身分型政策範例 AWS Ground Station](#)。

中的 ACLs AWS Ground Station

支援 ACL：否

存取控制清單 (ACL) 可控制哪些主體 (帳戶成員、使用者或角色) 擁有存取某資源的許可。ACL 類似於資源型政策，但它們不使用 JSON 政策文件格式。

ABAC 搭配 AWS Ground Station

支援 ABAC (政策中的標籤)：是

屬性型存取控制 (ABAC) 是一種授權策略，依據稱為標籤的屬性來定義許可。您可以將標籤連接至 IAM 實體 AWS 和資源，然後設計 ABAC 政策，以便在委託人的標籤符合資源上的標籤時允許操作。

如需根據標籤控制存取，請使用 `aws:ResourceTag/key-name`、`aws:RequestTag/key-name` 或 `aws:TagKeys` 條件索引鍵，在政策的 [條件元素](#) 中，提供標籤資訊。

如果服務支援每個資源類型的全部三個條件金鑰，則對該服務而言，值為 Yes。如果服務僅支援某些資源類型的全部三個條件金鑰，則值為 Partial。

如需 ABAC 的詳細資訊，請參閱《IAM 使用者指南》中的 [使用 ABAC 授權定義許可](#)。如要查看含有設定 ABAC 步驟的教學課程，請參閱《IAM 使用者指南》中的 [使用屬性型存取控制 \(ABAC\)](#)。

搭配 使用臨時登入資料 AWS Ground Station

支援臨時憑證：是

臨時登入資料提供 AWS 資源的短期存取權，當您使用聯合或切換角色時，會自動建立。AWS 建議您動態產生臨時登入資料，而不是使用長期存取金鑰。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM 中的臨時安全憑證與可與 IAM 搭配運作的 AWS 服務](#)。

的跨服務主體許可 AWS Ground Station

支援轉寄存取工作階段 (FAS)：是

轉送存取工作階段 (FAS) 使用呼叫的委託人許可 AWS 服務，結合 AWS 服務請求向下游服務提出請求。如需提出 FAS 請求時的政策詳細資訊，請參閱 [轉發存取工作階段](#)。

的服務角色 AWS Ground Station

支援服務角色：否

服務角色是服務擔任的 [IAM 角色](#)，可代您執行動作。IAM 管理員可以從 IAM 內建立、修改和刪除服務角色。如需詳細資訊，請參閱《IAM 使用者指南》中的 [建立角色以委派許可給 AWS 服務](#)。

Warning

變更服務角色的許可可能會中斷 AWS Ground Station 功能。只有在 AWS Ground Station 提供指引時，才能編輯服務角色。

的服務連結角色 AWS Ground Station

支援服務連結角色：是

服務連結角色是連結至的一種服務角色 AWS 服務。服務可以擔任代表您執行動作的角色。服務連結角色會出現在您的 AWS 帳戶，並由服務擁有。IAM 管理員可以檢視，但不能編輯服務連結角色的許可。

如需建立或管理服务連結角色的詳細資訊，請參閱[可搭配 IAM 運作的AWS 服務](#)。在資料表中尋找服務，其中包含服務連結角色欄中的 Yes。選擇是連結，以檢視該服務的服務連結角色文件。

的身分型政策範例 AWS Ground Station

根據預設，使用者和角色不具備建立或修改 AWS Ground Station 資源的權限。若要授予使用者對其所需資源執行動作的許可，IAM 管理員可以建立 IAM 政策。

如需了解如何使用這些範例 JSON 政策文件建立 IAM 身分型政策，請參閱《IAM 使用者指南》中的[建立 IAM 政策 \(主控台\)](#)。

如需定義的動作和資源類型的詳細資訊 AWS Ground Station，包括每種資源類型的 ARNs 格式，請參閱《服務授權參考》中的[的動作、資源和條件索引鍵 AWS Ground Station](#)。

主題

- [政策最佳實務](#)
- [使用 AWS Ground Station 主控台](#)
- [允許使用者檢視他們自己的許可](#)

政策最佳實務

身分型政策會判斷您帳戶中的某個人員是否可以建立、存取或刪除 AWS Ground Station 資源。這些動作可能會讓您的 AWS 帳戶產生費用。當您建立或編輯身分型政策時，請遵循下列準則及建議事項：

- 開始使用 AWS 受管政策並邁向最低權限許可 – 若要開始將許可授予您的使用者和工作負載，請使用將許可授予許多常見使用案例的 AWS 受管政策。它們可在您的 AWS 帳戶中使用。我們建議您定義特定於使用案例 AWS 的客戶受管政策，以進一步減少許可。如需更多資訊，請參閱《IAM 使用者指南》中的[AWS 受管政策](#)或[任務職能的AWS 受管政策](#)。
- 套用最低權限許可 – 設定 IAM 政策的許可時，請僅授予執行任務所需的許可。為實現此目的，您可以定義在特定條件下可以對特定資源採取的動作，這也稱為最低權限許可。如需使用 IAM 套用許可的更多相關資訊，請參閱《IAM 使用者指南》中的[IAM 中的政策和許可](#)。

- 使用 IAM 政策中的條件進一步限制存取權 – 您可以將條件新增至政策，以限制動作和資源的存取。例如，您可以撰寫政策條件，指定必須使用 SSL 傳送所有請求。如果透過特定等使用服務動作 AWS 服務，您也可以使用條件來授予其存取權 CloudFormation。如需詳細資訊，請參閱《IAM 使用者指南》中的 [IAM JSON 政策元素：條件](#)。
- 使用 IAM Access Analyzer 驗證 IAM 政策，確保許可安全且可正常運作 – IAM Access Analyzer 驗證新政策和現有政策，確保這些政策遵從 IAM 政策語言 (JSON) 和 IAM 最佳實務。IAM Access Analyzer 提供 100 多項政策檢查及切實可行的建議，可協助您撰寫安全且實用的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 [使用 IAM Access Analyzer 驗證政策](#)。
- 需要多重要素驗證 (MFA) – 如果您的案例需要 IAM 使用者或中的根使用者 AWS 帳戶，請開啟 MFA 以提高安全性。如需在呼叫 API 操作時請求 MFA，請將 MFA 條件新增至您的政策。如需詳細資訊，請參閱《IAM 使用者指南》中的 [透過 MFA 的安全 API 存取](#)。

如需 IAM 中最佳實務的相關資訊，請參閱《IAM 使用者指南》中的 [IAM 安全最佳實務](#)。

使用 AWS Ground Station 主控台

若要存取 AWS Ground Station 主控台，您必須擁有一組最低許可。這些許可必須允許您列出和檢視中 AWS Ground Station 資源的詳細資訊 AWS 帳戶。如果您建立比最基本必要許可更嚴格的身分型政策，則對於具有該政策的實體 (使用者或角色) 而言，主控台就無法如預期運作。

對於僅呼叫 AWS CLI 或 AWS API 的使用者，您不需要允許最低主控台許可。反之，只需允許存取符合他們嘗試執行之 API 操作的動作就可以了。

為了確保使用者和角色仍然可以使用 AWS Ground Station 主控台，請將 AWS Ground Station *ConsoleAccess* 或 *ReadOnly* AWS 受管政策連接到實體。如需詳細資訊，請參閱《IAM 使用者指南》中的 [新增許可到使用者](#)。

允許使用者檢視他們自己的許可

此範例會示範如何建立政策，允許 IAM 使用者檢視附加到他們使用者身分的內嵌及受管政策。此政策包含在主控台或使用或 AWS CLI AWS API 以程式設計方式完成此動作的許可。

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "ViewOwnUserInfo",
      "Effect": "Allow",
      "Action": [
```

```
        "iam:GetUserPolicy",
        "iam:ListGroupsForUser",
        "iam:ListAttachedUserPolicies",
        "iam:ListUserPolicies",
        "iam:GetUser"
    ],
    "Resource": ["arn:aws:iam::*:user/${aws:username}"]
},
{
    "Sid": "NavigateInConsole",
    "Effect": "Allow",
    "Action": [
        "iam:GetGroupPolicy",
        "iam:GetPolicyVersion",
        "iam:GetPolicy",
        "iam:ListAttachedGroupPolicies",
        "iam:ListGroupPolicies",
        "iam:ListPolicyVersions",
        "iam:ListPolicies",
        "iam:ListUsers"
    ],
    "Resource": "*"
}
]
```

對 AWS Ground Station 身分和存取進行故障診斷

使用以下資訊來協助您診斷和修正使用 AWS Ground Station 和 IAM 時可能遇到的常見問題。

主題

- [我無權在 中執行動作 AWS Ground Station](#)
- [我未獲得執行 iam:PassRole 的授權](#)
- [我想要允許 以外的人員 AWS 帳戶 存取我的 AWS Ground Station 資源](#)

我無權在 中執行動作 AWS Ground Station

如果您收到錯誤，告知您未獲授權執行動作，您的政策必須更新，允許您執行動作。

下列範例錯誤會在mateojackson IAM 使用者嘗試使用主控台檢視一個虛構 *my-example-widget* 資源的詳細資訊，但卻無虛構 `groundstation:GetWidget` 許可時發生。

```
User: arn:aws:iam::123456789012:user/mateojackson is not authorized to perform:
groundstation:GetWidget on resource: my-example-widget
```

在此情況下，必須更新 mateojackson 使用者的政策，允許使用 `groundstation:GetWidget` 動作存取 *my-example-widget* 資源。

如果您需要協助，請聯絡您的 AWS 管理員。您的管理員提供您的簽署憑證。

我未獲得執行 iam:PassRole 的授權

如果您收到錯誤，告知您未獲授權執行 `iam:PassRole` 動作，您的政策必須更新，允許您將角色傳遞給 AWS Ground Station。

有些 AWS 服務可讓您將現有角色傳遞給該服務，而不是建立新的服務角色或服務連結角色。如需執行此作業，您必須擁有將角色傳遞至該服務的許可。

名為 marymajor 的 IAM 使用者嘗試使用主控台在 AWS Ground Station 中執行動作時，發生下列範例錯誤。但是，動作要求服務具備服務角色授予的許可。Mary 沒有將角色傳遞給服務的許可。

```
User: arn:aws:iam::123456789012:user/marymajor is not authorized to perform:
iam:PassRole
```

在這種情況下，Mary 的政策必須更新，允許她執行 `iam:PassRole` 動作。

如果您需要協助，請聯絡您的 AWS 管理員。您的管理員提供您的簽署憑證。

我想要允許以外的人員 AWS 帳戶存取我的 AWS Ground Station 資源

您可以建立一個角色，讓其他帳戶中的使用者或您組織外部的人員存取您的資源。您可以指定要允許哪些信任物件取得該角色。針對支援基於資源的政策或存取控制清單 (ACL) 的服務，您可以使用那些政策來授予人員存取您的資源的許可。

如需進一步了解，請參閱以下內容：

- 若要了解是否 AWS Ground Station 支援這些功能，請參閱 [AWS Ground Station 如何使用 IAM](#)。
- 若要了解如何 AWS 帳戶 在您擁有的 資源之間提供存取權，請參閱 [《IAM 使用者指南》中的在您擁有 AWS 帳戶 的另一個 IAM 使用者中提供存取權](#)。

- 若要了解如何將資源的存取權提供給第三方 AWS 帳戶，請參閱《IAM 使用者指南》中的[將存取權提供給第三方 AWS 帳戶 擁有](#)。
- 如需了解如何透過聯合身分提供存取權，請參閱《IAM 使用者指南》中的[將存取權提供給在外部進行身分驗證的使用者 \(聯合身分\)](#)。
- 如需了解使用角色和資源型政策進行跨帳戶存取之間的差異，請參閱《IAM 使用者指南》中的[IAM 中的跨帳戶資源存取](#)。

AWS 的 受管政策 AWS Ground Station

AWS 受管政策是由 AWS AWS 受管政策建立和管理的獨立政策旨在為許多常用案例提供許可，以便您可以開始將許可指派給使用者、群組和角色。

請記住，AWS 受管政策可能不會授予特定使用案例的最低權限許可，因為這些許可可供所有 AWS 客戶使用。我們建議您定義特定於使用案例的[客戶管理政策](#)，以便進一步減少許可。

您無法變更 AWS 受管政策中定義的許可。如果 AWS 更新受 AWS 管政策中定義的許可，則更新會影響政策連接的所有委託人身分（使用者、群組和角色）。當新的 AWS 服務 啟動或新的 API 操作可供現有服務使用時，AWS 最有可能更新 AWS 受管政策。

如需詳細資訊，請參閱 IAM 使用者指南中的[AWS 受管政策](#)。

AWS 受管政策：AWSGroundStationAgentInstancePolicy

您可將 AWSGroundStationAgentInstancePolicy 政策連接到 IAM 身分。

此政策會將 AWS Ground Station 代理程式許可授予 Amazon EC2 執行個體，允許執行個體在 Ground Station 聯絡期間傳送和接收資料。此政策中的所有許可都來自 Ground Station 服務。

許可詳細資訊

此政策包含以下許可。

- `groundstation` – 允許資料流程端點執行個體呼叫 Ground Station Agent APIs。

若要檢視最新版本的 JSON 政策文件，請參閱《AWS 受管政策參考指南》中的 [AWSGroundStationAgentInstancePolicy](#)。

AWS 受管政策：

AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy

您無法將 AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy 連接至 IAM 實體。此政策會連接到服務連結角色，AWS Ground Station 允許代表您執行動作。如需詳細資訊，請參閱 [使用服務連結角色](#)。

此政策會授予允許尋找公 AWS Ground Station 有 IPv4 地址的 EC2 許可。

許可詳細資訊

此政策包含以下許可。

- `ec2:DescribeAddresses` – 允許代表您 AWS Ground Station 列出與 EIPs 相關聯的所有 IPs。
- `ec2:DescribeNetworkInterfaces` – 允許代表您 AWS Ground Station 取得與 EC2 執行個體相關聯的網路介面資訊。

若要檢視最新版本的 JSON 政策文件，請參閱《AWS 受管政策參考指南》中的 [AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy](#)。

AWS Ground Station AWS 受管政策的更新

檢視自此服務開始追蹤這些變更 AWS Ground Station 以來，AWS 受管政策更新的詳細資訊。如需此頁面變更的自動提醒，請訂閱 AWS Ground Station 文件歷史記錄頁面上的 RSS 摘要。

變更	描述	Date
AWSGroundStationAgentInstancePolicy – 更新至現有政策	AWS Ground Station 新增了允許代理程式擷取任務回應	2025 年 11 月 13 日

變更	描述	Date
	URLs以進行增強型 Ground Station 操作的許可。	
AWSGroundStationAgentInstancePolicy – 新政策	AWS Ground Station 新增了新政策，以提供資料流程端點執行個體使用 AWS Ground Station Agent 的許可。	2023 年 4 月 12 日
AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy – 新政策	AWS Ground Station 新增了新的政策，授予 EC2 許可，以允許尋找與 EIPs 相關聯的公有 AWS Ground Station 有 IPv4 地址，以及與 EC2 執行個體相關聯的網路介面。	2022 年 11 月 2 日
AWS Ground Station 開始追蹤變更	AWS Ground Station 已開始追蹤 AWS 受管政策的變更。	2021 年 3 月 1 日

使用 Ground Station 的服務連結角色

AWS Ground Station use AWS Identity and Access Management (IAM) [服務連結角色](#)。服務連結角色是直接連結至 Ground Station 的唯一 IAM 角色類型。Ground Station 會預先定義服務連結角色，並包含該服務代表您呼叫其他 AWS 服務所需的所有許可。

服務連結角色可讓您更輕鬆地設定 Ground Station，因為您不必手動新增必要的許可。Ground Station 定義其服務連結角色的許可，除非另有定義，否則只有 Ground Station 可以擔任其角色。定義的許可包括信任政策和許可政策，且該許可政策無法附加至其他 IAM 實體。

如需有關支援服務連結角色的其他服務的資訊，請參閱[AWS 使用 IAM 的服務](#)，並在服務連結角色欄中尋找具有是的服務。選擇具有連結的是，以檢視該服務的服務連結角色文件。

Ground Station 的服務連結角色許可

Ground Station 使用名為 `AWSServiceRoleForGroundStationDataflowEndpointGroup` 的服務連結角色 – AWS GroundStation 使用此服務連結角色來叫用 EC2 來尋找公有 IPv4 地址。

`AWSServiceRoleForGroundStationDataflowEndpointGroup` 服務連結角色信任下列服務擔任該角色：

- `groundstation.amazonaws.com`

名為 `AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy` 的角色許可政策可讓 Ground Station 對指定的資源完成下列動作：

- 動作：`all AWS resources (*)` 上的 `ec2:DescribeAddresses`
- 動作：`all AWS resources (*)` 上的 `ec2:DescribeNetworkInterfaces`

動作可讓 Ground Station 列出與 EIPs 相關聯的所有 IPs。

動作可讓 Ground Station 取得與 EC2 執行個體相關聯之網路介面的相關資訊

您必須設定許可，IAM 實體 (如使用者、群組或角色) 才可建立、編輯或刪除服務連結角色。如需詳細資訊，請參閱 IAM 使用者指南中的 [服務連結角色許可](#)。

為 Ground Station 建立服務連結角色

您不需要手動建立服務連結角色，當您在 AWS CLI 或 AWS API 中建立 `DataflowEndpointGroup` 時，Ground Station 會為您建立服務連結角色。

若您刪除此服務連結角色，之後需要再次建立，您可以在帳戶中使用相同程序重新建立角色。當您建立 `DataflowEndpointGroup` 時，Ground Station 會再次為您建立服務連結角色。

您也可以使用 IAM 主控台，透過資料交付至 Amazon EC2 使用案例來建立服務連結角色。在 AWS CLI 或 AWS API 中，使用服務名稱建立 `groundstation.amazonaws.com` 服務連結角色。如需詳細資訊，請參閱《IAM 使用者指南》中的「[建立服務連結角色](#)」。如果您刪除此服務連結角色，您可以使用此相同的程序以再次建立該角色。

編輯 Ground Station 的服務連結角色

Ground Station 不允許您編輯 `AWSServiceRoleForGroundStationDataflowEndpointGroup` 服務連結角色。因為有各種實體可能會參考服務連結角色，所以您無法在建立角色之後變更角色名稱。然而，您可以使用 IAM 來編輯角色描述。如需詳細資訊，請參閱《IAM 使用者指南》中的 [編輯服務連結角色](#)。

刪除 Ground Station 的服務連結角色

若您不再使用需要服務連結角色的功能或服務，我們建議您刪除該角色。如此一來，您就沒有未主動監控或維護的未使用實體。

您必須先使用服務連結角色刪除 DataflowEndpointGroups，才能刪除服務連結角色。這可避免您不小心撤銷 DataflowEndpointGroups 的許可。如果服務連結角色與多個 DataflowEndpointGroups 搭配使用，您必須先刪除使用服務連結角色的所有 DataflowEndpointGroups，才能將其刪除。

Note

如果 Ground Station 服務在您嘗試刪除資源時使用角色，則刪除可能會失敗。若此情況發生，請等待數分鐘後並再次嘗試操作。

刪除 AWSServiceRoleForGroundStationDataflowEndpointGroup 所使用的 Ground Station 資源

- 透過 AWS CLI 或 AWS API 刪除 DataflowEndpointGroups。

使用 IAM 手動刪除服務連結角色

使用 IAM 主控台 AWS CLI、或 AWS API 來刪除

AWSServiceRoleForGroundStationDataflowEndpointGroup 服務連結角色。如需詳細資訊，請參閱《IAM 使用者指南》中的[刪除服務連結角色](#)。

Ground Station 服務連結角色支援的區域

Ground Station 支援在所有提供服務的區域中使用服務連結角色。如需詳細資訊，請參閱[區域表](#)。

疑難排解

NOT_AUTHORIZED_TO_CREATE_SLR - 這表示您帳戶中用來呼叫 CreateDataflowEndpointGroup API 的角色沒有 iam:CreateServiceLinkedRole 許可。具有 iam:CreateServiceLinkedRole 許可的管理員必須手動為您的帳戶建立服務連結角色。

的靜態資料加密 AWS Ground Station

AWS Ground Station 預設提供加密，以使用 AWS 擁有的加密金鑰保護您的靜態敏感資料。

- AWS 擁有的金鑰 - 預設 AWS Ground Station 使用這些金鑰自動加密可直接識別個人身分的資料和暫時性資料。您無法檢視、管理或使用擁有 AWS 的金鑰，或稽核其使用方式；不過，您不需要採取任何動作或變更程式來保護加密資料的金鑰。如需詳細資訊，請參閱《[AWS Key Management Service 開發人員指南](#)》中的[AWS擁有的金鑰](#)。

根據預設，加密靜態資料有助於降低保護敏感資料所涉及的操作開銷和複雜性。同時，它可以建立符合嚴格加密合規以及法規要求的安全應用程式。

AWS Ground Station 對所有敏感的靜態資料強制執行加密，不過，對於某些 AWS Ground Station 資源，例如暫時性資料，您可以選擇使用客戶受管金鑰來取代預設的 AWS 受管金鑰。

- 客戶受管金鑰 - AWS Ground Station 支援使用您建立、擁有和管理的對稱客戶受管金鑰來取代現有的 AWS 擁有加密。您可以完全控制此層加密，因此能執行以下任務：
 - 建立和維護金鑰政策
 - 建立和維護 IAM 政策和授予操作
 - 啟用和停用金鑰政策
 - 輪換金鑰密碼編譯資料
 - 新增 標籤
 - 建立金鑰別名
 - 安排金鑰供刪除

如需詳細資訊，請參閱《[AWS Key Management Service 開發人員指南](#)》中的[客戶受管金鑰](#)。

下表摘要說明 AWS Ground Station 支援使用客戶受管金鑰的資源

資料類型	AWS 擁有的金鑰加密	客戶自管金鑰加密 (選用)
Ephemeris 資料用於計算衛星的軌跡	已啟用	已啟用
用來命令天線的方位海拔暫時性曲線	已啟用	已啟用

Note

AWS Ground Station 會使用 自動啟用靜態加密 AWS 擁有的金鑰，以免費保護個人身分識別資料。不過，使用客戶受管金鑰會產生 AWS KMS 費用。如需定價的詳細資訊，請參閱 [AWS Key Management Service 定價](#)。

如需詳細資訊 AWS KMS，請參閱 [AWS Key Management Service 開發人員指南](#)。

如需每個資源類型的特定資訊，請參閱：

- [TLE 和 OEM ephemeris 資料的靜態加密](#)
- [疊流高度暫時性的靜態加密](#)

建立客戶自管金鑰

您可以使用 AWS 管理主控台或 AWS KMS APIs 來建立對稱客戶受管金鑰。

建立對稱客戶受管金鑰

遵循《[AWS Key Management Service 開發人員指南](#)》中建立對稱客戶受管金鑰的步驟。

金鑰政策概觀

金鑰政策會控制客戶受管金鑰的存取權限。每個客戶受管金鑰都必須只有一個金鑰政策，其中包含決定誰可以使用金鑰及其使用方式的陳述式。在建立客戶受管金鑰時，可以指定金鑰政策。如需詳細資訊，請參閱《[AWS Key Management Service 開發人員指南](#)》中的[管理對客戶受管金鑰的存取](#)。

若要將客戶受管金鑰與 AWS Ground Station 資源搭配使用，您必須設定金鑰政策以授予 AWS Ground Station 服務適當的許可。特定許可和政策組態取決於您正在加密的資源類型：

- 如需 TLE 和 OEM ephemeris 資料 - 如需特定金鑰政策需求和範例[TLE 和 OEM ephemeris 資料的靜態加密](#)，請參閱。
- 如需方位提升暫時性資料 - 如需特定金鑰政策需求和範例[疊流高度暫時性的靜態加密](#)，請參閱。

Note

金鑰政策組態在 ephemeris 類型之間有所不同。TLE 和 OEM ephemeris 資料使用授予金鑰存取，而方位提升 ephemeris 使用直接金鑰政策許可。請確定您根據要加密的特定資源類型來設定金鑰政策。

如需在[政策中指定許可](#)和[疑難排解金鑰存取](#)的詳細資訊，請參閱 AWS Key Management Service 開發人員指南。

指定的客戶受管金鑰 AWS Ground Station

您可以指定客戶受管金鑰來加密下列資源：

- Ephemeris (TLE、OEM 和方位提升)

建立資源時，您可以提供 kmsKeyArn 來指定資料金鑰

- kmsKeyArn - AWS KMS 客戶受管[金鑰的金鑰識別符](#)

AWS Ground Station 加密內容

[加密內容](#)是一組選用的金鑰/值對，其中包含有關資料的其他內容資訊。AWS KMS 會使用加密內容做為額外的已驗證資料，以支援已驗證的加密。當您在加密資料的請求中包含加密內容時，會將加密內容 AWS KMS 繫結至加密的資料。若要解密資料，您必須在請求中包含相同的加密內容。

AWS Ground Station 根據加密的資源使用不同的加密內容，並為每個建立的金鑰授權指定特定的加密內容。

如需資源特定的加密內容詳細資訊，請參閱：

- [TLE 和 OEM ephemeris 資料的靜態加密](#)
- [疊流高度暫時性的靜態加密](#)

TLE 和 OEM ephemeris 資料的靜態加密

TLE 和 OEM ephemeris 的關鍵政策需求

若要搭配 ephemeris 資料使用客戶受管金鑰，您的金鑰政策必須將下列許可授予 AWS Ground Station 服務：

- [kms:CreateGrant](#) - 在客戶受管金鑰上建立存取授權。授予在客戶受管金鑰上執行[授予操作](#)的 AWS Ground Station 存取權，以讀取和儲存加密的資料。
- [kms:DescribeKey](#) - 提供客戶受管金鑰詳細資訊，AWS Ground Station 允許在嘗試使用提供的金鑰之前驗證金鑰。

如需[使用授權](#)的詳細資訊，請參閱 AWS Key Management Service 開發人員指南。

使用客戶受管金鑰建立 ephemeris 的 IAM 使用者許可

當 AWS Ground Station 在密碼編譯操作中使用客戶受管金鑰時，它會代表建立 ephemeris 資源的使用者。

若要使用客戶受管金鑰建立 ephemeris 資源，使用者必須具有在客戶受管金鑰上呼叫下列操作的許可：

- [kms:CreateGrant](#) - 允許使用者代表客戶受管金鑰建立授予 AWS Ground Station。
- [kms:DescribeKey](#) - 允許使用者檢視客戶受管金鑰詳細資訊，以驗證金鑰。

您可以在金鑰政策或在 IAM 政策中指定這些必要的許可 (如果金鑰政策允許)。這些許可可確保使用者可以授權 代表他們 AWS Ground Station 使用客戶受管金鑰進行加密操作。

如何在 中 使用 AWS Ground Station AWS KMS 授予進行暫時性差異

AWS Ground Station 需要[金鑰授權](#)才能使用您的客戶受管金鑰。

當您上傳使用客戶受管金鑰加密的 ephemeris 時，會透過傳送 [CreateGrant](#) 請求至 來代表您 AWS Ground Station 建立金鑰授權 AWS KMS。中的授予 AWS KMS 用於授予 AWS Ground Station 您帳戶中 AWS KMS 金鑰的存取權。

這可讓 AWS Ground Station 執行下列動作：

- 呼叫 [GenerateDataKey](#) 以產生加密的資料金鑰並將其存放，因為不會立即使用資料金鑰進行加密。
- 呼叫 [Decrypt](#) 以使用儲存的加密資料金鑰來存取加密的資料。
- 呼叫 [Encrypt](#) 以使用資料金鑰來加密資料。
- 設定淘汰委託人，以允許服務轉換為 [RetireGrant](#)。

您可以隨時撤銷授予的存取權。如果您這麼做，AWS Ground Station 則無法存取客戶受管金鑰加密的任何資料，這會影響依賴該資料的操作。例如，如果您從目前用於聯絡的 ephemeris 移除金鑰授予，AWS Ground Station 則將無法在聯絡期間使用提供的 ephemeris 資料來指向天線。這會導致聯絡人以 FAILED 狀態結束。

Ephemeris 加密內容

加密 ephemeris 資源的金鑰授予會繫結至特定的衛星 ARN。

```
"encryptionContext": {
  "aws:groundstation:arn":
  "arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE",
  "aws:s3:arn":
  "arn:aws:s3:::customerephemerisbucket/0034abcd-12ab-34cd-56ef-123456SAMPLE"
}
```

Note

金鑰授權會針對相同的金鑰衛星對重複使用。

使用加密內容進行監控

當您使用對稱客戶受管金鑰來加密暫時性資料時，您也可以使用稽核記錄和日誌中的加密內容來識別客戶受管金鑰的使用方式。加密內容也會出現在 [AWS CloudTrail](#) 或 [Amazon CloudWatch Logs](#) 產生的日誌中。

使用加密內容控制對客戶受管金鑰的存取

您也可以在金鑰政策和 IAM 政策中，使用加密內容作為 conditions 來控制對於對稱客戶受管金鑰的存取。您也可以在授予中使用加密內容條件。

AWS Ground Station 在授予中使用加密內容限制，以控制對您帳戶或區域中客戶受管金鑰的存取。授予條件會要求授予允許的操作使用指定的加密內容。

以下是授予特定加密內容之客戶受管金鑰存取權的金鑰政策陳述式範例。此政策陳述式中的條件會要求具有指定加密內容的加密內容條件。

下列範例顯示繫結至衛星之 ephemeris 資料的金鑰政策：

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Create Grant on key",
      "Effect": "Allow",
```

```

    "Principal": {
      "Service": "groundstation.us-east-1.amazonaws.com"
    },
    "Action": "kms:CreateGrant",
    "Resource": "*",
    "Condition": {
      "StringEquals": {
        "kms:EncryptionContext:aws:groundstation:arn":
"arn:aws:groundstation::123456789012:satellite/satellite-id"
      }
    }
  }
]
}

```

監控 ephemeris 的加密金鑰

當您搭配 ephemeris 資源使用 AWS Key Management Service 客戶受管金鑰時，您可以使用 [AWS CloudTrail](#) 或 [Amazon CloudWatch logs](#) 來追蹤 AWS Ground Station 傳送的請求 AWS KMS。下列範例是 [CreateGrant](#)、[GenerateDataKey](#)、[Decrypt](#) 和 [DescribeKey](#) 的 CloudTrail 事件，用於監控 AWS KMS 呼叫的操作 AWS Ground Station，以存取客戶受管金鑰加密的資料。

CreateGrant

當您使用 AWS KMS 客戶受管金鑰加密 ephemeris 資源時，會代表您 AWS Ground Station 傳送 [CreateGrant](#) 請求，以存取您 AWS 帳戶中的 AWS KMS 金鑰。建立的 AWS Ground Station 授予專屬於與客戶 AWS KMS 受管金鑰相關聯的資源。此外，AWS Ground Station 當您刪除資源時，會使用 [RetireGrant](#) 操作來移除授予。

下列範例事件會記錄 ephemeris 的 [CreateGrant](#) 操作：

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ASIAIOSFODNN7EXAMPLE",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",

```

```

        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
    },
    "webIdFederationData": {},
    "attributes": {
        "creationDate": "2022-02-22T22:22:22Z",
        "mfaAuthenticated": "false"
    }
},
"invokedBy": "AWS Internal"
},
"eventTime": "2022-02-22T22:22:22Z",
"eventSource": "kms.amazonaws.com",
"eventName": "CreateGrant",
"awsRegion": "us-west-2",
"sourceIPAddress": "AWS Internal",
"userAgent": "ExampleDesktop/1.0 (V1; OS)",
"requestParameters": {
    "operations": [
        "GenerateDataKeyWithoutPlaintext",
        "Decrypt",
        "Encrypt"
    ],
    "constraints": {
        "encryptionContextSubset": {
            "aws:groundstation:arn":
"arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE"
        }
    },
    "granteePrincipal": "groundstation.us-west-2.amazonaws.com",
    "retiringPrincipal": "groundstation.us-west-2.amazonaws.com",
    "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
},
"responseElements": {
    "grantId":
"0ab0ac0d0b000f00ea00cc0a0e00fc00bce000c000f0000000c0bc0a0000aaafSAMPLE"
},
"requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"readOnly": false,
"resources": [

```

```

    {
      "accountId": "111122223333",
      "type": "AWS::KMS::Key",
      "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    }
  ],
  "eventType": "AwsApiCall",
  "managementEvent": true,
  "recipientAccountId": "111122223333",
  "eventCategory": "Management"
}

```

DescribeKey

當您使用 AWS KMS 客戶受管金鑰加密 ephemeral 資源時，會代表您 AWS Ground Station 傳送 [DescribeKey](#) 請求，以驗證請求的金鑰是否存在於您的帳戶中。

下列範例事件會記錄 [DescribeKey](#) 操作：

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ASIAIOSFODNN7EXAMPLE",
    "arn": "arn:aws:sts::111122223333:assumed-role/User/Role",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Role",
        "accountId": "111122223333",
        "userName": "User"
      },
      "webIdFederationData": {},
      "attributes": {
        "creationDate": "2022-02-22T22:22:22Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "AWS Internal"
  },
}

```

```

    "eventTime": "2022-02-22T22:22:22Z",
    "eventSource": "kms.amazonaws.com",
    "eventName": "DescribeKey",
    "awsRegion": "us-west-2",
    "sourceIPAddress": "AWS Internal",
    "userAgent": "AWS Internal",
    "requestParameters": {
      "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
    },
    "responseElements": null,
    "requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
    "readOnly": true,
    "resources": [
      {
        "accountId": "111122223333",
        "type": "AWS::KMS::Key",
        "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
      }
    ],
    "eventType": "AwsApiCall",
    "managementEvent": true,
    "recipientAccountId": "111122223333",
    "eventCategory": "Management"
  }

```

GenerateDataKey

當您使用 AWS KMS 客戶受管金鑰來加密 ephemeral 資源時，會將 [GenerateDataKey](#) 請求 AWS Ground Station 傳送至 [AWS Ground Station](#)，以產生用於加密資料的資料金鑰。

下列範例事件會記錄 ephemeral 的 [GenerateDataKey](#) 操作：

```

{
  "eventVersion": "1.08",
  "userIdentity": {
    "type": "AWSService",
    "invokedBy": "AWS Internal"
  },
  "eventTime": "2022-02-22T22:22:22Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "GenerateDataKey",

```

```

"awsRegion": "us-west-2",
"sourceIPAddress": "AWS Internal",
"userAgent": "AWS Internal",
"requestParameters": {
  "keySpec": "AES_256",
  "encryptionContext": {
    "aws:groundstation:arn":
"arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE",
    "aws:s3:arn":
"arn:aws:s3:::customerephemerisbucket/0034abcd-12ab-34cd-56ef-123456SAMPLE"
  },
  "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
},
"responseElements": null,
"requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"sharedEventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"eventCategory": "Management"
}

```

Decrypt

當您使用 AWS KMS 客戶受管金鑰來加密 ephemeris 資源時，如果已使用相同的客戶受管金鑰加密，則 AWS Ground Station 會使用 [Decrypt](#) 操作來解密提供的 ephemeris。例如，如果 ephemeris 正在從 S3 儲存貯體上傳，並使用指定的金鑰在該儲存貯體中加密。

下列範例事件會記錄 ephemeris 的 [Decrypt](#) 操作：

```

{
  "eventVersion": "1.08",

```

```
"userIdentity": {
  "type": "AWSInternal",
  "invokedBy": "AWS Internal"
},
"eventTime": "2022-02-22T22:22:22Z",
"eventSource": "kms.amazonaws.com",
"eventName": "Decrypt",
"awsRegion": "us-west-2",
"sourceIPAddress": "AWS Internal",
"userAgent": "AWS Internal",
"requestParameters": {
  "encryptionContext": {
    "aws:groundstation:arn":
"arn:aws:groundstation::111122223333:satellite/00a770b0-082d-45a4-80ed-SAMPLE",
    "aws:s3:arn":
"arn:aws:s3:::customerephemerisbucket/0034abcd-12ab-34cd-56ef-123456SAMPLE"
  },
  "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
},
"responseElements": null,
"requestID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"eventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"sharedEventID": "ff000af-00eb-00ce-0e00-ea000fb0fba0SAMPLE",
"eventCategory": "Management"
}
```

疊流高度暫時性的靜態加密

方位提升暫時性的關鍵政策要求

若要將客戶受管金鑰與方位提升暫時性資料搭配使用，您的金鑰政策必須將下列許可授予 AWS Ground Station 服務。與使用授予的 TLE 和 OEM ephemeris 資料不同，azimuth elevation ephemeris 會使用直接金鑰政策許可來進行加密操作。這是管理許可和使用您的金鑰的更簡單方法。

- [kms:GenerateDataKey](#) - 產生用於加密方位提升暫時性資料的資料金鑰。
- [kms:Decrypt](#) - 存取方位提升暫時性資料時解密加密的資料金鑰。

授予客戶受管金鑰 AWS Ground Station 存取權的金鑰政策範例

Note

使用方位提升暫時性時，您必須直接在金鑰政策中設定這些許可。必須在您的金鑰政策陳述式中授予區域 AWS Ground Station 服務主體（例如 `groundstation.region.amazonaws.com`）這些許可。如果沒有這些新增至金鑰政策的陳述式，AWS Ground Station 將無法存放或存取您的自訂方位提升暫時性。

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      }
    }
  ]
}
```

```
    },
    "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
    ],
    "Resource": "*"
}
]
```

使用客戶受管金鑰建立方位提升暫時性的 IAM 使用者許可

當 AWS Ground Station 在密碼編譯操作中使用客戶受管金鑰時，它會代表正在建立方位提升暫時性資源的使用者。

若要使用客戶受管金鑰建立方位提升暫時性資源，使用者必須具有在客戶受管金鑰上呼叫下列操作的許可：

- [kms:GenerateDataKey](#) - 允許使用者產生用於加密方位提升暫時性資料的資料金鑰。
- [kms:Decrypt](#) - 允許使用者在存取方位提升暫時性資料時解密資料金鑰。
- [kms:DescribeKey](#) - 允許使用者檢視客戶受管金鑰詳細資訊，以驗證金鑰。

您可以在金鑰政策或在 IAM 政策中指定這些必要的許可 (如果金鑰政策允許)。這些許可可確保使用者可以授權 代表他們 AWS Ground Station 使用客戶受管金鑰進行加密操作。

如何 AWS Ground Station 將關鍵政策用於方位提升暫時性

當您使用客戶受管金鑰提供方位提升暫時性資料時，AWS Ground Station 會使用金鑰政策來存取您的加密金鑰。許可是透過 AWS Ground Station 金鑰政策陳述式直接授予給，而不是像使用 TLE 或 OEM ephemeris 資料一樣透過授予授予授予。

如果您移除對客戶受管金鑰的 AWS Ground Station 存取權，AWS Ground Station 則無法存取該金鑰加密的任何資料，這會影響依賴該資料的操作。例如，如果您移除目前用於聯絡的方位角升高暫時性的金鑰政策許可，AWS Ground Station 將無法在聯絡期間使用提供的方位角升高資料來命令天線。這會導致聯絡人以 FAILED 狀態結束。

方位浮水印加密內容

當 AWS Ground Station 使用您的 AWS KMS 金鑰來加密方位提升暫時性資料時，服務會指定 [加密內容](#)。加密內容是額外的驗證資料 (AAD)，AWS KMS 用於確保資料完整性。為加密操作指定加密內容

時，此服務必須為解密操作指定相同的加密內容。否則，解密會失敗。加密內容也會寫入 CloudTrail 日誌，以協助您了解使用指定 AWS KMS 金鑰的原因。您的 CloudTrail 日誌可能包含許多描述 AWS KMS 使用金鑰的項目，但每個日誌項目中的加密內容可協助您判斷該特定使用的原因。

AWS Ground Station 指定以下加密內容，當其使用客戶受管金鑰在方位提升暫時性上執行密碼編譯操作時：

```
{
  "encryptionContext": {
    "aws:groundstation:ground-station-id": "Ohio 1",
    "aws:groundstation:arn": "arn:aws:groundstation:us-east-2:111122223333:ephemeris/00a770b0-082d-45a4-80ed-SAMPLE",
    "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/00a770b0-082d-45a4-80ed-SAMPLE/raw"
  }
}
```

加密內容包含：

`aws:groundstation:ground-station-id`

與方位角海拔面積相關聯的地面站名稱。

`aws : groundstation : arn`

ephemeris 資源的 ARN。

`aws : s3 : arn`

存放在 Amazon S3 中 ephemeris 的 ARN。

使用加密內容控制對客戶受管金鑰的存取

您可以使用 IAM 條件陳述式來控制對客戶受管金鑰的 AWS Ground Station 存取。在 `kms:GenerateDataKey` 和 `kms:Decrypt` 動作上新增條件陳述式會限制 AWS KMS 可以使用的地面站。

以下是金鑰政策陳述式範例，可將 AWS Ground Station 存取權授予特定地面站點特定區域中的客戶受管金鑰。此政策陳述式中的條件要求對指定符合金鑰政策中條件之加密內容的金鑰的所有加密和解密存取權。

授予存取特定地面站之 AWS Ground Station 客戶受管金鑰的金鑰政策範例

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringEquals": {
          "kms:EncryptionContext:aws:groundstation:ground-station-id":
            "specific-ground-station-name"
        }
      }
    }
  ]
}
```

授予 AWS Ground Station 存取多個地面站點之客戶受管金鑰的金鑰政策範例

JSON

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Allow AWS Ground Station to Describe key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": "kms:DescribeKey",
      "Resource": "*"
    },
    {
      "Sid": "Allow AWS Ground Station to Encrypt and Decrypt with key",
      "Effect": "Allow",
      "Principal": {
        "Service": "groundstation.us-east-1.amazonaws.com"
      },
      "Action": [
        "kms:GenerateDataKey",
        "kms:Decrypt"
      ],
      "Resource": "*",
      "Condition": {
        "StringLike": {
          "kms:EncryptionContext:aws:groundstation:ground-station-id":
            [
              "specific-ground-station-name-1",
              "specific-ground-station-name-2"
            ]
        }
      }
    }
  ]
}
```

監控您的加密金鑰是否有方位提升暫時性

當您將 AWS KMS 客戶受管金鑰與方位提升 ephemeral 資源搭配使用時，您可以使用 [CloudTrail](#) 或 [CloudWatch 日誌](#) 來追蹤 AWS Ground Station 傳送的請求 AWS KMS。下列範例是 [GenerateDataKey](#) 和 [Decrypt](#) 的 CloudTrail 事件，用於監控 AWS KMS 呼叫的操作 AWS Ground Station，以存取客戶受管金鑰加密的資料。

GenerateDataKey

當您使用 AWS KMS 客戶受管金鑰來加密方位提升暫時性資源時，會將 [GenerateDataKey](#) 請求 AWS Ground Station 傳送至，AWS KMS 以產生用於加密資料的資料金鑰。

下列範例事件會記錄方位角升高暫時性的 [GenerateDataKey](#) 操作：

```
{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "AssumedRole",
    "principalId": "ASIAIOSFODNN7EXAMPLE",
    "arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
    "accountId": "111122223333",
    "accessKeyId": "ASIAIOSFODNN7EXAMPLE",
    "sessionContext": {
      "sessionIssuer": {
        "type": "Role",
        "principalId": "ASIAIOSFODNN7EXAMPLE",
        "arn": "arn:aws:iam::111122223333:role/Admin",
        "accountId": "111122223333",
        "userName": "Admin"
      },
      "attributes": {
        "creationDate": "2025-08-25T14:45:48Z",
        "mfaAuthenticated": "false"
      }
    },
    "invokedBy": "AWS Internal"
  },
  "eventTime": "2025-08-25T14:52:02Z",
  "eventSource": "kms.amazonaws.com",
  "eventName": "GenerateDataKey",
  "awsRegion": "us-west-2",
  "sourceIPAddress": "AWS Internal",
  "userAgent": "AWS Internal",
```

```

"requestParameters": {
  "keySpec": "AES_256",
  "encryptionContext": {
    "aws:groundstation:arn": "arn:aws:groundstation:us-
west-2:111122223333:ephemeris/bb650670-7a4b-4152-bd60-SAMPLE",
    "aws:groundstation:ground-station-id": "Ohio 1",
    "aws:s3:arn": "arn:aws:s3:::customerephemerisbucket/bb650670-7a4b-4152-
bd60-SAMPLE/raw"
  },
  "keyId": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
},
"responseElements": null,
"requestID": "ef6f9a8f-8ef6-46a1-bdcb-123456SAMPLE",
"eventID": "952842d4-1389-3232-b885-123456SAMPLE",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
    "ARN": "arn:aws:kms:us-
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"
  }
],
"eventType": "AwsApiCall",
"managementEvent": true,
"recipientAccountId": "111122223333",
"sharedEventID": "8424f6b6-2280-4d1d-b9fd-0348b1546cba",
"eventCategory": "Management"
}

```

Decrypt

當您使用 AWS KMS 客戶受管金鑰來加密方位提升暫時性資源時，如果提供的方位提升暫時性資料已使用相同的客戶受管金鑰加密，則 AWS Ground Station 會使用[解密](#)操作來解密他們所提供的方位提升暫時性資料。

下列範例事件會記錄方位角升高暫時性的[解密](#)操作：

```

{
  "eventVersion": "1.11",
  "userIdentity": {
    "type": "AssumedRole",

```

```
"principalId": "ASIAIOSFODNN7EXAMPLE",
"arn": "arn:aws:sts::111122223333:assumed-role/Admin/SampleUser01",
"accountId": "111122223333",
"accessKeyId": "ASIAIOSFODNN7EXAMPLE",
"sessionContext": {
  "sessionIssuer": {
    "type": "Role",
    "principalId": "ASIAIOSFODNN7EXAMPLE",
    "arn": "arn:aws:iam::111122223333:role/Admin",
    "accountId": "111122223333",
    "userName": "Admin"
  }
},
"attributes": {
  "creationDate": "2025-08-25T14:45:48Z",
  "mfaAuthenticated": "false"
},
"invokedBy": "AWS Internal",
"eventTime": "2025-08-25T14:54:01Z",
"eventSource": "kms.amazonaws.com",
"eventName": "Decrypt",
"awsRegion": "us-west-2",
"sourceIPAddress": "AWS Internal",
"userAgent": "AWS Internal",
"requestParameters": {
  "encryptionContext": {
    "aws:groundstation:arn": "arn:aws:groundstation:us-
west-2:111122223333:ephemeris/bb650670-7a4b-4152-bd60-SAMPLE",
    "aws:groundstation:ground-station-id": "Ohio 1",
    "aws:s3:arn": "arn:aws:s3::customerephemerisbucket/bb650670-7a4b-4152-
bd60-SAMPLE/raw"
  },
  "encryptionAlgorithm": "SYMMETRIC_DEFAULT"
},
"responseElements": null,
"requestID": "a2f46066-49fb-461a-93cb-123456SAMPLE",
"eventID": "e997b426-e3ad-31c7-a308-123456SAMPLE",
"readOnly": true,
"resources": [
  {
    "accountId": "111122223333",
    "type": "AWS::KMS::Key",
```

```
    "ARN": "arn:aws:kms:us-  
west-2:111122223333:key/1234abcd-12ab-34cd-56ef-123456SAMPLE"  
  }  
],  
"eventType": "AwsApiCall",  
"managementEvent": true,  
"recipientAccountId": "111122223333",  
"sharedEventID": "477b568e-7f56-4f04-905c-623ff146f30d",  
"eventCategory": "Management"  
}
```

傳輸期間的資料加密 AWS Ground Station

AWS Ground Station 依預設提供加密，以在傳輸期間保護您的敏感資料。視任務設定檔組態而定，可以透過兩種方式在 AWS Ground Station 天線位置和 Amazon EC2 執行個體之間串流資料。

- AWS Ground Station 代理程式
- Dataflow 端點

每個串流資料的方法都會以不同的方式處理傳輸中的資料加密。下列幾節會描述每一種方法。

AWS Ground Station 代理程式串流

AWS Ground Station 代理程式會使用客戶受管 AWS KMS 金鑰來加密其串流。在您的 Amazon EC2 執行個體上執行的 AWS Ground Station 代理程式會自動解密串流，以提供解密的資料。

在 [streamsKmsKey](#) 參數MissionProfile中建立時，會指定用於加密串流的 AWS KMS 金鑰。授予金鑰 AWS Ground Station 存取權的所有許可都是透過連接到的 AWS KMS 金鑰政策來處理streamsKmsKey。

Dataflow 端點串流

資料流程端點串流會使用 [Datagram Transport Layer Security \(DTLS\)](#) 進行加密。這是使用自我簽署憑證完成的，不需要額外的組態。

範例任務設定檔組態

提供的範例示範如何取得公有廣播衛星，並建立支援該衛星的任務設定檔。產生的範本可協助您進行公有廣播衛星聯絡，並協助您做出有關衛星的決策。

主題

- [JPSS-1 - 公有廣播衛星 \(PBS\) - 評估](#)
- [使用 Amazon S3 資料交付的公有廣播衛星](#)
- [使用資料流程端點的公有廣播衛星 \(窄頻帶\)](#)
- [使用資料流程端點的公有廣播衛星 \(解調和解碼\)](#)
- [使用 AWS Ground Station 代理程式的公有廣播衛星 \(寬頻\)](#)

JPSS-1 - 公有廣播衛星 (PBS) - 評估

此範例區段符合 [客戶加入程序概觀](#)。它提供與的簡短相容性分析，AWS Ground Station 並為接下來的特定範例設定階段。

如 [公有廣播衛星](#) 章節所述，您可以利用可公開取得的特定衛星或衛星的通訊路徑。在本節中，我們會在 AWS Ground Station 術語中描述 [JPSS-1](#)。作為參考，我們利用 [關節極衛星系統 1 \(JPSS-1\) Spacecraft High Rate Data \(HRD\) 到 Direct Broadcast Stations \(DBS\) Radio Frequency \(RF\) Interface Control Document \(ICD\)](#) 來完成範例。此外，值得注意的是，JPSS-1 與 NORAD ID 43013 相關聯。

JPSS-1 衛星提供一個上行和三個直接下行通訊路徑，如 ICD 的圖 1-1 所示。在這四個通訊路徑中，只有單一高速率資料 (HRD) 下行通訊路徑可供公有使用。根據此，您會看到此路徑也會有更具體的資料與其相關聯。四個路徑如下所示：

- 2067.27 MHz 中心頻率的命令路徑 (上行)，資料速率為 2-128 kbps。此路徑不可公開存取。
- 中心頻率為 2247.5 MHz 的遙測路徑 (下行)，資料速率為 1-524 kbps。此路徑不可公開存取。
- SMD 路徑 (下行)，中心頻率為 26.7034 GHz，資料速率為 150-300 Mbps。此路徑不可公開存取。
- 中心頻率為 7812 MHz 時 HRD 路徑 (下行) 的 RF，資料速率為 15 Mbps。它具有 30 MHz 頻寬，並且是 right-hand-circular-polarized。當您使用加入 JPSS-1 時 AWS Ground Station，這是您可以存取的通訊路徑。此通訊路徑包含檢測科學資料、檢測工程資料、檢測遙測資料和即時太空船管家資料。

當我們比較潛在的資料路徑時，我們看到命令（上行）、遙測（下行）和 HRD（下行）路徑符合的頻率、頻寬和多通道並行使用功能 AWS Ground Station。SMD 路徑不相容，因為中心頻率超出現有接收器的範圍。如需支援功能的詳細資訊，請參閱 [AWS Ground Station 站點功能](#)。

Note

由於 SMD 路徑與不相容 AWS Ground Station，因此範例組態中不會呈現此路徑。

Note

由於 ICD 中未定義命令（上行）和遙測（下行）路徑，也無法公開使用，因此使用時提供的值是名目值。

使用 Amazon S3 資料交付的公有廣播衛星

此範例建立在 使用者指南的 [JPSS-1 - 公有廣播衛星 \(PBS\) - 評估](#) 區段中完成的分析。

在此範例中，您需要假設一個案例 -- 您想要將 HRD 通訊路徑擷取為數位中繼頻率，並將它存放以供未來批次處理。這可節省原始無線電頻率 (RF) 的階段中正交 (I/Q) 樣本在數位化之後。資料放入 Amazon S3 儲存貯體後，您可以使用任何您想要的軟體來解調和解碼資料。如需詳細處理範例，請參閱 [MathWorks 教學課程](#)。使用此範例後，您可以考慮新增 Amazon EC2 Spot 定價元件來處理資料並降低整體處理成本。

通訊路徑

本節代表 [規劃您的資料流程通訊路徑](#) 入門。

下列所有範本程式碼片段都屬於範本的資源區段 CloudFormation。

Resources:

```
# Resources that you would like to create should be placed within the Resources section.
```

Note

如需 CloudFormation 範本內容的詳細資訊，請參閱[範本章節](#)。

鑑於我們將單一通訊路徑交付至 Amazon S3 的情況，您知道您將擁有單一非同步交付路徑。根據 [非同步資料交付](#) 區段，您必須定義 Amazon S3 儲存貯體。

```
# The S3 bucket where AWS Ground Station will deliver the downlinked data.
GroundStationS3DataDeliveryBucket:
  Type: AWS::S3::Bucket
  DeletionPolicy: Retain
  UpdateReplacePolicy: Retain
  Properties:
    # Results in a bucket name formatted like: aws-groundstation-data-{account id}-
    {region}-{random 8 character string}
    BucketName: !Join ["-", ["aws-groundstation-data", !Ref AWS::AccountId, !Ref
AWS::Region, !Select [0, !Split ["-", !Select [2, !Split ["/", !Ref AWS::StackId]]]]]]
```

此外，您將需要建立適當的角色和政策，以允許使用儲存貯 AWS Ground Station 體。

```
# The IAM role that AWS Ground Station will assume to have permission find and write
# data to your S3 bucket.
GroundStationS3DataDeliveryRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Statement:
        - Action:
            - 'sts:AssumeRole'
          Effect: Allow
          Principal:
            Service:
              - groundstation.amazonaws.com
          Condition:
            StringEquals:
              "aws:SourceAccount": !Ref AWS::AccountId
            ArnLike:
```

```

    "aws:SourceArn": !Sub "arn:aws:groundstation:${AWS::Region}:
    ${AWS::AccountId}:config/s3-recording/*"

# The S3 bucket policy that defines what actions AWS Ground Station can perform on
your S3 bucket.
GroundStationS3DataDeliveryBucketPolicy:
  Type: AWS::IAM::Policy
  Properties:
    PolicyDocument:
      Statement:
        - Action:
            - 's3:GetBucketLocation'
          Effect: Allow
          Resource:
            - !GetAtt GroundStationS3DataDeliveryBucket.Arn
        - Action:
            - 's3:PutObject'
          Effect: Allow
          Resource:
            - !Join [ "/", [ !GetAtt GroundStationS3DataDeliveryBucket.Arn, "*" ] ]
    PolicyName: GroundStationS3DataDeliveryPolicy
  Roles:
    - !Ref GroundStationS3DataDeliveryRole

```

AWS Ground Station 組態

本節代表[建立組態](#)入門。

您需要使用追蹤組態，才能在上使用自動追蹤設定偏好設定。選取 PREFERRED 做為自動追蹤可以改善訊號品質，但由於 JPSS-1 ephemeris 品質足夠，因此不需要符合訊號品質。

```

TrackingConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Tracking Config"
    ConfigData:
      TrackingConfig:
        Autotrack: "PREFERRED"

```

根據通訊路徑，您需要定義天線下行組態來代表衛星部分，以及 s3 記錄來參考您剛建立的 Amazon S3 儲存貯體。

```
# The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
# downlink data from your satellite.
JpssDownlinkDigIfAntennaConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Downlink DigIF Antenna Config"
    ConfigData:
      AntennaDownlinkConfig:
        SpectrumConfig:
          Bandwidth:
            Units: "MHz"
            Value: 30
          CenterFrequency:
            Units: "MHz"
            Value: 7812
          Polarization: "RIGHT_HAND"

# The AWS Ground Station S3 Recording Config that defines the S3 bucket and IAM role
to use
# when AWS Ground Station delivers the downlink data.
S3RecordingConfig:
  Type: AWS::GroundStation::Config
  DependsOn: GroundStationS3DataDeliveryBucketPolicy
  Properties:
    Name: "JPSS S3 Recording Config"
    ConfigData:
      S3RecordingConfig:
        BucketArn: !GetAtt GroundStationS3DataDeliveryBucket.Arn
        RoleArn: !GetAtt GroundStationS3DataDeliveryRole.Arn
```

AWS Ground Station 任務描述檔

本節代表[建立任務描述檔](#)入門。

現在您已有相關聯的組態，您可以使用它們來建構資料流程。您將使用其餘參數的預設值。

```
# The AWS Ground Station Mission Profile that groups the above configurations to
define how to downlink data.
JpssAsynchMissionProfile:
  Type: AWS::GroundStation::MissionProfile
  Properties:
    Name: "43013 JPSS Asynchronous Data"
    MinimumViableContactDurationSeconds: 180
    TrackingConfigArn: !Ref TrackingConfig
    DataflowEdges:
      - Source: !Ref JpssDownlinkDigIfAntennaConfig
        Destination: !Ref S3RecordingConfig
```

將它放在一起

使用上述資源，您現在可以排程 JPSS-1 聯絡人，以便從任何已加入的 進行非同步資料交付 AWS Ground Station [AWS Ground Station 位置](#)。

以下是完整的 CloudFormation 範本，其中包含本節所述的所有資源，合併為可以直接用於 的單一範本 CloudFormation。

名為 的 CloudFormation 範本 AquaSnppJpss-1TerraDigIfS3DataDelivery.yml 包含 Amazon S3 儲存貯體，以及排程聯絡人和接收 VITA-49 Signal/IP 直接廣播資料所需的 AWS Ground Station 資源。

如果 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 未加入您的帳戶，請參閱 [加入衛星](#)。

Note

您可以使用有效的 AWS 登入資料來存取客戶加入 Amazon S3 儲存貯體，以存取範本。以下連結使用區域 Amazon S3 儲存貯體。變更 us-west-2 區域碼，代表您要建立 CloudFormation 堆疊的對應區域。

此外，下列指示使用 YAML。但是，範本同時提供 YAML 和 JSON 格式。若要使用 JSON，請在下載範本 .json 時將 .yaml 副檔名取代為。

若要使用 下載範本 AWS CLI，請使用下列命令：

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/
AquaSnppJpss-1TerraDigIfS3DataDelivery.yml .
```

您可以在瀏覽器中瀏覽至以下 URL，以在主控台中檢視和下載範本：

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-west-2/AquaSnppJpss-1TerraDigIfS3DataDelivery.yml
```

您可以使用 CloudFormation 以下連結直接在 中指定範本：

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/AquaSnppJpss-1TerraDigIfS3DataDelivery.yml
```

使用資料流程端點的公有廣播衛星（窄頻帶）

此範例建立在使用者指南的 [JPSS-1 - 公有廣播衛星 \(PBS\) - 評估](#) 區段中完成的分析。

若要完成此範例，您需要假設一個案例 -- 您想要將 HRD 通訊路徑擷取為數位中繼頻率 (DigIF)，並使用 SDR 在 Amazon EC2 執行個體上的資料流程端點應用程式收到時加以處理。

通訊路徑

本節代表 [規劃您的資料流程通訊路徑](#) 入門。在此範例中，您將在 CloudFormation 範本中建立兩個區段：參數和資源區段。

Note

如需 CloudFormation 範本內容的詳細資訊，請參閱 [範本章節](#)。

針對參數區段，您將新增下列參數。透過 CloudFormation 主控台建立堆疊時，您將指定這些值。

Parameters:

EC2Key:

Description: The SSH key used to access the EC2 receiver instance. Choose any SSH key if you are not creating an EC2 receiver instance. For instructions on how to create an SSH key see <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-key-pairs.html>

Type: AWS::EC2::KeyPair::KeyName

ConstraintDescription: must be the name of an existing EC2 KeyPair.

ReceiverAMI:

Description: The Ground Station DDX AMI ID you want to use. Please note that AMIs are region specific. For instructions on how to retrieve an AMI see <https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-configuration.html#dataflows.ec2-configuration.amis>

Type: AWS::EC2::Image::Id

Note

您需要建立金鑰對，並提供 Amazon EC2 EC2Key 參數的名稱。請參閱[為您的 Amazon EC2 執行個體建立金鑰對](#)。

此外，建立 CloudFormation 堆疊時，您需要提供正確的區域特定 AMI ID。請參閱[AWS Ground Station Amazon Machine Image AMIs](#)。

剩餘的範本程式碼片段屬於範本的資源區段 CloudFormation 。

Resources:

```
# Resources that you would like to create should be placed within the resource section.
```

鑒於我們將單一通訊路徑交付至 EC2 執行個體的情況，您將擁有單一同步交付路徑。根據[同步資料交付](#)區段，您必須使用資料流程端點應用程式設定 Amazon EC2 執行個體，並建立一或多個資料流程端點群組。

```
# The EC2 instance that will send/receive data to/from your satellite using AWS
Ground Station.
ReceiverInstance:
  Type: AWS::EC2::Instance
  Properties:
    DisableApiTermination: false
    IamInstanceProfile: !Ref GeneralInstanceProfile
    ImageId: !Ref ReceiverAMI
    InstanceType: m5.4xlarge
    KeyName: !Ref EC2Key
    Monitoring: true
    PlacementGroupName: !Ref ClusterPlacementGroup
    SecurityGroupIds:
      - Ref: InstanceSecurityGroup
    SubnetId: !Ref ReceiverSubnet
```

```

BlockDeviceMappings:
  - DeviceName: /dev/xvda
    Ebs:
      VolumeType: gp2
      VolumeSize: 40
Tags:
  - Key: Name
    Value: !Join [ "-", [ "Receiver" , !Ref "AWS::StackName" ] ]
UserData:
  Fn::Base64:
    |
    #!/bin/bash
    exec > >(tee /var/log/user-data.log|logger -t user-data -s 2>/dev/console)
2>&1
    echo `date +%F %R:%S` ` "INFO: Logging Setup" >&2

    GROUND_STATION_DIR="/opt/aws/groundstation"
    GROUND_STATION_BIN_DIR="${GROUND_STATION_DIR}/bin"
    STREAM_CONFIG_PATH="${GROUND_STATION_DIR}/customer_stream_config.json"

    echo "Creating ${STREAM_CONFIG_PATH}"
    cat << STREAM_CONFIG > "${STREAM_CONFIG_PATH}"
    {
      "ddx_streams": [
        {
          "streamName": "Downlink",
          "maximumWanRate": 4000000000,
          "lanConfigDevice": "lo",
          "lanConfigPort": 50000,
          "wanConfigDevice": "eth1",
          "wanConfigPort": 55888,
          "isUplink": false
        }
      ]
    }
    STREAM_CONFIG

    echo "Waiting for dataflow endpoint application to start"
    while netstat -lnt | awk '$4 ~ /:80$/ {exit 1}'; do sleep 10; done

    echo "Configuring dataflow endpoint application streams"
    python "${GROUND_STATION_BIN_DIR}/configure_streams.py" --configFileName
"${STREAM_CONFIG_PATH}"
    sleep 2

```

```

python "${GROUND_STATION_BIN_DIR}/save_default_config.py"

exit 0

# The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
# Station will use to send/receive data to/from your satellite.
DataflowEndpointGroup:
  Type: AWS::GroundStation::DataflowEndpointGroup
  Properties:
    ContactPostPassDurationSeconds: 180
    ContactPrePassDurationSeconds: 120
    EndpointDetails:
      - Endpoint:
          Name: !Join [ "-", [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
          Address:
            Name: !GetAtt ReceiverInstanceNetworkInterface.PrimaryPrivateIpAddress
            Port: 55888
          SecurityDetails:
            SecurityGroupIds:
              - Ref: "DataflowEndpointSecurityGroup"
            SubnetIds:
              - !Ref ReceiverSubnet
            RoleArn: !GetAtt DataDeliveryServiceRole.Arn

# The security group for your EC2 instance.
InstanceSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: AWS Ground Station receiver instance security group.
    VpcId: !Ref ReceiverVPC
    SecurityGroupIngress:
      # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
      - IpProtocol: udp
        FromPort: 55888
        ToPort: 55888
        SourceSecurityGroupId: !Ref DataflowEndpointSecurityGroup
        Description: "AWS Ground Station Downlink Stream"

# The security group that the ENI created by AWS Ground Station belongs to.
DataflowEndpointSecurityGroup:
  Type: AWS::EC2::SecurityGroup

```

```
Properties:
  GroupDescription: Security Group for AWS Ground Station registration of Dataflow
Endpoint Groups
  VpcId: !Ref ReceiverVPC
  SecurityGroupEgress:
    - IpProtocol: udp
      FromPort: 55888
      ToPort: 55888
      CidrIp: 10.0.0.0/8
      Description: "AWS Ground Station Downlink Stream To 10/8"
    - IpProtocol: udp
      FromPort: 55888
      ToPort: 55888
      CidrIp: 172.16.0.0/12
      Description: "AWS Ground Station Downlink Stream To 172.16/12"
    - IpProtocol: udp
      FromPort: 55888
      ToPort: 55888
      CidrIp: 192.168.0.0/16
      Description: "AWS Ground Station Downlink Stream To 192.168/16"

# The placement group in which your EC2 instance is placed.
ClusterPlacementGroup:
  Type: AWS::EC2::PlacementGroup
  Properties:
    Strategy: cluster

ReceiverVPC:
  Type: AWS::EC2::VPC
  Properties:
    CidrBlock: "10.0.0.0/16"
  Tags:
    - Key: "Name"
      Value: "AWS Ground Station - PBS to dataflow endpoint Example VPC"
    - Key: "Description"
      Value: "VPC for EC2 instance receiving AWS Ground Station data"

ReceiverSubnet:
  Type: AWS::EC2::Subnet
  Properties:
    # Ensure your CidrBlock will always have at least one available IP address per
    dataflow endpoint.
    # See https://docs.aws.amazon.com/vpc/latest/userguide/subnet-sizing.html for
    subnet sizing guidelines.
```

```
CidrBlock: "10.0.0.0/24"
Tags:
  - Key: "Name"
    Value: "AWS Ground Station - PBS to dataflow endpoint Example Subnet"
  - Key: "Description"
    Value: "Subnet for EC2 instance receiving AWS Ground Station data"
VpcId: !Ref ReceiverVPC

# An ENI providing a fixed IP address for AWS Ground Station to connect to.
ReceiverInstanceNetworkInterface:
  Type: AWS::EC2::NetworkInterface
  Properties:
    Description: Floating network interface providing a fixed IP address for AWS
Ground Station to connect to.
    GroupSet:
      - !Ref InstanceSecurityGroup
    SubnetId: !Ref ReceiverSubnet

# Attach the ENI to the EC2 instance.
ReceiverInstanceInterfaceAttachment:
  Type: AWS::EC2::NetworkInterfaceAttachment
  Properties:
    DeleteOnTermination: false
    DeviceIndex: "1"
    InstanceId: !Ref ReceiverInstance
    NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface
```

此外，您還需要建立適當的政策和角色，以允許在帳戶中 AWS Ground Station 建立彈性網路界面 (ENI)。

```
# AWS Ground Station assumes this role to create/delete ENIs in your account in order
to stream data.
DataDeliveryServiceRole:
  Type: AWS::IAM::Role
  Properties:
    Policies:
      - PolicyDocument:
          Statement:
            - Action:
                - ec2:CreateNetworkInterface
                - ec2>DeleteNetworkInterface
```

```
    - ec2:CreateNetworkInterfacePermission
    - ec2>DeleteNetworkInterfacePermission
    - ec2:DescribeSubnets
    - ec2:DescribeVpcs
    - ec2:DescribeSecurityGroups
    Effect: Allow
    Resource: '*'
    Version: '2012-10-17'
    PolicyName: DataDeliveryServicePolicy
AssumeRolePolicyDocument:
    Version: 2012-10-17
    Statement:
        - Effect: Allow
          Principal:
            Service:
              - groundstation.amazonaws.com
          Action:
            - sts:AssumeRole

# The EC2 instance assumes this role.
InstanceRole:
    Type: AWS::IAM::Role
    Properties:
        AssumeRolePolicyDocument:
            Version: "2012-10-17"
            Statement:
                - Effect: "Allow"
                  Principal:
                    Service:
                      - "ec2.amazonaws.com"
                  Action:
                    - "sts:AssumeRole"
        Path: "/"
        ManagedPolicyArns:
            - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
            - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
            - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
            - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM

# The instance profile for your EC2 instance.
GeneralInstanceProfile:
    Type: AWS::IAM::InstanceProfile
    Properties:
        Roles:
```

```
- !Ref InstanceRole
```

AWS Ground Station 組態

本節代表[建立組態](#)入門。

您需要使用追蹤組態，才能在上使用自動追蹤來設定偏好設定。選擇 PREFERRED 作為自動追蹤可以改善訊號品質，但由於足夠的 JPSS-1 暫時性品質，因此不需要滿足訊號品質。

```
TrackingConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Tracking Config"
    ConfigData:
      TrackingConfig:
        Autotrack: "PREFERRED"
```

根據通訊路徑，您需要定義天線下行組態來代表衛星部分，以及定義端點詳細資訊之資料流程端點群組的資料流程端點組態。

```
# The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
# downlink data from your satellite.
SnppJpssDownlinkDigIfAntennaConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "SNPP JPSS Downlink DigIF Antenna Config"
    ConfigData:
      AntennaDownlinkConfig:
        SpectrumConfig:
          Bandwidth:
            Units: "MHz"
            Value: 30
          CenterFrequency:
            Units: "MHz"
            Value: 7812
          Polarization: "RIGHT_HAND"
```

```
# The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
# from your satellite.
DownlinkDigIfEndpointConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "Aqua SNPP JPSS Downlink DigIF Endpoint Config"
    ConfigData:
      DataflowEndpointConfig:
        DataflowEndpointName: !Join [ "-", [ !Ref "AWS::StackName" , "Downlink" ] ]
        DataflowEndpointRegion: !Ref AWS::Region
```

AWS Ground Station 任務描述檔

本節代表[建立任務描述檔](#)入門。

現在您已有相關聯的組態，您可以使用它們來建構資料流程。您將使用其餘參數的預設值。

```
# The AWS Ground Station Mission Profile that groups the above configurations to
define how to
# uplink and downlink data to your satellite.
SnpjpsMissionProfile:
  Type: AWS::GroundStation::MissionProfile
  Properties:
    Name: "37849 SNPP And 43013 JPSS"
    ContactPrePassDurationSeconds: 120
    ContactPostPassDurationSeconds: 60
    MinimumViableContactDurationSeconds: 180
    TrackingConfigArn: !Ref TrackingConfig
    DataflowEdges:
      - Source: !Ref SnpjpsDownlinkDigIfAntennaConfig
        Destination: !Ref DownlinkDigIfEndpointConfig
```

將它放在一起

使用上述資源，您現在可以排程 JPSS-1 聯絡人，以便從任何已加入的 同步資料交付 AWS Ground Station [AWS Ground Station 位置](#)。

以下是完整的 CloudFormation 範本，其中包含本節所述的所有資源，合併為可以直接使用的單一範本 CloudFormation。

名為 `AquaSnppJpssTerraDigIF.yml` 的 CloudFormation 範本旨在讓您快速存取，以開始接收 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 衛星的數位化中繼頻率 (DigIF) 資料。它包含 Amazon EC2 執行個體和接收原始 DigIF 直接廣播資料所需的 CloudFormation 資源。

如果 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 未加入您的帳戶，請參閱 [加入衛星](#)。

Note

您可以使用有效的 AWS 登入資料存取客戶加入 Amazon S3 儲存貯體來存取範本。以下連結使用區域 Amazon S3 儲存貯體。變更 `us-west-2` 區域碼以代表您要在其中建立 CloudFormation 堆疊的對應區域。

此外，下列指示使用 YAML。但是，範本同時提供 YAML 和 JSON 格式。若要使用 JSON，請在下載範本 `.json` 時將 `.yml` 副檔名取代為。

若要使用下載範本 AWS CLI，請使用下列命令：

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/AquaSnppJpssTerraDigIF.yml .
```

您可以在瀏覽器中瀏覽至以下 URL，以在主控台中檢視和下載範本：

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-west-2/AquaSnppJpssTerraDigIF.yml
```

您可以使用以下連結直接在 CloudFormation 中指定範本：

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/AquaSnppJpssTerraDigIF.yml
```

範本定義了哪些其他資源？

`AquaSnppJpssTerraDigIF` 範本包含下列其他資源：

- (選用) CloudWatch Event Triggers - AWS Lambda 使用聯絡 AWS Ground Station 前後由傳送的 CloudWatch Events 觸發的函數。AWS Lambda 函數將啟動並選擇性地停止您的接收者執行個體。

- (選用) 聯絡的 EC2 驗證 - 使用 Lambda 為具有 SNS 通知的聯絡設定 Amazon EC2 執行個體驗證系統的選項 (SNS)。請務必注意，這可能會產生費用，這取決於您目前的用量。
- Ground Station Amazon Machine Image Retrieval Lambda - 選擇執行個體中安裝的軟體以及您選擇的 AMI 的選項。軟體選項包括 DDX 2.6.2 Only 和 DDX 2.6.2 with qRadio 3.6.0。隨著其他軟體更新和功能發佈，這些選項將繼續擴展。
- 其他任務描述檔 - 其他公有廣播衛星 (Aqua、SNPP 和 Terra) 的任務描述檔。
- 其他天線下行組態 - 其他公有廣播衛星 (Aqua、SNPP 和 Terra) 的天線下行組態。

在此範本中，衛星的值和參數已經產生。這些參數可讓您輕鬆地 AWS Ground Station 立即與這些衛星搭配使用。您不需要設定自己的值，即可在使用此範本 AWS Ground Station 時使用。但是，您可以自訂這些值，讓範本適用於您的使用案例。

我可以在什麼地方接收我的資料？

資料流程端點群組的設定，是使用以範本的一部分建立的接收器執行個體網路界面。接收者執行個體使用資料流程端點應用程式，在資料流程端點定義的連接埠 AWS Ground Station 上接收來自的資料串流。接收到以後，資料就可以透過 UDP 連接埠 50000 在接收器執行個體的回送介面卡上使用。如需設定資料流程端點群組的詳細資訊，請參閱 [AWS::GroundStation::DataflowEndpointGroup](#)。

使用資料流程端點的公有廣播衛星（解調和解碼）

此範例會建置使用者指南 [JPSS-1 - 公有廣播衛星 \(PBS\) - 評估](#) 區段中完成的分析。

若要完成此範例，您需要假設一個案例 -- 您想要使用資料流程端點，以解調和解碼的直接廣播資料的形式擷取 HRD 通訊路徑。如果您計劃使用 NASA Direct Readout Labs 軟體 (RT-STPS 和 IPOPP) 處理資料，則此範例是一個很好的起點。

通訊路徑

本節代表 [規劃您的資料流程通訊路徑](#) 入門。在此範例中，您將在 CloudFormation 範本中建立兩個區段：參數和資源區段。

Note

如需 CloudFormation 範本內容的詳細資訊，請參閱 [範本章節](#)。

針對參數區段，您將新增下列參數。透過 CloudFormation 主控台建立堆疊時，您將指定這些值。

Parameters:**EC2Key:**

Description: The SSH key used to access the EC2 receiver instance. Choose any SSH key if you are not creating an EC2 receiver instance. For instructions on how to create an SSH key see <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-key-pairs.html>

Type: AWS::EC2::KeyPair::KeyName

ConstraintDescription: must be the name of an existing EC2 KeyPair.

ReceiverAMI:

Description: The Ground Station DDX AMI ID you want to use. Please note that AMIs are region specific. For instructions on how to retrieve an AMI see <https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-configuration.html#dataflows.ec2-configuration.amis>

Type: AWS::EC2::Image::Id

Note

您需要建立金鑰對，並提供 Amazon EC2 EC2Key 參數的名稱。請參閱[為您的 Amazon EC2 執行個體建立金鑰對](#)。

此外，建立 CloudFormation 堆疊時，您需要提供正確的區域特定 AMI ID。請參閱[AWS Ground Station Amazon Machine Image AMIs](#)。

剩餘的範本程式碼片段屬於範本的資源區段 CloudFormation 。

Resources:

```
# Resources that you would like to create should be placed within the resource section.
```

鑑於我們將單一通訊路徑交付至 EC2 執行個體的情況，您將擁有單一同步交付路徑。根據[同步資料交付](#)區段，您必須使用資料流程端點應用程式設定 Amazon EC2 執行個體，並建立一或多個資料流程端點群組。

```
# The EC2 instance that will send/receive data to/from your satellite using AWS Ground Station.
ReceiverInstance:
```

```
Type: AWS::EC2::Instance
```

```
Properties:
```

```
  DisableApiTermination: false
  IamInstanceProfile: !Ref GeneralInstanceProfile
  ImageId: !Ref ReceiverAMI
  InstanceType: m5.4xlarge
  KeyName: !Ref EC2Key
  Monitoring: true
  PlacementGroupName: !Ref ClusterPlacementGroup
  SecurityGroupIds:
```

```
  - Ref: InstanceSecurityGroup
```

```
SubnetId: !Ref ReceiverSubnet
```

```
BlockDeviceMappings:
```

```
  - DeviceName: /dev/xvda
```

```
    Ebs:
```

```
      VolumeType: gp2
```

```
      VolumeSize: 40
```

```
Tags:
```

```
  - Key: Name
```

```
    Value: !Join [ "-", [ "Receiver", !Ref "AWS::StackName" ] ]
```

```
UserData:
```

```
  Fn::Base64:
```

```
    |
```

```
    #!/bin/bash
```

```
    exec > >(tee /var/log/user-data.log|logger -t user-data -s 2>/dev/console)
```

```
2>&1
```

```
    echo `date +%F %R:%S` "INFO: Logging Setup" >&2
```

```
    GROUND_STATION_DIR="/opt/aws/groundstation"
```

```
    GROUND_STATION_BIN_DIR="${GROUND_STATION_DIR}/bin"
```

```
    STREAM_CONFIG_PATH="${GROUND_STATION_DIR}/customer_stream_config.json"
```

```
    echo "Creating ${STREAM_CONFIG_PATH}"
```

```
    cat << STREAM_CONFIG > "${STREAM_CONFIG_PATH}"
```

```
    {
```

```
      "ddx_streams": [
```

```
        {
```

```
          "streamName": "Downlink",
```

```
          "maximumWanRate": 4000000000,
```

```
          "lanConfigDevice": "lo",
```

```
          "lanConfigPort": 50000,
```

```
          "wanConfigDevice": "eth1",
```

```
          "wanConfigPort": 55888,
```

```
          "isUplink": false
```

```

    }
  ]
}
STREAM_CONFIG

echo "Waiting for dataflow endpoint application to start"
while netstat -lnt | awk '$4 ~ /:80$/ {exit 1}'; do sleep 10; done

echo "Configuring dataflow endpoint application streams"
python "${GROUND_STATION_BIN_DIR}/configure_streams.py" --configFileName
"${STREAM_CONFIG_PATH}"
sleep 2
python "${GROUND_STATION_BIN_DIR}/save_default_config.py"

exit 0

```

```

# The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
# Station will use to send/receive data to/from your satellite.
DataflowEndpointGroup:
  Type: AWS::GroundStation::DataflowEndpointGroup
  Properties:
    ContactPostPassDurationSeconds: 180
    ContactPrePassDurationSeconds: 120
    EndpointDetails:
      - Endpoint:
          Name: !Join [ "-", [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
          Address:
            Name: !GetAtt ReceiverInstanceNetworkInterface.PrimaryPrivateIpAddress
            Port: 55888
          SecurityDetails:
            SecurityGroupIds:
              - Ref: "DataflowEndpointSecurityGroup"
            SubnetIds:
              - !Ref ReceiverSubnet
            RoleArn: !GetAtt DataDeliveryServiceRole.Arn

# The security group that the ENI created by AWS Ground Station belongs to.
DataflowEndpointSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:

```

```
GroupDescription: Security Group for AWS Ground Station registration of Dataflow
Endpoint Groups
  VpcId: !Ref ReceiverVPC
  SecurityGroupEgress:
    - IpProtocol: udp
      FromPort: 55888
      ToPort: 55888
      CidrIp: 10.0.0.0/8
      Description: "AWS Ground Station Downlink Stream To 10/8"
    - IpProtocol: udp
      FromPort: 55888
      ToPort: 55888
      CidrIp: 172.16.0.0/12
      Description: "AWS Ground Station Downlink Stream To 172.16/12"
    - IpProtocol: udp
      FromPort: 55888
      ToPort: 55888
      CidrIp: 192.168.0.0/16
      Description: "AWS Ground Station Downlink Stream To 192.168/16"

# The placement group in which your EC2 instance is placed.
ClusterPlacementGroup:
  Type: AWS::EC2::PlacementGroup
  Properties:
    Strategy: cluster

# The security group for your EC2 instance.
InstanceSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: AWS Ground Station receiver instance security group.
    VpcId: !Ref ReceiverVPC
    SecurityGroupIngress:
      # To allow SSH access to the instance, add another rule allowing tcp port 22
      from your CidrIp
      - IpProtocol: tcp
        FromPort: 22
        ToPort: 22
        SourceSecurityGroupId: !Ref DataflowEndpointSecurityGroup
        Description: "AWS Ground Station Downlink Stream"

ReceiverVPC:
  Type: AWS::EC2::VPC
  Properties:
```

```
CidrBlock: "10.0.0.0/16"
Tags:
  - Key: "Name"
    Value: "AWS Ground Station - PBS to dataflow endpoint Demod Decode Example
VPC"
  - Key: "Description"
    Value: "VPC for EC2 instance receiving AWS Ground Station data"

ReceiverSubnet:
  Type: AWS::EC2::Subnet
  Properties:
    CidrBlock: "10.0.0.0/24"
    Tags:
      - Key: "Name"
        Value: "AWS Ground Station - PBS to dataflow endpoint Demod Decode Example
Subnet"
      - Key: "Description"
        Value: "Subnet for EC2 instance receiving AWS Ground Station data"
    VpcId: !Ref ReceiverVPC

# An ENI providing a fixed IP address for AWS Ground Station to connect to.
ReceiverInstanceNetworkInterface:
  Type: AWS::EC2::NetworkInterface
  Properties:
    Description: Floating network interface providing a fixed IP address for AWS
Ground Station to connect to.
    GroupSet:
      - !Ref InstanceSecurityGroup
    SubnetId: !Ref ReceiverSubnet

# Attach the ENI to the EC2 instance.
ReceiverInstanceInterfaceAttachment:
  Type: AWS::EC2::NetworkInterfaceAttachment
  Properties:
    DeleteOnTermination: false
    DeviceIndex: "1"
    InstanceId: !Ref ReceiverInstance
    NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

# The instance profile for your EC2 instance.
GeneralInstanceProfile:
  Type: AWS::IAM::InstanceProfile
  Properties:
    Roles:
```

```
- !Ref InstanceRole
```

您也需要適當的政策、角色和設定檔，AWS Ground Station 才能允許在帳戶中建立彈性網路界面 (ENI)。

```
# AWS Ground Station assumes this role to create/delete ENIs in your account in order
to stream data.
DataDeliveryServiceRole:
  Type: AWS::IAM::Role
  Properties:
    Policies:
      - PolicyDocument:
          Statement:
            - Action:
                - ec2:CreateNetworkInterface
                - ec2>DeleteNetworkInterface
                - ec2:CreateNetworkInterfacePermission
                - ec2>DeleteNetworkInterfacePermission
                - ec2:DescribeSubnets
                - ec2:DescribeVpcs
                - ec2:DescribeSecurityGroups
              Effect: Allow
              Resource: '*'
            Version: '2012-10-17'
          PolicyName: DataDeliveryServicePolicy
    AssumeRolePolicyDocument:
      Version: 2012-10-17
      Statement:
        - Effect: Allow
          Principal:
            Service:
              - groundstation.amazonaws.com
          Action:
            - sts:AssumeRole

# The EC2 instance assumes this role.
InstanceRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
```

```

Statement:
  - Effect: "Allow"
    Principal:
      Service:
        - "ec2.amazonaws.com"
    Action:
      - "sts:AssumeRole"
Path: "/"
ManagedPolicyArns:
  - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
  - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
  - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
  - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM

```

AWS Ground Station 組態

本節代表 使用者指南 [建立組態](#) 的。

您需要使用追蹤組態，才能在上使用自動追蹤設定偏好設定。選取 PREFERRED 做為自動追蹤可以改善訊號品質，但由於 JPSS-1 ephemeris 品質足夠，因此不需要符合訊號品質。

```

TrackingConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Tracking Config"
    ConfigData:
      TrackingConfig:
        Autotrack: "PREFERRED"

```

根據通訊路徑，您需要定義 antenna-downlink-demod-decode 組態來代表衛星部分，以及資料流程端點組態來參考定義端點詳細資訊的資料流程端點群組。

Note

如需如何設定 DemodulationConfig、和 值的詳細資訊 DecodeConfig，請參閱 [天線下行解調解碼組態](#)。

```
# The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
# downlink data from your satellite.
JpssDownlinkDemodDecodeAntennaConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Downlink Demod Decode Antenna Config"
    ConfigData:
      AntennaDownlinkDemodDecodeConfig:
        SpectrumConfig:
          CenterFrequency:
            Value: 7812
            Units: "MHz"
          Polarization: "RIGHT_HAND"
          Bandwidth:
            Value: 30
            Units: "MHz"
        DemodulationConfig:
          UnvalidatedJSON: '{
            "type":"QPSK",
            "qpsk":{
              "carrierFrequencyRecovery":{
                "centerFrequency":{
                  "value":7812,
                  "units":"MHz"
                },
                "range":{
                  "value":250,
                  "units":"kHz"
                }
              },
              "symbolTimingRecovery":{
                "symbolRate":{
                  "value":15,
                  "units":"Msps"
                },
                "range":{
                  "value":0.75,
                  "units":"ksps"
                }
              },
              "matchedFilter":{
                "type":"ROOT_RAISED_COSINE",
```

```
        "rolloffFactor":0.5
      }
    }
  }'
DecodeConfig:
  UnvalidatedJSON: '{
    "edges":[
      {
        "from":"I-Ingress",
        "to":"IQ-Recombiner"
      },
      {
        "from":"Q-Ingress",
        "to":"IQ-Recombiner"
      },
      {
        "from":"IQ-Recombiner",
        "to":"CcsdsViterbiDecoder"
      },
      {
        "from":"CcsdsViterbiDecoder",
        "to":"NrzMDecoder"
      },
      {
        "from":"NrzMDecoder",
        "to":"UncodedFramesEgress"
      }
    ],
    "nodeConfigs":{
      "I-Ingress":{
        "type":"CODED_SYMBOLS_INGRESS",
        "codedSymbolsIngress":{
          "source":"I"
        }
      },
      "Q-Ingress":{
        "type":"CODED_SYMBOLS_INGRESS",
        "codedSymbolsIngress":{
          "source":"Q"
        }
      },
      "IQ-Recombiner":{
        "type":"IQ_RECOMBINER"
```

```

    },
    "CcsdsViterbiDecoder":{
      "type":"CCSDS_171_133_VITERBI_DECODER",
      "ccsds171133ViterbiDecoder":{
        "codeRate":"ONE_HALF"
      }
    },
    "NrzmDecoder":{
      "type":"NRZ_M_DECODER"
    },
    "UncodedFramesEgress":{
      "type":"UNCODED_FRAMES_EGRESS"
    }
  }
}'

```

```

# The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
# from your satellite.
DownlinkDemodDecodeEndpointConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "Aqua SNPP JPSS Downlink Demod Decode Endpoint Config"
    ConfigData:
      DataflowEndpointConfig:
        DataflowEndpointName: !Join [ "-", [ !Ref "AWS::StackName" , "Downlink" ] ]
        DataflowEndpointRegion: !Ref AWS::Region

```

AWS Ground Station 任務描述檔

本節代表 使用者指南 [建立任務描述檔](#)。

現在您已有相關聯的組態，您可以使用它們來建構資料流程。您將使用其餘參數的預設值。

```

# The AWS Ground Station Mission Profile that groups the above configurations to
define how to
# uplink and downlink data to your satellite.
SnppJpssMissionProfile:

```

```
Type: AWS::GroundStation::MissionProfile
Properties:
  Name: "37849 SNPP And 43013 JPSS"
  ContactPrePassDurationSeconds: 120
  ContactPostPassDurationSeconds: 60
  MinimumViableContactDurationSeconds: 180
  TrackingConfigArn: !Ref TrackingConfig
  DataflowEdges:
    - Source: !Join [ "/", [ !Ref JpssDownlinkDemodDecodeAntennaConfig,
      "UncodedFramesEgress" ] ]
      Destination: !Ref DownlinkDemodDecodeEndpointConfig
```

將它放在一起

使用上述資源，您現在可以排程 JPSS-1 聯絡人，以便從任何已加入的 同步資料交付 AWS Ground Station [AWS Ground Station 位置](#)。

以下是完整的 CloudFormation 範本，其中包含本節所述的所有資源，合併為可以直接使用的單一範本 CloudFormation。

名為 `AquaSnppJpss.yml` 的 CloudFormation 範本旨在讓您快速存取，以開始接收 Aqua、SNPP 和 JPSS-1/NOAA-20 衛星的資料。它包含 Amazon EC2 執行個體和必要的 AWS Ground Station 資源，以排程聯絡人並接收解調和解碼的直接廣播資料。

如果 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 未加入您的帳戶，請參閱 [加入衛星](#)。

Note

您可以使用有效的 AWS 登入資料來存取客戶加入 Amazon S3 儲存貯體，以存取範本。以下連結使用區域 Amazon S3 儲存貯體。變更 `us-west-2` 區域碼，代表您要建立 CloudFormation 堆疊的對應區域。

此外，下列指示使用 YAML。但是，範本同時提供 YAML 和 JSON 格式。若要使用 JSON，請在下載範本 `.json` 時將 `.yml` 副檔名取代為。

若要使用 下載範本 AWS CLI，請使用下列命令：

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/AquaSnppJpss.yml .
```

您可以在瀏覽器中瀏覽至以下 URL，以在主控台中檢視和下載範本：

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-west-2/AquaSnppJpss.yml
```

您可以使用 CloudFormation 以下連結直接在 中指定範本：

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/AquaSnppJpss.yml
```

範本定義了哪些其他資源？

AquaSnppJpss 範本包含下列其他資源：

- (選用) CloudWatch Event Triggers - AWS Lambda 使用聯絡 AWS Ground Station 前後由 傳送的 CloudWatch Events 觸發的函數。AWS Lambda 函數將啟動並選擇性地停止您的接收者執行個體。
- (選用) 聯絡的 EC2 驗證 - 使用 Lambda 為具有 SNS 通知的聯絡設定 Amazon EC2 執行個體驗證系統的選項 (SNS 通知)。請務必注意，這可能會根據您目前的使用量產生費用。
- Ground Station Amazon Machine Image Retrieval Lambda - 選擇執行個體中安裝的軟體以及您選擇的 AMI 的選項。軟體選項包括 DDX 2.6.2 Only 和 DDX 2.6.2 with qRadio 3.6.0。如果您想要使用寬頻 DigIF 資料交付和 AWS Ground Station 代理程式，請參閱 [使用 AWS Ground Station 代理程式的公有廣播衛星 \(寬頻\)](#)。隨著其他軟體更新和功能發佈，這些選項將繼續擴展。
- 其他任務描述檔 - 其他公有廣播衛星 (Aqua、SNPP 和 Terra) 的任務描述檔。
- 其他天線下行組態 - 其他公有廣播衛星 (Aqua、SNPP 和 Terra) 的天線下行組態。

在此範本中，衛星的值和參數已經產生。這些參數可讓您立即與這些衛星 AWS Ground Station 搭配使用。您不需要設定自己的值，即可在使用此範本 AWS Ground Station 時使用。但是，您可以自訂這些值，讓範本適用於您的使用案例。

我可以在什麼地方接收我的資料？

資料流程端點群組的設定，是使用以範本的一部分建立的接收器執行個體網路界面。接收者執行個體使用資料流程端點應用程式，從資料流程端點定義的 AWS Ground Station 連接埠接收資料串流。接收到以後，資料就可以透過 UDP 連接埠 50000 在接收器執行個體的回送介面卡上使用。如需設定資料流程端點群組的詳細資訊，請參閱 [AWS::GroundStation::DataflowEndpointGroup](#)。

使用 AWS Ground Station 代理程式的公有廣播衛星 (寬頻)

此範例建立在 使用者指南的 [JPSS-1 - 公有廣播衛星 \(PBS\) - 評估](#) 區段中完成的分析。

若要完成此範例，您需要假設一個案例 -- 您想要將 HRD 通訊路徑擷取為寬頻數位中繼頻率 (DigIF)，並在 Amazon EC2 執行個體上的 AWS Ground Station 代理程式使用 SDR 接收時加以處理。

Note

實際 JPSS HRD 通訊路徑訊號的頻寬為 30 MHz，但您將設定天線下行組態，將其視為頻寬為 100 MHz 的訊號，以便它可以流經 AWS Ground Station 代理程式在此範例中接收的正確路徑。

通訊路徑

本節代表[規劃您的資料流程通訊路徑](#)入門。在此範例中，您需要 CloudFormation 範本中尚未在其他範例映射區段中使用的額外區段。

Note

如需 CloudFormation 範本內容的詳細資訊，請參閱[範本章節](#)。

首先，在 CloudFormation 範本中為依區域列出的 AWS Ground Station 字首清單設定映射區段。這可讓 Amazon EC2 執行個體安全群組輕鬆參考字首清單。如需使用字首清單的詳細資訊，請參閱[使用 AWS Ground Station 代理程式的 VPC 組態](#)。

```
Mappings:
  PrefixListId:
    us-east-2:
      groundstation: pl-087f83ba4f34e3bea
    us-west-2:
      groundstation: pl-0cc36273da754ebdc
    us-east-1:
      groundstation: pl-0e5696d987d033653
    eu-central-1:
      groundstation: pl-03743f81267c0a85e
    sa-east-1:
      groundstation: pl-098248765e9effc20
    ap-northeast-2:
      groundstation: pl-059b3e0b02af70e4d
    ap-southeast-1:
```

```
groundstation: pl-0d9b804fe014a6a99
ap-southeast-2:
  groundstation: pl-08d24302b8c4d2b73
me-south-1:
  groundstation: pl-02781422c4c792145
eu-west-1:
  groundstation: pl-03fa6b266557b0d4f
eu-north-1:
  groundstation: pl-033e44023025215c0
af-south-1:
  groundstation: pl-0382d923a9d555425
```

針對參數區段，您將新增下列參數。透過 CloudFormation 主控台建立堆疊時，您將指定這些值。

Parameters:

EC2Key:

Description: The SSH key used to access the EC2 receiver instance. Choose any SSH key if you are not creating an EC2 receiver instance. For instructions on how to create an SSH key see <https://docs.aws.amazon.com/AWSEC2/latest/UserGuide/create-key-pairs.html>

Type: AWS::EC2::KeyPair::KeyName

ConstraintDescription: must be the name of an existing EC2 KeyPair.

AZ:


Description: "The AvailabilityZone that the resources of this stack will be created in. (e.g. us-east-2a)"

Type: AWS::EC2::AvailabilityZone::Name

ReceiverAMI:

Description: The Ground Station Agent AMI ID you want to use. Please note that AMIs are region specific. For instructions on how to retrieve an AMI see <https://docs.aws.amazon.com/ground-station/latest/ug/dataflows.ec2-configuration.html#dataflows.ec2-configuration.amis>

Type: AWS::EC2::Image::Id

 Note

您需要建立金鑰對，並提供 Amazon EC2 EC2Key 參數的名稱。請參閱[為您的 Amazon EC2 執行個體建立金鑰對](#)。

此外，建立 CloudFormation 堆疊時，您需要提供正確的區域特定 AMI ID。請參閱 [AWS Ground Station Amazon Machine Image AMIs](#)。

剩餘的範本程式碼片段屬於範本的資源區段 CloudFormation 。

Resources:

```
# Resources that you would like to create should be placed within the Resources section.
```

鑑於我們將單一通訊路徑交付至 Amazon EC2 執行個體的案例，您知道您將擁有單一同步交付路徑。根據 [同步資料交付](#) 區段，您必須使用 AWS Ground Station Agent 設定 Amazon EC2 執行個體，並建立一或多個資料流程端點群組。首先，您將先為 AWS Ground Station 代理程式設定 Amazon VPC。

```
ReceiverVPC:
```

```
  Type: AWS::EC2::VPC
```

```
  Properties:
```

```
    EnableDnsSupport: 'true'
```

```
    EnableDnsHostnames: 'true'
```

```
    CidrBlock: 10.0.0.0/16
```

```
  Tags:
```

```
    - Key: "Name"
```

```
      Value: "AWS Ground Station Example - PBS to AWS Ground Station Agent VPC"
```

```
    - Key: "Description"
```

```
      Value: "VPC for EC2 instance receiving AWS Ground Station data"
```

```
PublicSubnet:
```

```
  Type: AWS::EC2::Subnet
```

```
  Properties:
```

```
    VpcId: !Ref ReceiverVPC
```

```
    MapPublicIpOnLaunch: 'true'
```

```
    AvailabilityZone: !Ref AZ
```

```
    CidrBlock: 10.0.0.0/20
```

```
  Tags:
```

```
    - Key: "Name"
```

```
      Value: "AWS Ground Station Example - PBS to AWS Ground Station Agent Public Subnet"
```

```
    - Key: "Description"
```

```
      Value: "Subnet for EC2 instance receiving AWS Ground Station data"
```

```
RouteTable:
  Type: AWS::EC2::RouteTable
  Properties:
    VpcId: !Ref ReceiverVPC
    Tags:
      - Key: Name
        Value: AWS Ground Station Example - RouteTable

RouteTableAssociation:
  Type: AWS::EC2::SubnetRouteTableAssociation
  Properties:
    RouteTableId: !Ref RouteTable
    SubnetId: !Ref PublicSubnet

Route:
  Type: AWS::EC2::Route
  DependsOn: InternetGateway
  Properties:
    RouteTableId: !Ref RouteTable
    DestinationCidrBlock: '0.0.0.0/0'
    GatewayId: !Ref InternetGateway

InternetGateway:
  Type: AWS::EC2::InternetGateway
  Properties:
    Tags:
      - Key: Name
        Value: AWS Ground Station Example - Internet Gateway

GatewayAttachment:
  Type: AWS::EC2::VPCGatewayAttachment
  Properties:
    VpcId: !Ref ReceiverVPC
    InternetGatewayId: !Ref InternetGateway
```

Note

如需 AWS Ground Station 代理程式支援之 VPC 組態的詳細資訊，請參閱 [AWS Ground Station 代理程式需求 - VPC 圖表](#)。

接著，您將設定接收者 Amazon EC2 執行個體。

```
# The placement group in which your EC2 instance is placed.
ClusterPlacementGroup:
  Type: AWS::EC2::PlacementGroup
  Properties:
    Strategy: cluster

# This is required for the EIP if the receiver EC2 instance is in a private subnet.
# This ENI must exist in a public subnet, be attached to the receiver and be
associated with the EIP.
ReceiverInstanceNetworkInterface:
  Type: AWS::EC2::NetworkInterface
  Properties:
    Description: Floating network interface
    GroupSet:
      - !Ref InstanceSecurityGroup
    SubnetId: !Ref PublicSubnet

# An EIP providing a fixed IP address for AWS Ground Station to connect to. Attach it
to the receiver instance created in the stack.
ReceiverInstanceElasticIp:
  Type: AWS::EC2::EIP
  Properties:
    Tags:
      - Key: Name
        Value: !Join [ "-", [ "EIP" , !Ref "AWS::StackName" ] ]

# Attach the ENI to the EC2 instance if using a separate public subnet.
# Requires the receiver instance to be in a public subnet (SubnetId should be the id
of a public subnet)
ReceiverNetworkInterfaceAttachment:
  Type: AWS::EC2::NetworkInterfaceAttachment
  Properties:
    DeleteOnTermination: false
    DeviceIndex: 1
    InstanceId: !Ref ReceiverInstance
    NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

# Associate EIP with the ENI if using a separate public subnet for the ENI.
ReceiverNetworkInterfaceElasticIpAssociation:
  Type: AWS::EC2::EIPAssociation
  Properties:
    AllocationId: !GetAtt [ReceiverInstanceElasticIp, AllocationId]
```

```

    NetworkInterfaceId: !Ref ReceiverInstanceNetworkInterface

# The EC2 instance that will send/receive data to/from your satellite using AWS
Ground Station.
ReceiverInstance:
  Type: AWS::EC2::Instance
  DependsOn: PublicSubnet
  Properties:
    DisableApiTermination: false
    IamInstanceProfile: !Ref GeneralInstanceProfile
    ImageId: !Ref ReceiverAMI
    AvailabilityZone: !Ref AZ
    InstanceType: c5.24xlarge
    KeyName: !Ref EC2Key
    Monitoring: true
    PlacementGroupName: !Ref ClusterPlacementGroup
    SecurityGroupIds:
      - Ref: InstanceSecurityGroup
    SubnetId: !Ref PublicSubnet
    Tags:
      - Key: Name
        Value: !Join [ "-", [ "Receiver" , !Ref "AWS::StackName" ] ]
# agentCpuCores list in the AGENT_CONFIG below defines the cores that the AWS
Ground Station Agent is allowed to run on. This list can be changed to suit your use-
case, however if the agent isn't supplied with enough cores data loss may occur.
UserData:
  Fn::Base64:
    Fn::Sub:
      - |
        #!/bin/bash
        yum -y update

        AGENT_CONFIG_PATH="/opt/aws/groundstation/etc/aws-gs-agent-config.json"
        cat << AGENT_CONFIG > "$AGENT_CONFIG_PATH"
        {
          "capabilities": [
            "arn:aws:groundstation:${AWS::Region}:${AWS::AccountId}:dataflow-
endpoint-group/${DataflowEndpointGroupId}"
          ],
          "device": {
            "privateIps": [
              "127.0.0.1"
            ],
            "publicIps": [

```

```

        "${EIP}"
    ],
    "agentCpuCores": [
24,25,26,27,28,29,30,31,32,33,34,35,36,37,38,39,40,41,42,43,44,72,73,74,75,76,77,78,79,80,81,8
    ]
}
AGENT_CONFIG

systemctl start aws-groundstation-agent
systemctl enable aws-groundstation-agent

# <Tuning Section Start>
# Visit the AWS Ground Station Agent Documentation in the User Guide for
more details and guidance updates

# Set IRQ affinity with list of CPU cores and Receive Side Scaling mask
# Core list should be the first two cores (and hyperthreads) on each
socket

# Mask set to everything currently
# https://github.com/torvalds/linux/blob/v4.11/Documentation/networking/
scaling.txt#L80-L96
echo "@reboot sudo /opt/aws/groundstation/bin/set_irq_affinity.sh '0 1 48
49' 'ffffffff,ffffffff,ffffffff' >>/var/log/user-data.log 2>&1" >>/var/spool/cron/root

# Reserving the port range defined in the GS agent ingress address in
the Dataflow Endpoint Group so the kernel doesn't steal any of them from the GS agent.
These ports are the ports that the GS agent will ingress data
# across, so if the kernel steals one it could cause problems ingressing
data onto the instance.
echo net.ipv4.ip_local_reserved_ports="42000-50000" >> /etc/sysctl.conf

# </Tuning Section End>

# We have to reboot for linux kernel settings to apply
shutdown -r now

- DataflowEndpointGroupId: !Ref DataflowEndpointGroup
  EIP: !Ref ReceiverInstanceElasticIp

```

```

# The AWS Ground Station Dataflow Endpoint Group that defines the endpoints that AWS
Ground
# Station will use to send/receive data to/from your satellite.
DataflowEndpointGroup:
  Type: AWS::GroundStation::DataflowEndpointGroup
  Properties:
    ContactPostPassDurationSeconds: 180
    ContactPrePassDurationSeconds: 120
    EndpointDetails:
      - AwsGroundStationAgentEndpoint:
          Name: !Join [ "-", [ !Ref "AWS::StackName" , "Downlink" ] ] # needs to
match DataflowEndpointConfig name
          EgressAddress:
            SocketAddress:
              Name: 127.0.0.1
              Port: 55000
            IngressAddress:
              SocketAddress:
                Name: !Ref ReceiverInstanceElasticIp
              PortRange:
                Minimum: 42000
                Maximum: 55000

```

您也需要適當的政策、角色和設定檔，AWS Ground Station 才能允許在帳戶中建立彈性網路界面 (ENI)。

```

# The security group for your EC2 instance.
InstanceSecurityGroup:
  Type: AWS::EC2::SecurityGroup
  Properties:
    GroupDescription: AWS Ground Station receiver instance security group.
    VpcId: !Ref ReceiverVPC
    SecurityGroupEgress:
      - CidrIp: 0.0.0.0/0
        Description: Allow all outbound traffic by default
        IpProtocol: "-1"
    SecurityGroupIngress:
      # To allow SSH access to the instance, add another rule allowing tcp port 22
from your CidrIp
      - IpProtocol: udp
        Description: Allow AWS Ground Station Incoming Dataflows

```

```
ToPort: 50000
FromPort: 42000
SourcePrefixListId:
  Fn::FindInMap:
    - PrefixListId
    - Ref: AWS::Region
    - groundstation

# The EC2 instance assumes this role.
InstanceRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Version: "2012-10-17"
      Statement:
        - Effect: "Allow"
          Principal:
            Service:
              - "ec2.amazonaws.com"
          Action:
            - "sts:AssumeRole"
    Path: "/"
    ManagedPolicyArns:
      - arn:aws:iam::aws:policy/AmazonS3ReadOnlyAccess
      - arn:aws:iam::aws:policy/service-role/AmazonEC2ContainerServiceforEC2Role
      - arn:aws:iam::aws:policy/CloudWatchAgentServerPolicy
      - arn:aws:iam::aws:policy/service-role/AmazonEC2RoleforSSM
      - arn:aws:iam::aws:policy/AWSGroundStationAgentInstancePolicy
    Policies:
      - PolicyDocument:
          Statement:
            - Action:
                - sts:AssumeRole
              Effect: Allow
              Resource: !GetAtt GroundStationKmsKeyRole.Arn
            Version: "2012-10-17"
          PolicyName: InstanceGroundStationApiAccessPolicy

# The instance profile for your EC2 instance.
GeneralInstanceProfile:
  Type: AWS::IAM::InstanceProfile
  Properties:
    Roles:
      - !Ref InstanceRole
```

```
# The IAM role that AWS Ground Station will assume to access and use the KMS Key for
data delivery
GroundStationKmsKeyRole:
  Type: AWS::IAM::Role
  Properties:
    AssumeRolePolicyDocument:
      Statement:
        - Action: sts:AssumeRole
          Effect: Allow
          Principal:
            Service:
              - groundstation.amazonaws.com
          Condition:
            StringEquals:
              "aws:SourceAccount": !Ref AWS::AccountId
            ArnLike:
              "aws:SourceArn": !Sub "arn:${AWS::Partition}:groundstation:
${AWS::Region}:${AWS::AccountId}:mission-profile/*"
        - Action: sts:AssumeRole
          Effect: Allow
          Principal:
            AWS: !Sub "arn:${AWS::Partition}:iam:${AWS::AccountId}:root"

GroundStationKmsKeyAccessPolicy:
  Type: AWS::IAM::Policy
  Properties:
    PolicyDocument:
      Statement:
        - Action:
            - kms:Decrypt
          Effect: Allow
          Resource: !GetAtt GroundStationDataDeliveryKmsKey.Arn
    PolicyName: GroundStationKmsKeyAccessPolicy
  Roles:
    - Ref: GroundStationKmsKeyRole

GroundStationDataDeliveryKmsKey:
  Type: AWS::KMS::Key
  Properties:
    KeyPolicy:
      Statement:
        - Action:
            - kms:CreateAlias
```

```

    - kms:Describe*
    - kms:Enable*
    - kms:List*
    - kms:Put*
    - kms:Update*
    - kms:Revoke*
    - kms:Disable*
    - kms:Get*
    - kms>Delete*
    - kms:ScheduleKeyDeletion
    - kms:CancelKeyDeletion
    - kms:GenerateDataKey
    - kms:TagResource
    - kms:UntagResource
  Effect: Allow
  Principal:
    AWS: !Sub "arn:${AWS::Partition}:iam:${AWS::AccountId}:root"
  Resource: "*"
- Action:
  - kms:Decrypt
  - kms:GenerateDataKeyWithoutPlaintext
  Effect: Allow
  Principal:
    AWS: !GetAtt GroundStationKmsKeyRole.Arn
  Resource: "*"
  Condition:
    StringEquals:
      "kms:EncryptionContext:sourceAccount": !Ref AWS::AccountId
    ArnLike:
      "kms:EncryptionContext:sourceArn": !Sub "arn:
${AWS::Partition}:groundstation:${AWS::Region}:${AWS::AccountId}:mission-profile/*"
- Action:
  - kms>CreateGrant
  Effect: Allow
  Principal:
    AWS: !Sub "arn:${AWS::Partition}:iam:${AWS::AccountId}:root"
  Resource: "*"
  Condition:
    ForAllValues:StringEquals:
      "kms:GrantOperations":
        - Decrypt
        - GenerateDataKeyWithoutPlaintext
      "kms:EncryptionContextKeys":
        - sourceArn

```

```

    - sourceAccount
      ArnLike:
        "kms:EncryptionContext:sourceArn": !Sub "arn:
        ${AWS::Partition}:groundstation:${AWS::Region}:${AWS::AccountId}:mission-profile/*"
      StringEquals:
        "kms:EncryptionContext:sourceAccount": !Ref AWS::AccountId
      Version: "2012-10-17"
      EnableKeyRotation: true

```

AWS Ground Station 組態

本節代表[建立組態](#)入門。

您需要使用追蹤組態，才能在上使用自動追蹤設定偏好設定。選取 PREFERRED 做為自動追蹤可以改善訊號品質，但由於 JPSS-1 ephemeris 品質足夠，因此不需要符合訊號品質。

```

TrackingConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "JPSS Tracking Config"
    ConfigData:
      TrackingConfig:
        Autotrack: "PREFERRED"

```

根據通訊路徑，您需要定義天線下行組態來代表衛星部分，以及定義端點詳細資訊的資料流程端點群組的資料流程端點組態。

```

# The AWS Ground Station Antenna Downlink Config that defines the frequency spectrum
used to
# downlink data from your satellite.
SnppJpssDownlinkDigIfAntennaConfig:
  Type: AWS::GroundStation::Config
  Properties:
    Name: "SNPP JPSS Downlink WBDigIF Antenna Config"
    ConfigData:
      AntennaDownlinkConfig:
        SpectrumConfig:
          Bandwidth:

```

```

        Units: "MHz"
        Value: 100
    CenterFrequency:
        Units: "MHz"
        Value: 7812
    Polarization: "RIGHT_HAND"

# The AWS Ground Station Dataflow Endpoint Config that defines the endpoint used to
downlink data
# from your satellite.
DownlinkDigIfEndpointConfig:
    Type: AWS::GroundStation::Config
    Properties:
        Name: "Aqua SNPP JPSS Terra Downlink DigIF Endpoint Config"
        ConfigData:
            DataflowEndpointConfig:
                DataflowEndpointName: !Join [ "-", [ !Ref "AWS::StackName" , "Downlink" ] ]
                DataflowEndpointRegion: !Ref AWS::Region

```

AWS Ground Station 任務描述檔

本節代表[建立任務描述檔](#)入門。

現在您已擁有相關聯的組態，您可以使用它們來建構資料流程。您將使用其餘參數的預設值。

```

# The AWS Ground Station Mission Profile that groups the above configurations to
define how to
# uplink and downlink data to your satellite.
SnpjMissionProfile:
    Type: AWS::GroundStation::MissionProfile
    Properties:
        Name: !Sub 'JPSS WBDigIF gs-agent EC2 Delivery'
        ContactPrePassDurationSeconds: 120
        ContactPostPassDurationSeconds: 120
        MinimumViableContactDurationSeconds: 180
        TrackingConfigArn: !Ref TrackingConfig
        DataflowEdges:
            - Source: !Ref SnpjDownlinkDigIfAntennaConfig
              Destination: !Ref DownlinkDigIfEndpointConfig
        StreamsKmsKey:
            KmsKeyArn: !GetAtt GroundStationDataDeliveryKmsKey.Arn

```

```
StreamsKmsRole: !GetAtt GroundStationKmsKeyRole.Arn
```

將它放在一起

透過上述資源，您現在可以排程 JPSS-1 聯絡人，以便從任何已加入的 同步資料交付 AWS Ground Station [AWS Ground Station 位置](#)。

以下是完整的 CloudFormation 範本，其中包含本節所述的所有資源，合併為可以直接使用的單一範本 CloudFormation。

名為 `DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml` 的 CloudFormation 範本旨在讓您快速存取，以開始接收 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 衛星的數位化中繼頻率 (DigIF) 資料。它包含 Amazon EC2 執行個體，以及使用 AWS Ground Station Agent 接收原始 DigIF 直接廣播資料所需的 CloudFormation 資源。

如果 Aqua、SNPP、JPSS-1/NOAA-20 和 Terra 未加入您的帳戶，請參閱 [加入衛星](#)。

Note

您可以使用有效的 AWS 登入資料來存取客戶加入 Amazon S3 儲存貯體，以存取範本。以下連結使用區域 Amazon S3 儲存貯體。變更 `us-west-2` 區域碼，代表您要建立 CloudFormation 堆疊的對應區域。

此外，下列指示使用 YAML。但是，範本同時提供 YAML 和 JSON 格式。若要使用 JSON，請在下載範本 `.json` 時將 `.yml` 副檔名取代為。

若要使用 下載範本 AWS CLI，請使用下列命令：

```
aws s3 cp s3://groundstation-cloudformation-templates-us-west-2/agent/ec2_delivery/DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml .
```

您可以在瀏覽器中瀏覽至以下 URL，以在主控台中檢視和下載範本：

```
https://s3.console.aws.amazon.com/s3/object/groundstation-cloudformation-templates-us-west-2/agent/ec2_delivery/DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml
```

您可以使用 CloudFormation 以下連結直接在 中指定範本：

```
https://groundstation-cloudformation-templates-us-west-2.s3.us-west-2.amazonaws.com/agent/ec2_delivery/DirectBroadcastSatelliteWbDigIfEc2DataDelivery.yml
```

範本定義了哪些其他資源？

DirectBroadcastSatelliteWbDigIfEc2DataDelivery 範本包含下列其他資源：

- 接收者執行個體彈性網路界面 - (條件式) 如果提供彈性網路界面，則會在 PublicSubnetId 指定的子網路中建立彈性網路界面。如果接收者執行個體位於私有子網路中，則此為必要項目。彈性網路界面將與 EIP 建立關聯，並連接到接收者執行個體。
- 接收者執行個體彈性 IP - AWS Ground Station 將連線的彈性 IP。這會連接至接收器執行個體或彈性網路界面。
- 下列其中一個彈性 IP 關聯：
 - 接收者執行個體與彈性 IP 關聯 - 如果未指定 PublicSubnetId，則彈性 IP 與接收者執行個體的關聯。這需要 SubnetId 參考公有子網路。
 - 接收者執行個體彈性網路界面與彈性 IP 關聯 - 如果指定 PublicSubnetId，則彈性 IP 與接收者執行個體彈性網路界面的關聯。
- (選用) CloudWatch Event Triggers - AWS Lambda 使用聯絡 AWS Ground Station 前後由傳送的 CloudWatch Events 觸發的函數。AWS Lambda 函數將啟動並選擇性地停止您的接收者執行個體。
- (選用) Amazon EC2 聯絡驗證 - 使用 Lambda 為具有 SNS 通知的聯絡設定 Amazon EC2 執行個體驗證系統的選項 (SNS 通知)。請務必注意，這可能會根據您目前的使用量產生費用。
- 其他任務描述檔 - 其他公有廣播衛星 (Aqua、SNPP 和 Terra) 的任務描述檔。
- 其他天線下行組態 - 其他公有廣播衛星 (Aqua、SNPP 和 Terra) 的天線下行組態。

在此範本中，衛星的值和參數已經產生。這些參數可讓您輕鬆地 AWS Ground Station 立即與這些衛星搭配使用。您不需要設定自己的值，即可在使用此範本 AWS Ground Station 時使用。但是，您可以自訂這些值，讓範本適用於您的使用案例。

我可以在什麼地方接收我的資料？

資料流程端點群組的設定，是使用以範本的一部分建立的接收器執行個體網路界面。接收者執行個體使用 AWS Ground Station 代理程式，在資料流程端點定義的連接埠 AWS Ground Station 上接收來自的資料串流。如需設定資料流程端點群組的詳細資訊，請參閱 [AWS::GroundStation::DataflowEndpointGroup](#)。如需代理程式的詳細資訊 AWS Ground Station，請參閱 [什麼是 AWS Ground Station 代理程式？](#)

疑難排解

下列文件可協助您疑難排解使用 時可能發生的問題 AWS Ground Station。

主題

- [對將資料交付至 Amazon EC2 的聯絡人進行故障診斷](#)
- [故障診斷失敗的聯絡案例](#)
- [故障診斷失敗的聯絡更新](#)
- [故障診斷 FAILED_TO_SCHEDULE 聯絡人](#)
- [針對未處於 HEALTHY 狀態的 DataflowEndpointGroups 進行故障診斷](#)
- [故障診斷無效的短暫性事件](#)
- [對未收到資料的聯絡人進行故障診斷](#)
- [對遙測進行故障診斷](#)

對將資料交付至 Amazon EC2 的聯絡人進行故障診斷

如果您無法成功完成 AWS Ground Station 聯絡，則需要確認 Amazon EC2 執行個體是否正在執行、確認資料流程端點應用程式正在執行，並確認資料流程端點應用程式的串流已正確設定。

Note

DataDefender (DDX) 是目前支援的資料流程端點應用程式範例 AWS Ground Station

必要條件

下列程序假設 Amazon EC2 執行個體已設定。若要在 中設定 Amazon EC2 執行個體 AWS Ground Station，請參閱[入門](#)。

步驟 1：確認您的 EC2 執行個體正在執行

下列程序說明如何在主控台中尋找您的 Amazon EC2 執行個體，並在未執行時啟動執行個體。

1. 找到用於您要疑難排解之聯絡人的 Amazon EC2 執行個體。使用下列步驟：
 - a. 在CloudFormation儀表板中，選取包含 Amazon EC2 執行個體的堆疊。

- b. 選擇資源索引標籤，並在邏輯 ID 欄中尋找您的 Amazon EC2 執行個體。驗證執行個體是在 Status (狀態) 欄中建立。
 - c. 在實體 ID 欄中，選擇 Amazon EC2 執行個體的連結。這將帶您前往 Amazon EC2 管理主控台。
2. 在 Amazon EC2 管理主控台中，確保您的 Amazon EC2 執行個體狀態正在執行。
3. 如果您的執行個體正在執行，請繼續下一個步驟。如果您的執行個體未執行，請依照以下步驟啟動執行個體：
 - 選取 Amazon EC2 執行個體後，選擇動作 > 執行個體狀態 > 開始。

步驟 2：判斷使用的資料流程應用程式類型

如果您使用 AWS Ground Station 代理程式進行資料交付，請重新導向至 [AWS Ground Station 客服人員疑難排解](#) 一節。否則，如果您使用的是 DataDefender (DDX) 應用程式，請繼續 [the section called “步驟 3：確認資料流程應用程式正在執行”](#)。

步驟 3：確認資料流程應用程式正在執行

驗證 DataDefender 狀態需要您連線到 Amazon EC2 中的執行個體。如需連接至執行個體的詳細資訊，請參閱 [連接至 Linux 執行個體](#)。

下列程序提供在 SSH 用戶端中使用命令的故障診斷步驟。

1. 開啟終端機或命令提示，並使用 SSH 連線至 Amazon EC2 執行個體。轉送遠端主機的連接埠 80，以檢視 DataDefender Web UI。下列命令示範如何使用 SSH 透過啟用連接埠轉送的堡壘連線至 Amazon EC2 執行個體。

Note

您必須以特定 SSH 金鑰、堡壘主機名稱和 Amazon EC2 執行個體主機名稱取代 <SSH KEY>、<BASTION HOST> 和 <HOST>。

適用於 Windows

```
ssh -L 8080:localhost:80 -o ProxyCommand="C:\Windows\System32\OpenSSH\ssh.exe -o \"ForwardAgent yes\" -W %h:%p -i \"<SSH KEY>\" ec2-user@<BASTION HOST>" -i "<SSH KEY>" ec2-user@<HOST>
```

適用於 Mac

```
ssh -L 8080:localhost:80 -o ProxyCommand="ssh -A -o 'ForwardAgent yes' -W %h:%p -i <SSH KEY> ec2-user@<BASTION HOST>" -i <SSH KEY> ec2-user@<HOST>
```

2. 驗證 DataDefender (也稱為 DDX) 正在執行，方法是針對輸出中名為 ddx 的執行程序進行 grepping (檢查)。以下提供一個執行中處理程序和一個成功範例輸出的 Grepping 檢查命令。

```
[ec2-user@Receiver-Instance ~]$ ps -ef | grep ddx
      Rtlogic  4977      1 10 Oct16 ?          2-00:22:14 /opt/rtlogic/ddx/
bin/ddx -m/opt/rtlogic/ddx/modules -p/opt/rtlogic/ddx/plugins -c/opt/rtlogic/
ddx/bin/ddx.xml -umask=077 -daemon -f installed=true -f security=true -f enable
  HttpsForwarding=true
      Ec2-user 18787 18657  0 16:51 pts/0      00:00:00 grep -color=auto ddx
```

如果 DataDefender 正在執行，請跳至 [the section called “步驟 4：確認您的資料流程應用程式串流已設定”](#) 否則，請繼續下一個步驟。

3. 使用以下命令啟動 DataDefender。

```
sudo service rtlogic-ddx start
```

如果 DataDefender 在使用命令後正在執行，請跳至 [the section called “步驟 4：確認您的資料流程應用程式串流已設定”](#) 否則，請繼續下一個步驟。

4. 使用以下命令檢查下列檔案，以查看安裝和設定 DataDefender 時是否有任何錯誤。

```
cat /var/log/user-data.log
      cat /opt/aws/groundstation/.startup.out
```

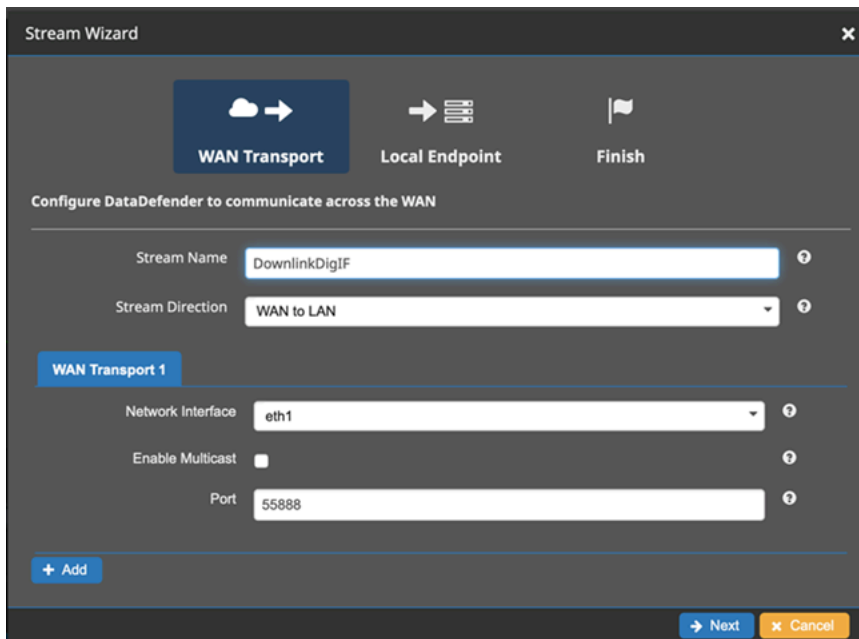
Note

檢查這些檔案時發現的常見問題是 Amazon EC2 執行個體在其中執行的 Amazon VPC 無法存取 Amazon S3 下載安裝檔案。如果您在日誌中發現此問題，請檢查 EC2 執行個體的 Amazon VPC 和安全群組設定，以確保它們不會封鎖對 Amazon S3 的存取。

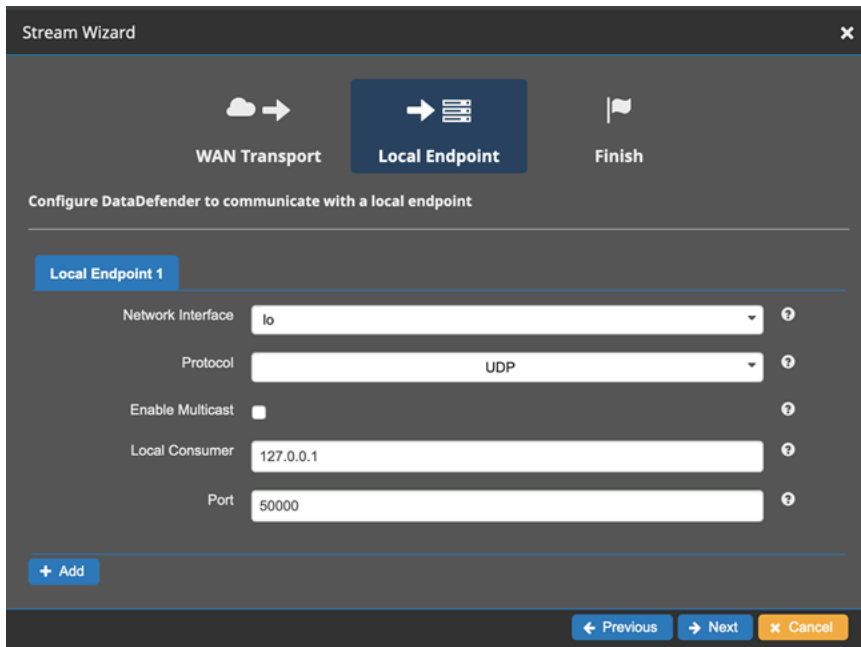
如果 DataDefender 在檢查您的 Amazon VPC 設定後正在執行，請繼續 [the section called “步驟 4：確認您的資料流程應用程式串流已設定”](#)。如果問題持續發生，請[聯絡 AWS Support](#) 並傳送日誌檔案與問題描述。

步驟 4：確認您的資料流程應用程式串流已設定

1. 在 Web 瀏覽器中，透過在地址列中輸入下列地址來存取 DataDefender Web 使用者介面：localhost : 8080。然後按 Enter 鍵。
2. 在 DataDefender 儀表中，選擇 Go to Details (移至詳細資訊)。
3. 從串流清單中選取串流，然後選擇 Edit Stream (編輯串流)。
4. 在 Stream Wizard (串流精靈) 對話方塊中，執行下列動作：
 - a. 在 WAN 傳輸窗格中，確定已選取串流方向的 WAN 至 LAN。
 - b. 在 Port (連接埠) 方塊中，確保您已為資料流程端點群組選擇的 WAN 連接埠存在。此連接埠預設為 55888。然後選擇下一步。



- c. 在 Local Endpoint (本機端點) 窗格中，確保有效的連接埠存在於 Port (連接埠) 方塊。此連接埠預設為 50000。這是 DataDefender 從 AWS Ground Station 服務接收資料後，您會在其中接收資料的連接埠。然後選擇下一步。



- d. 如果您已變更任何值，請透過其餘功能表選擇 Finish (完成)。否則您可以從 Stream Wizard (串流精靈) 功能表取消。

您現在已確保 Amazon EC2 執行個體和 DataDefender 都已正確執行並設定以接收資料 AWS Ground Station。繼續進行 [the section called “步驟 5：確定您的接收器執行個體（多個）子網路中有足夠的可用 IP 地址”](#)。

步驟 5：確定您的接收器執行個體（多個）子網路中有足夠的可用 IP 地址

下列程序說明如何在主控台的 Amazon EC2 接收器執行個體中尋找可用的 IP 地址數量。

1. 針對用於您要疑難排解之聯絡案例的每個 Amazon EC2 接收器執行個體。使用下列步驟：
 - a. 在 CloudFormation 儀表板中，選取包含 Amazon EC2 執行個體的堆疊。
 - b. 選擇資源索引標籤，並在邏輯 ID 欄中尋找您的 Amazon EC2 執行個體。驗證執行個體是在 Status (狀態) 欄中建立。
 - c. 在實體 ID 欄中，選擇 Amazon EC2 執行個體的連結。這將帶您前往 Amazon EC2 管理主控台。
2. 在 Amazon EC2 管理主控台中，尋找並按一下 Amazon EC2 接收器執行個體的執行個體摘要中的子網路 ID 連結。這將帶您前往對應的 Amazon VPC 管理主控台。

3. 在 Amazon VPC 管理主控台中選取相符的子網路，並檢查子網路的詳細資訊以取得可用的 IPv4 地址。如果此數字不等於使用此 Amazon EC2 接收器執行個體的資料流程端點，請執行下列動作：
 - a. 更新 CloudFormation 範本對應的子網路 CidrBlock 以正確調整大小。如需子網路大小的詳細資訊，請參閱[子網路 CIDR 區塊](#)。
 - b. 使用更新後的 CloudFormation 範本重新部署堆疊。

如果您持續遇到問題，[請聯絡 AWS Support](#)。

故障診斷失敗的聯絡案例

當 AWS Ground Station 偵測到資源組態問題時，聯絡人的終端聯絡人狀態將為 FAILED。以下提供可能導致 FAILED 聯絡人的常見使用案例，以及協助故障診斷的步驟。

Note

本指南專門用於 FAILED 聯絡狀態 - 不適用於其他失敗狀態，例如 AWS_FAILED、AWS_CANCELLED 或 FAILED_TO_SCHEDULE。如需聯絡狀態的詳細資訊，請參閱 [the section called “AWS Ground Station 聯絡狀態”](#)

Dataflow 端點 FAILED 使用案例

以下是可能導致資料流程端點型資料流程的 FAILED 聯絡狀態的常見使用案例清單：

- 資料流程端點永遠不會連線 - AWS Ground Station 從未建立一或多個資料流程的天線與資料流程端點群組之間的連線。
- Dataflow 端點會延遲連線 - 在聯絡開始時間之後建立一或多個資料流程的 AWS Ground Station 天線與 Dataflow 端點群組之間的連線。
- Dataflow 端點的子網路沒有可用的 IP 地址 - 由於接收器執行個體子網路中沒有任何可用的 IP 地址，因此 AWS Ground Station 的資料交付解決方案無法在您的私有網路中建立 ENI。
- Dataflow 端點的子網路無效 - 由於無法存取 Dataflow 端點群組中指定的子網路，AWS Ground Station 的資料交付解決方案無法在私有網路中建立 ENI。

對於任何資料流程端點失敗案例，建議查看下列項目：

- 在聯絡開始時間之前，確認接收者 Amazon EC2 執行個體已成功啟動。
- 確認資料流程端點軟體已在聯絡期間啟動並執行。
- 確保每個接收器執行個體子網路的每個資料流程端點至少有一個可用的 IP 地址。
- 透過在 [中](#)設定的資料流程，確保與 Dataflow 端點群組相關聯的子網路[設定 Amazon VPC](#)保持作用中且可供使用 AWS Ground Station。

如需更具體[對將資料交付至 Amazon EC2 的聯絡人進行故障診斷](#)的故障診斷步驟，請參閱上的一節。

AWS Ground Station 代理程式失敗的使用案例

以下是可能導致客服人員型資料流程聯絡狀態失敗的常見使用案例清單：

- AWS Ground Station 客服人員從未報告狀態 - 負責協調資料流程端點群組上一或多個資料流程之資料交付的客服人員從未成功報告狀態 AWS Ground Station。此狀態更新應在聯絡結束時間的幾秒鐘內進行。
- AWS Ground Station 客服人員延遲開始 - 負責協調 Dataflow 端點群組上一或多個資料流程資料交付的客服人員，在聯絡開始時間後延遲開始。

對於任何 AWS Ground Station 客服人員資料流程失敗案例，建議查看下列項目：

- 在聯絡開始時間之前，確認接收者 Amazon EC2 執行個體已成功啟動。
- 確認客服人員應用程式已在聯絡開始時和期間啟動並執行。
- 確認客服人員應用程式和 Amazon EC2 執行個體未在聯絡結束後 15 秒內關閉。這可讓客服人員有足夠的時間向其報告狀態 AWS Ground Station。

如需更具體[對將資料交付至 Amazon EC2 的聯絡人進行故障診斷](#)的故障診斷步驟，請參閱上的一節。

故障診斷失敗的聯絡更新

當您呼叫 [UpdateContact](#) API 時，會對請求 AWS Ground Station 執行同步驗證。如果驗證通過，則會以非同步方式處理更新，將變更傳播到天線區域。同步驗證錯誤會直接在 HTTP 回應中傳回。非同步失敗會透過聯絡版本上的 `failureCodes`和 `failureMessage` 欄位回報，您可以透過呼叫 [DescribeContactVersion](#) 來檢視更新失敗的版本。

如需聯絡版本控制的詳細資訊，請參閱 [更新聯絡人和聯絡人版本控制](#)。

同步驗證錯誤

當 [UpdateContact](#) 請求驗證失敗時，HTTP 回應中會直接傳回下列錯誤。

ResourceNotFoundException：找不到聯絡人

常見原因

指定的 `contactId` 不存在或屬於不同的 AWS 帳戶。

解決方案

1. 驗證 `contactId` 是否正確。
2. 確認您正在使用擁有聯絡人之 AWS 帳戶的登入資料。
3. 使用 [ListContacts](#) 尋找正確的 `contactId`。

ConflictException：無法更新聯絡人

常見原因

聯絡人處於不允許更新的狀態。只有在聯絡案例處於 SCHEDULED、或 PASS 狀態時PREPASS，才能呼叫 [UpdateContact](#) API。如果另一個更新已在進行中（最新的聯絡版本處於 UPDATING 狀態），也會發生此錯誤。

解決方案

1. 呼叫 [DescribeContact](#) 以檢查目前的聯絡狀態。
2. 如果聯絡人處於終端狀態（例如 COMPLETED、或 CANCELLED)FAILED，則無法更新。只有在聯絡人處於 SCHEDULED、或 PASS 狀態時PREPASS，才能更新聯絡人。如需終端狀態的完整清單，請參閱 [AWS Ground Station 聯絡狀態](#)。
3. 如果另一個更新正在進行中，請等待目前的更新達到 ACTIVE或 FAILED_TO_UPDATE 狀態，然後再提交另一個更新。您可以輪詢 [DescribeContactVersion](#) API，或使用 AWS SDKs和 提供的ContactUpdated等待程式便利公用程式 AWS Command Line Interface。

InvalidParameterException：無效的請求參數

常見原因

請求包含無效的參數。常見原因包括：

- 遺失或空白 `clientToken`。
- 單一請求中包含多種類型的 `ProgramTrackSettings` (方位/上升、OEM 和 TLE)。每個請求只允許一種類型。
- 將 `satelliteArn` 設定為 `null`，而不核准在聯絡人的地面工作站進行 [方位角抬高暫時性](#)。
- 當 `satelliteArn` 為 `null` `AzElProgramTrackSettings` 時遺失。
- 提供 `ephemerisId` 未與指定相關聯的 `satelliteArn`。
- 對於聯絡時間範圍，衛星沒有來自地面工作站的有效可見性時段。
- 衛星未加入地面工作站，或沒有任務設定檔所需的授權。
- 任務描述檔包含 [天線下行解調解碼組態組態](#)，不支援聯絡人更新。

解決方案

1. 檢閱回應中的錯誤訊息，了解哪個參數無效的詳細資訊。
2. 請確定每個 `ProgramTrackSettings` 請求只提供一種類型的。
3. 如果在沒有 `方位角/上升指向角` `satelliteArn` 的情況下使用 `方位角/上升指向角` `satelliteArn`，請確認您的帳戶已在地面站點獲得此功能的核准。如需詳細資訊，請參閱 [提供方位提升暫時性資料](#)。
4. 確認您參考的暫時性關聯至正確的衛星，並涵蓋聯絡時間範圍。

ResourceLimitExceededException：已達到最大版本限制

常見原因

聯絡人已達到版本數量上限 (128)。每次呼叫 [UpdateContact](#) 都會建立新的版本，且聯絡人不能超過此限制。

解決方案

1. 此限制無法提高。如果您需要進一步變更，請取消聯絡人並保留新的聯絡人。

非同步失敗代碼

下列失敗代碼會出現在 `FAILED_TO_UPDATE` 狀態為的聯絡版本的 `failureCodes` 欄位中。使用 [DescribeContactVersion](#) 擷取這些詳細資訊。`failureMessage` 欄位提供有關失敗的其他內容。

失敗代碼	常見原因	Resolution
INTERNAL_ERROR	處理更新時發生非預期的內部錯誤。	重試更新。如果問題仍持續發生，請聯絡 AWS 支援 。
INVALID_SATELLITE_ARN	更新請求中提供的衛星 ARN 無效或不存在。	驗證衛星 ARN 並確認衛星已在您的帳戶中註冊。
INVALID_UPDATE_CONTACT_REQUEST	更新請求包含同步驗證期間未攔截到的無效參數。	檢閱 failureMessage 以取得詳細資訊，並更正請求參數。
EPHEMERIS_NOT_FOUND	追蹤覆寫中參考的 ephemeris 不存在。	驗證 ephemerisId 並確認尚未刪除 ephemeris。
EPHEMERIS_TIME_RANGE_INVALID	ephemeris 不涵蓋聯絡的時間範圍。	上傳涵蓋完整聯絡時間範圍的新臨時事件。如果暫時時間範圍無法延長，請取消聯絡人，並在暫時時段保留新的聯絡人。如需詳細資訊，請參閱 提供自訂暫時性資料 。
EPHEMERIS_NOT_ENABLED	參考的 ephemeris 未處於 ENABLED 狀態。	檢查暫時性更新狀態，並在重試更新之前啟用。
SATELLITE_DOES_NOT_MATCH_EPHEMERIS	ephemeris 與更新請求中指定的衛星沒有關聯。	確保 ephemerisId 屬於中指定的衛星 satelliteArn 。
NOT_ONBOARDED_TO_AZEL_EPHEMERIS	您的帳戶未獲核准在聯絡人的地面工作站使用 方位提升暫時性資料 。方位浮水印是一種限制功能，適用於有限數量的專業使用案例。	如果您的使用案例需要方位提升暫時性，請透過 開立 AWS 支援 票證 AWS Support Center Console 以請求存取。或者，如果符合您的使用案例，請考慮使用 TLE ephemeris 資料 或 OEM ephemeris 資料 。
AZEL_EPHEMERIS_NOT_FOUND	請求中參考的 方位提升暫時性 不存在。	確認 ephemerisId 並確認尚未刪除方位角海拔凹凸。

失敗代碼	常見原因	Resolution
AZEL_EPHE MERIS_WRONG_GROUND_STATION	方位海拔 混淆是為與聯絡人所使用的不同地面站點所建立。	為正確的地面站點上傳新的方位角海拔暫時性，或使用符合聯絡人地面站點的現有暫時性。
AZEL_EPHE MERIS_INVALID_STATUS	方位角海拔高度不 處於有效狀態以供使用。	檢查暫時性狀態。它必須處於ENABLED 狀態。如果暫時性驗證失敗，請上傳更正的版本。
AZEL_EPHE MERIS_TIME_RANGE_INVALID	方位提升暫時性 不涵蓋聯絡的時間範圍。	上傳涵蓋完整聯絡時間範圍的新方位浮水印。如果暫時時間範圍無法延長，請取消聯絡人，並在暫時時段保留新的聯絡人。

檢查更新的狀態

呼叫 `UpdateContact` 後，新的聯絡版本會開始處於 `UPDATING` 狀態。在此期間，[DescribeContact](#) 會繼續傳回先前作用中版本的聯絡人。在新版本傳播至天線並達到 `ACTIVE` 狀態 `DescribeContact` 之前，不會出現在 `中`。若要檢查特定版本的狀態，請使用 [DescribeContactVersion](#)。

若要判斷更新是否成功或失敗：

1. 使用 呼叫 [DescribeContactVersion](#)，`contactId` 並由 `UpdateContact` 回應 `versionId` 傳回。
2. 檢查 `version.status` 欄位。狀態 `ACTIVE` 表示已成功套用更新。狀態 `FAILED_TO_UPDATE` 表示更新失敗。
3. 如果狀態為 `FAILED_TO_UPDATE`，請檢查 `version.failureCodes` 和 `version.failureMessage` 欄位以取得錯誤的詳細資訊。

Tip

AWS SDKs 和 AWS Command Line Interface 支援在版本達到 `ACTIVE` 或 `FAILED_TO_UPDATE` 狀態 `DescribeContactVersion` 之前自動輪詢的 `ContactUpdated` 等待程式。例如，AWS Command Line Interface 提供 [aws Groundstation 等待 contact-updated](#) 命令。使用等待程式，而不是實作您自己的輪詢邏輯。

故障診斷 FAILED_TO_SCHEDULE 聯絡人

當偵測到資源組態或內部系統的問題時 AWS Ground Station，聯絡人將結束為 FAILED_TO_SCHEDULE 狀態。結尾為 FAILED_TO_SCHEDULE 狀態的聯絡人將選擇性地提供 errorMessage 以用於其他內容。如需描述聯絡人的資訊，請參閱 [DescribeContact](#) API。

以下提供可能導致 FAILED_TO_SCHEDULE 聯絡人的常見使用案例，以及協助疑難排解的步驟。

Note

本指南專門用於 FAILED_TO_SCHEDULE 聯絡狀態，不適用於其他失敗狀態，例如 AWS_FAILED、AWS_CANCELLED 或 FAILED。如需聯絡狀態的詳細資訊，請參閱 [the section called “AWS Ground Station 聯絡狀態”](#)

不支援在天線下行解碼組態中指定的設定

用來排程此聯絡人的 [任務描述](#) 檔具有無效的 [antenna-downlink-demod-decode](#) 組態。

先前現有的 AntennaDownlinkDemodDecode 組態

- 如果您的 antenna-downlink-demod-decode 組態最近已變更 - 在嘗試排程之前復原至先前運作的版本。
- 如果這是現有組態的刻意變更，或先前未成功排程的現有組態 - 請依照下一個步驟，了解如何加入新的 AntennaDownlinkDemodDecode 組態。

新建立的 AntennaDownlinkDemodDecode 組態

AWS Ground Station 直接聯絡 以加入您的新組態。使用 [AWS Support](#) 建立案例 contactId，包括以 FAILED_TO_SCHEDULE 狀態結尾的

一般故障診斷步驟

如果上述疑難排解步驟無法解決您的問題：

- 使用相同的任務設定檔重新嘗試排程聯絡人或排程另一個聯絡人。如需如何保留聯絡人的資訊，請參閱 [ReserveContact](#)。
- 如果您繼續收到此任務設定檔的 FAILED_TO_SCHEDULE 狀態，[請聯絡 AWS Support](#)

針對未處於 HEALTHY 狀態的 DataflowEndpointGroups 進行故障診斷

下列是資料流程端點群組可能未處於 HEALTHY 狀態的原因，以及要採取的適當修正動作。

- NO_REGISTERED_AGENT - 啟動 EC2 執行個體，這會註冊代理程式。請注意，您必須擁有有效的控制器組態檔案，此呼叫才能成功。如需設定該檔案的詳細資訊，[使用 AWS Ground Station 代理程式](#)，請參閱。
- INVALID_IP_OWNERSHIP - 使用 DeleteDataflowEndpointGroup API 刪除 Dataflow 端點群組，然後使用 CreateDataflowEndpointGroup API，使用與 EC2 執行個體相關聯的 IP 地址和連接埠重新建立 Dataflow 端點群組。
- UNVERIFIED_IP_OWNERSHIP - IP 地址尚未驗證。驗證會定期進行，因此應該自行解析。
- NOT_AUTHORIZED_TO_CREATE_SLR - 帳戶未獲授權建立必要的服務連結角色。檢查 [中的疑難排解步驟 使用 Ground Station 的服務連結角色](#)

故障診斷無效的短暫性事件

當您將暫時性資料上傳至時 AWS Ground Station，它會經歷非同步驗證工作流程。如果驗證失敗，暫時性狀態會變更為 INVALID。[DescribeEphemeris](#) 回應中的錯誤訊息提供詳細資訊，協助您識別和解決問題。

了解暫時性驗證錯誤

當暫時性驗證失敗時，[DescribeEphemeris](#) API 回應會包含兩個欄位，以協助診斷問題：

errorCode

識別特定驗證錯誤的機器可讀取程式碼。這可用於程式設計錯誤處理。

errorMessage

驗證錯誤的人類可讀描述，其中包含錯誤的特定詳細資訊，以及如何修正錯誤的指引。

無效暫時性的 [DescribeEphemeris](#) 回應範例：

```
{
```

```
"ephemerisId": "abc12345-6789-def0-1234-567890abcdef",
"name": "My Invalid Ephemeris",
"status": "INVALID",
"creationTime": 1620254718.765,
"invalidReason": "METADATA_INVALID",
"errorCode": "OBJECT_NAME_MISSING",
"errorMessage": "Metadata field missing: OBJECT_NAME",
"suppliedData": {
  "tle": {
    "ephemerisData": "[...]"
  }
}
```

TLE ephemerides 的常見驗證錯誤

以下是上傳 TLE 暫時性資料時遇到的常見驗證錯誤：

不相符的衛星目錄編號

錯誤：「ephemeris 中存在的衛星目錄號碼與關聯的衛星目錄號碼不符」

解決方案：確認 TLE 行中的 NORAD ID/衛星目錄編號符合您衛星的衛星目錄編號。00000 用於沒有指派目錄編號的衛星。

無效的平均動作

錯誤：「所提供 ephemeris 的平均動作與最新的參考 ephemeris 差異太大」

解決方案：確認您的 TLE 資料正確，並代表有效的軌道。Ground Station 會在驗證期間使用 Space-Track 半聚體做為參考。

OEM ephemerides 的常見驗證錯誤

以下是上傳 OEM 暫時性資料時遇到的常見驗證錯誤：

無效的參考框架

錯誤：「不支援 REF_FRAME」

解決方案：更新您的 OEM 檔案以使用其中一個支援的參考框架：EME2000 或 ITRF2000。

缺少必要欄位

錯誤："Metadata 欄位遺失：INTERPOLATION"

解決方案：將 INTERPOLATION 和 INTERPOLATION_DEGREE 欄位新增至 OEM 中繼資料區段。這些是產生準確天線指向角度的必要 AWS Ground Station 項目。

不支援的時間系統

錯誤：「不支援 TIME_SYSTEM」

解決方案：確保您的 OEM 檔案使用 UTC 做為時間系統。

不支援的 OEM 版本

錯誤：「不支援 CCSDS_OEM_VERS」

解決方案：確保您的 OEM 檔案使用 CCSDS OEM 2.0 版。

方位角升高環羣化物的常見驗證錯誤

以下是上傳方位角海拔半合物時遇到的常見驗證錯誤：

遺失方位/上升資料

錯誤：「至少一個 TimeAzEl AzElSegment 欄位」

解決方案：確保方位提升資料中的每個區段都包含至少一個時間標記方位/提升對。

無效的方位角範圍（度）

錯誤：「AzEl az 必須大於或等於 -180 且小於或等於 360 度」

解決方案：確認方位角在 **【-180，360】** 度內。

無效的高度角度範圍（度）

錯誤：「AzEl el 必須大於或等於 -90 且小於或等於 90 度」

解決方案：確認高度角度在 **【-90，90】** 度內。

無效的方位角範圍（弧度）

錯誤：「AzEl az 必須大於或等於 -pi 且小於或等於 2pi 弧度」

解決方案：確認方位角位於 **【-π，2π】** 弧度內。

無效的高度角度範圍（弧度）

錯誤：「AzEl el 必須大於或等於 $-\pi/2$ 且小於或等於 $\pi/2$ 弧度」

解決方案：確認高度角度在 $[-\pi/2, \pi/2]$ 弧度內。

非單調時間值

錯誤：「AzElSegment 中的 TimeAzEl AzElSegment 項目必須暫時依順序排列」

解決方案：確保每個區段中的時間值嚴格增加。

區段不按順序排列

錯誤：「AzElSegments 必須暫時依順序排列」

解決方案：確保區段依時間順序排列。

重疊區段

錯誤：「至少一個區段的時間範圍與其他區段時間範圍重疊」

解決方案：確保每個客群都有唯一的非重疊時間範圍。一個區段endTime的 不應超過下一個區段startTime的。

疑難排解步驟

如果您的暫時性驗證失敗，請依照下列步驟解決問題：

1. 使用您的 ephemeris ID 呼叫 [DescribeEphemeris](#) 以擷取 errorCode 和 errorMessage。
2. 檢閱錯誤訊息，以取得驗證檢查失敗的特定詳細資訊。
3. 更正 ephemeris 資料中已識別的問題。
4. 使用 [CreateEphemeris](#) 上傳具有更正資料的新 ephemeris。
5. 監控新的暫時性狀態，直到其達到ENABLED狀態為止。
6. 如果不再需要，請使用 [DeleteEphemeris](#) 刪除無效的暫時性資料。

完成錯誤碼參考

下列各節提供 ephemeris 驗證失敗時可能傳回的所有errorCode值的完整映射，依高階invalidReason類別組織。

無效的原因：METADATA_INVALID

當所需的中繼資料欄位遺失、格式不正確或在 ephemeris 資料中包含不支援的值時，就會發生這些錯誤。

錯誤程式碼	錯誤訊息
MISMATCHED_SATCAT_ID	TLE ephemeris 中存在的衛星目錄號碼與關聯的衛星目錄號碼不相符
OEM_VERSION_UNSUPPORTED	不支援 OEM ephemeris CCSDS_OEM_VERS 中的 。支援的值：【2.0】
ORIGINATOR_MISSING	OEM ephemeris 缺少ORIGINATOR 標頭欄位
CREATION_DATE_MISSING	OEM ephemeris 缺少CREATION_DATE 標頭欄位
OBJECT_NAME_MISSING	OEM ephemeris 缺少OBJECT_NAME 中繼資料欄位
OBJECT_ID_MISSING	OEM ephemeris 缺少OBJECT_ID 中繼資料欄位
REF_FRAME_UNSUPPORTED	不支援 OEM ephemeris REF_FRAME 中的 。支援的值：【EME2000、ITRF2000】
REF_FRAME_EPOCH_UNSUPPORTED	不支援 OEM ephemeris 中的REF_FRAME_EPOCH 中繼資料欄位。請從 ephemeris 中移除此欄位
TIME_SYSTEM_UNSUPPORTED	不支援 OEM ephemeris TIME_SYSTEM 中的 。支援的值：【UTC】
CENTER_BODY_UNSUPPORTED	不支援 OEM ephemeris CENTER_BODY 中的 。支援的值：【Earth】
INTERPOLATION_MISSING	OEM ephemeris 缺少INTERPOLATION 中繼資料欄位
INTERPOLATION_DEGREE_INVALID	OEM ephemeris 中的插補程度必須大於插補方法的 0
AZ_EL_SEGMENT_LIST_MISSING	azElSegmentList 欄位遺失

錯誤程式碼	錯誤訊息
INSUFFICIENT_TIME_AZ_EL	至少一個 中沒有 TimeAzEl 欄位 azElSegmentList

無效的原因： **TIME_RANGE_INVALID**

當暫時性事件包含無效的時間範圍，包括開始/結束時間、區段排序、重疊區段或時間不一致的問題時，就會發生這些錯誤。

錯誤程式碼	錯誤訊息
START_TIME_IN_FUTURE	Ephemeris 開始時間是未來的時間，但必須是過去的時間
END_TIME_IN_PAST	Ephemeris 結束時間是過去的時間，但必須在未來
EXPIRATION_TIME_TOO_EARLY	提供的過期時間早於 ephemeris 結束時間
START_TIME_METADATA_TOO_EARLY	START_TIME 中繼資料值早於 OEM ephemeris 資料中存在的最早時間
STOP_TIME_METADATA_TOO_LATE	STOP_TIME 中繼資料值晚於 OEM ephemeris 資料中存在的最新時間
AZ_EL_SEGMENT_END_TIME_BEFORE_START_TIME	至少一個資料區段 endTime 的 早於區段的 startTime
AZ_EL_SEGMENT_TIME_S_OVERLAP	至少一個區段的時間範圍與其他區段時間範圍重疊
AZ_EL_SEGMENTS_OUT_OF_ORDER	區段不會暫時排序
TIME_AZ_EL_ITEMS_OUT_OF_ORDER	內的 TimeAzEl 項目 AzElSegment 必須暫時依順序排列
AZ_EL_SEGMENT_REFERENCE_EPOCH_INVALID	區段的參考 epoch 無效或格式不正確

錯誤程式碼	錯誤訊息
AZ_EL_SEGMENT_START_TIME_INVALID	區段有效時間範圍中的開始時間不會在第一個區段之後開始
AZ_EL_SEGMENT_END_TIME_INVALID	區段有效時間範圍中的結束時間不會在最後一個區段之後結束
AZ_EL_SEGMENT_VALID_TIME_RANGE_INVALID	區段的有效時間範圍無效
AZ_EL_SEGMENT_END_TIME_TOO_LATE	區段的結束時間超過參考 epoch 允許的持續時間上限
AZ_EL_TOTAL_DURATION_EXCEEDED	所有區段的總持續時間超過允許的指標角度持續時間上限

無效的原因： **TRAJECTORY_INVALID**

當暫時性資料包含無效的軌跡資料，包括軌道參數、角度範圍或單位的問題時，就會發生這些錯誤。

錯誤程式碼	錯誤訊息
MEAN_MOTION_INVALID	所提供 TLE ephemeris 的平均動作與最新的參考 ephemeris 差異太大。注意：Ground Station 會在驗證期間使用 Space-Track 暫時性資料作為參考
TIME_AZ_EL_AZ_RADIAN_RANGE_INVALID	AzEl az 必須大於或等於 $-\pi$ 且小於或等於 2π 弧度
TIME_AZ_EL_EL_RADIAN_RANGE_INVALID	AzEl el 必須大於或等於 $-\pi/2$ 且小於或等於 $\pi/2$ 弧度
TIME_AZ_EL_AZ_DEGREE_RANGE_INVALID	AzEl az 必須大於或等於 -180 且小於或等於 360 度
TIME_AZ_EL_EL_DEGREE_RANGE_INVALID	AzEl el 必須大於或等於 -90 度，且小於或等於 90 度

錯誤程式碼	錯誤訊息
TIME_AZ_EL_ANGLE_UNITS_INVALID	無效的 AzEl 角度單位

無效的原因：**KMS_KEY_INVALID**

當用於加密暫時性資料的 AWS Key Management Service (KMS) 金鑰發生問題時，就會發生這些錯誤。

錯誤程式碼	錯誤訊息
INSUFFICIENT_KMS_PERMISSIONS	Ground Station 沒有足夠的許可來存取此 ephemeris 的 KMS 金鑰

無效的原因：**VALIDATION_ERROR**

當 ephemeris 資料的一般驗證問題不屬於其他特定類別時，就會發生這些錯誤。

錯誤程式碼	錯誤訊息
INTERNAL_ERROR	ephemeris 驗證期間發生內部錯誤
FILE_FORMAT_INVALID	ephemeris 檔案格式無效或損毀。驗證檔案符合 ephemeris 類型的預期格式

對未收到資料的聯絡人進行故障診斷

聯絡人可能會成功顯示，但仍未收到任何資料。這可能表示您收到空白的 PCAP 檔案，或者如果您使用 S3 資料交付，則完全沒有 PCAP 檔案。這可能有許多原因。以下討論一些原因，以及如何解決這些原因。

下行組態不正確

從衛星接收資料的每個聯絡人都會有關聯的 [天線下行組態](#) 或 [天線下行解調解碼組態](#)。如果指定的組態與衛星傳輸的訊號不一致，AWS Ground Station 將無法接收傳輸的訊號。這將導致未收到任何資料 AWS Ground Station。

若要修正此問題，請確認您使用的組態同意衛星傳輸的訊號。例如，確認您已設定正確的中心頻率、頻寬、極化，並視需要解調和解碼參數。

衛星動作

有時候，衛星可能會執行動作，暫時停用其部分通訊系統。操作也可能大幅變更衛星在天空中的位置。AWS Ground Station 將無法接收來自未傳輸訊號之衛星的訊號，或者如果使用的暫時性訊號導致 AWS Ground Station 天線指向不存在衛星的天空中位置。

如果您嘗試與 NOAA 營運的公有廣播衛星通訊，您可以在 NOAA [衛星提醒訊息](#) 頁面上找到描述中斷或操作的訊息。訊息可能包含預期資料傳輸會恢復的時間軸，或可能張貼在後續訊息中。

如果您正在與自己的衛星通訊，您有責任了解您的衛星操作，以及這可能會影響與的通訊 AWS Ground Station。如果您執行的動作會影響衛星軌跡，這可能包括提供更新的自訂暫時性資料。如需提供自訂暫時性資料的詳細資訊，請參閱 [了解 AWS Ground Station 如何使用 ephemerides](#)。

AWS Ground Station 中斷

如果 AWS Ground Station 導致聯絡失敗或取消，AWS Ground Station 會將聯絡狀態設定為 `AWS_FAILED` 或 `AWS_CANCELLED`。如需聯絡生命週期的詳細資訊，請參閱 [了解聯絡生命週期](#)。在某些情況下，AWS Ground Station 可能會失敗，導致資料無法交付至您的帳戶，但不會導致聯絡人處於 `AWS_FAILED` 或 `AWS_CANCELLED` 狀態。發生這種情況時，AWS Ground Station 應該將帳戶特定事件發佈至您的 AWS 運作狀態儀表板。如需運作狀態儀表板的詳細資訊 AWS，請參閱 [AWS 運作狀態使用者指南](#)。

對遙測進行故障診斷

使用下列資訊來疑難排解遙測的常見問題。

常見的設定問題

IAM 許可錯誤

徵狀

呼叫 `CreateConfig` 建立 `TelemetrySinkConfig` 時，您會收到錯誤：

```
Unable to write to Kinesis Data Streams stream. Ensure that Ground Station has kinesis:PutRecord permissions for the given stream
```

原因

- `TelemetrySinkConfig` 中指定的 IAM 角色沒有寫入 Kinesis Data Streams 串流所需的許可。
- IAM 角色上的信任政策不允許 AWS Ground Station 擔任該角色。
- `TelemetrySinkConfig` 中的 Kinesis Data Streams 串流 ARN 不正確或串流不存在。

解決方案

1. 確認 IAM 角色存在且具有正確的許可。檢閱[步驟 2：建立 TelemetrySinkConfig](#)並確保遵循所有步驟。
2. 檢查 AWS Ground Station 是否可以擔任您的 IAM 角色：

```
aws iam get-role --role-name GroundStationTelemetryRole
```

確認信任政策包含 `groundstation.amazonaws.com` 做為信任的服務委託人。

3. 確認 IAM 角色具有必要的 Kinesis 許可：

```
aws iam list-attached-role-policies --role-name GroundStationTelemetryRole
```

確保政策包含串流的 `kinesis:DescribeStream`、`kinesis:PutRecord`、和 `kinesis:PutRecords` 許可。

4. 確認 Kinesis Data Streams 串流存在且 ARN 正確：

```
aws kinesis describe-stream \  
  --stream-name your-stream-name \  
  --region us-east-2
```

5. 如果使用客戶受管加密，請確認 IAM 角色具有 AWS KMS 金鑰的 `kms:GenerateDataKey` 許可。

PassRole 許可錯誤

徵狀

呼叫時 `CreateConfig`，您會收到有關沒有許可傳遞 IAM 角色的錯誤。

解決方案

確保您的 IAM 使用者或角色具有遙測 IAM 角色的 `iam:PassRole` 許可。將下列政策新增至您的使用者或角色：

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "iam:GetRole",
        "iam:PassRole"
      ],
      "Resource": "arn:aws:iam::9999999999:role/your-stream-name"
    }
  ]
}
```

Kinesis Data Streams 串流組態問題

徵狀

遙測交付失敗或間歇性。

原因

- Kinesis Data Streams 串流的容量不足以提供遙測輸送量。
- 其他應用程式正在使用串流，導致寫入限流。

解決方案

1. 檢查串流狀態：

```
aws kinesis describe-stream \
```

```
--stream-name your-stream-name \  
--region us-east-2
```

2. 使用 CloudWatch 指標監控寫入限流：

```
aws cloudwatch get-metric-statistics \  
--namespace AWS/Kinesis \  
--metric-name WriteProvisionedThroughputExceeded \  
--dimensions Name=StreamName,Value=your-stream-name \  
--start-time 2025-12-08T00:00:00Z \  
--end-time 2025-12-08T23:59:59Z \  
--period 60 \  
--statistics Sum \  
--region us-east-2
```

3. 如果偵測到限流，請考慮：

- 切換到隨需容量模式以進行自動擴展。
- 使用專用串流進行 AWS Ground Station 遙測。
- 如果使用佈建模式，請增加碎片數量。

遙測交付問題

沒有顯示遙測資料

徵狀

使用啟用遙測功能的任務設定檔排定聯絡後，Kinesis Data Streams 串流中不會顯示遙測資料。

可能的原因和解決方案

任務描述檔未啟用遙測

確認用於聯絡人的任務描述檔包含 `telemetrySinkConfigArn`：

```
aws groundstation get-mission-profile \  
--mission-profile-id 12345678-1234-1234-1234-123456789012 \  
--region us-east-2
```

檢查 `telemetrySinkConfigArn` 欄位的輸出。如果不存在，則任務設定檔未啟用遙測。

IAM 角色許可問題

檢閱 中的 IAM 許可疑難排解步驟 [IAM 許可錯誤](#)。

Kinesis Data Streams 串流不存在或位於錯誤的區域

確認串流存在於正確的區域中：

```
aws kinesis describe-stream \  
  --stream-name your-stream-name \  
  --region us-east-2
```

聯絡人尚未開始

遙測交付從聯絡開始時間開始。檢查聯絡人狀態，確認聯絡人已開始：

```
aws groundstation describe-contact \  
  --contact-id 12345678-1234-1234-1234-123456789012 \  
  --region us-east-2
```

間歇性遙測資料

徵狀

遙測資料交付的方式與差距或遺失記錄不一致。

可能的原因

- Kinesis Data Streams 串流容量問題或限流。請參閱 [Kinesis Data Streams 串流組態問題](#)。
- AWS Ground Station 與 Kinesis Data Streams 串流之間的網路連線問題。

解決方案

- 在 CloudWatch 中監控 Kinesis Data Streams 串流指標是否有限流或錯誤。
- 確保您的串流使用隨需容量模式或有足夠的佈建容量。
- 使用專用串流進行 AWS Ground Station 遙測，以避免與其他應用程式爭用。

資料格式問題

JSON 剖析錯誤

徵狀

您的應用程式在剖析遙測記錄為 JSON 時發生錯誤。

解決方案

- 驗證 Base64 解碼 - Kinesis Data Streams 串流中的資料是 Base64-encoded。務必先解碼資料，再將其剖析為 JSON。如需詳細資訊，請參閱[從 Kinesis Data Streams 串流讀取資料](#)。
- 檢查空白記錄 - 建立 TelemetrySinkConfig 時 AWS Ground Station 可傳送空白驗證記錄。您的應用程式應正常處理空白或格式不正確的記錄。
- 實作版本感知剖析 - 首先剖析 telemetryTypeAndVersion、telemetryType 和 telemetryVersion 欄位，以判斷每個記錄的適當結構描述。

未知遙測類型或版本

徵狀

您的應用程式遇到遙測類型或無法辨識的版本。

解決方案

這是預期的行為，因為新的遙測類型和結構描述版本可能會隨著時間推移而引入。您的應用程式應該：

- 記錄未知類型和版本以進行監控。
- 繼續處理已知類型和版本。
- 針對未知結構描述實作優雅處理。

如需結構描述版本控制的詳細資訊，請參閱[結構描述版本控制和演變](#)。

取得說明

如果您在遵循疑難排解步驟之後持續遇到問題，請聯絡 AWS Support。

要提供的資訊

聯絡支援時，請提供下列資訊：

- 遇到問題的聯絡人 IDs
- 使用的任務設定檔 ID
- TelemetrySinkConfig ARN
- Kinesis Data Streams 串流 ARN
- IAM 角色 ARN 和連接的政策
- CloudWatch Logs 或您的應用程式的錯誤訊息
- 發生問題時的時間戳記
- 已採取的故障診斷步驟

如需一般 AWS Ground Station 支援，請參閱[AWS Ground Station 《使用者指南》](#)。

配額和限制

您可以在端點和配額中檢視支援的區域、其相關聯的[AWS Ground Station 端點和配額](#)。

您可以使用 [Service Quotas 主控台](#)、[AWS API](#)與[AWS CLI](#)在需要時請求增加配額。

服務條款

如需 AWS Ground Station 服務條款，請參閱 [AWS 服務條款](#)。

AWS Ground Station 使用者指南的文件歷史記錄

下表說明 AWS Ground Station 使用者指南每個版本的重要變更。

變更	描述	日期
新功能	新增 AWS Ground Station 專用天線的文件。如需詳細資訊，請參閱 AWS Ground Station 專用天線 。	2026 年 4 月 14 日
新功能	新增 UpdateContact API 和聯絡版本控制的文件。如需詳細資訊，請參閱 更新聯絡人和聯絡人版本控制 。	2026 年 4 月 14 日
新功能	新增 ListAntennas 和 ListGroundStationReservations APIs 的文件。如需詳細資訊，請參閱 AWS Ground Station 位置 和 檢視地面站保留 。	2026 年 4 月 14 日
文件更新	新增其他功能至 CancelContact API，並包含上述功能和計量影響的相關資訊。如需詳細資訊，請參閱 了解聯絡計量 。	2025 年 12 月 10 日
文件更新	釐清 CloudWatch 指標會在與聯絡人的地面工作站相關聯的區域中發出。已修正中斷的連結。	2025 年 12 月 2 日
更新 AWS 受管政策	AWS Ground Station 已更新受管政策AWSGroundStationAgentInstancePolicy，以包含擷取任務回應 URLs 的其他許可。	2025 年 11 月 13 日

	如需詳細資訊，請參閱 AWS Ground Station AWS 受管政策的更新 。	
新功能	更新 使用者指南，以包含方位角海拔半合子。如需詳細資訊，請參閱 提供方位提升暫時性資料	2025 年 10 月 22 日
文件更新	跨區域資料交付不再需要特殊組態或核准。如需詳細資訊，請參閱 使用跨區域資料交付 。	2025 年 9 月 11 日
文件更新	新增有關已設定資源之聯絡使用率的說明。	2025 年 4 月 4 日
新功能	更新使用者指南以包含 AWS Ground Station 數位分身。	2024 年 8 月 6 日
文件更新	已更新使用者指南的許多章節，包括新圖表、範例等。	2024 年 7 月 18 日
文件更新	已將 RSS 摘要新增至 使用者指南。	2024 年 7 月 18 日
文件更新	將 AWS Ground Station 客服人員使用者指南分割為單獨的使用者指南。	2024 年 7 月 18 日
新功能	聯絡人現在可以在可見性時間範圍之外排程最多 30 秒。可見性時間包含在 DescribeContact 回應中。	2024 年 3 月 26 日
文件更新	改善組織並新增「EC2 執行個體選擇和 CPU 規劃」區段。	2024 年 3 月 6 日

文件更新	將新的最佳實務新增至 AWS Ground Station 代理程式使用者指南，以搭配 AWS Ground Station 代理程式執行服務和程序。	2024 年 2 月 23 日
文件更新	新增客服人員版本備註頁面。	2024 年 2 月 21 日
範本更新	新增對 DirectBroadcastSatelliteWbDigIfEc2DataDelivery 範本中個別公有子網路的支援。	2024 年 2 月 14 日
文件更新	在監控文件中新增對 AWS 使用者通知的推薦。	2023 年 8 月 6 日
文件更新	新增了使用要在主控台中 AWS Ground Station 顯示的名稱標記衛星的說明。	2023 年 7 月 26 日
新功能	新增了適用於寬頻 DigIF 資料交付版本的 AWS Ground Station 代理程式使用者指南。	2023 年 4 月 12 日
新的 AWS 受管政策	AWS Ground Station 新增名為 AWSGroundStationAgentInstancePolicy 的新政策。	2023 年 4 月 12 日
新功能	更新 CPE 預覽版本的使用者指南。	2022 年 11 月 9 日

新的 AWS 受管政策	AWS Ground Station 新增了 AWSServiceRoleForGroundStationDataflowEndpointGroup service-linked-role(SLR)，其中包含名為 AWSServiceRoleForGroundStationDataflowEndpointGroupPolicy 的新政策。	2022 年 11 月 2 日
新功能	更新 使用者指南以包含的整合 AWS CLI。	2020 年 4 月 17 日
新功能	已更新使用者指南，以包含與 CloudWatch 指標的整合。	2020 年 2 月 24 日
新範本	公有廣播衛星 (AquaSnppJpss 範本) 已新增至 AWS Ground Station 使用者指南。	2020 年 2 月 19 日
新功能	已更新使用者指南，加入跨區域資料傳送。	2020 年 2 月 5 日
文件更新	更新 AWS Ground Station 使用 CloudWatch Events 進行監控的範例和說明。	2020 年 2 月 4 日
文件更新	範本位置已更新，且已修訂「入門」和「故障診斷」章節。	2019 年 12 月 19 日
新的故障診斷區段	故障診斷章節已新增至 AWS Ground Station 使用者指南。	2019 年 11 月 7 日
新的入門主題	已更新入門主題，其中包含最新的 CloudFormation 範本。	2019 年 7 月 1 日
Kindle 版本	AWS Ground Station 使用者指南的發佈 Kindle 版本。	2019 年 6 月 20 日

[新的服務與指南](#)

這是 AWS Ground Station 和 2019 年 5 月 23 日
AWS Ground Station 使用者指
南的初始版本。

AWS 詞彙表

如需最新的 AWS 術語，請參閱 AWS 詞彙表 參考中的 [AWS 詞彙表](#)。

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。