



Uso de la integración del orquestador de flujos de trabajo de Control-M con
AWS Mainframe Modernization

Recomendaciones de AWS



Recomendaciones de AWS: Uso de la integración del orquestador de flujos de trabajo de Control-M con AWS Mainframe Modernization

Copyright © 2023 Amazon Web Services, Inc. and/or its affiliates. All rights reserved.

Las marcas comerciales y la imagen comercial de Amazon no se pueden utilizar en relación con ningún producto o servicio que no sea de Amazon de ninguna manera que pueda causar confusión entre los clientes y que menosprecie o desacredite a Amazon. Todas las demás marcas comerciales que no son propiedad de Amazon son propiedad de sus respectivos propietarios, que pueden o no estar afiliados, conectados o patrocinados por Amazon.

Table of Contents

Introducción	1
Información general	1
Resultados específicos	3
Términos de esta guía	3
Supuestos	4
Arquitectura	6
Integración de flujos de trabajo	7
Implementar un tipo de trabajo	7
Crear un perfil de conexión	8
Crear trabajos y programas	9
Iniciar los trabajos mediante un flujo de trabajo	10
Automatizar los trabajos en función de un programa	13
El trabajo base se ejecuta en función de eventos	14
Monitorear trabajos	14
Control-M Monitoring	14
La consola de administración de AWS	15
Prácticas recomendadas	16
Recursos relacionados	17
Historial de documentos	18
Glosario	19
Términos de DevOps	19
Términos de migración	22
Términos de modernización	30

Uso de la integración del orquestador de flujos de trabajo de Control-M con AWS Mainframe Modernization

Sunil Bemarker, Pablo Alonso Prieto, Vaidy Sankaran y Vij Balakrishna de Amazon Web Services (AWS) y Joe Goldberg de BMC Software, Inc.

noviembre de 2022

Para satisfacer las necesidades de una transformación digital y empresarial, los clientes modernizan las aplicaciones de unidad central a fin de lograr una mayor agilidad, reducir los costos e innovar más rápido. En re:Invent 2021, Amazon Web Services (AWS) anunció el servicio [AWS Mainframe Modernization](#) para ayudar a los clientes a modernizar sus cargas de trabajo de unidad central. AWS Mainframe Modernization brinda un entorno de tiempo de ejecución administrado y de alta disponibilidad nativo en la nube en AWS.

En esta guía, se brinda información detallada sobre cómo se puede integrar la orquestación del flujo de trabajo de Control-M con el servicio AWS Mainframe Modernization para ofrecerle una opción adicional a fin de modernizar sus aplicaciones por lotes durante su recorrido de modernización de la unidad central. Al utilizar las capacidades de Control-M, las organizaciones pueden racionalizar un entorno compuesto por cargas de trabajo existentes y migradas mediante una única herramienta que ofrece una visión integral.

Información general

La integración de Control-M y AWS Mainframe Modernization ofrece los siguientes beneficios:

- Ayuda a reducir las brechas de talento
- Admite la innovación rápida con un enfoque de DevOps ágil
- Brinda un acceso más sencillo a las aplicaciones y los datos sin cambios significativos
- Optimiza los costos de ejecutar o ampliar las aplicaciones
- Maximiza la agilidad empresarial y reduce los costos

El servicio AWS Mainframe Modernization lo ayuda a modernizar sus aplicaciones de unidad central para convertirlas en entornos de tiempo de ejecución administrados nativos en la nube de AWS. Ofrece herramientas y recursos para ayudarlo a planificar e implementar la migración y

modernización. Los usuarios pueden enviar o cancelar los trabajos por lotes y analizar los detalles de las ejecuciones de estos trabajos. Cada vez que un usuario envía un trabajo por lotes, el servicio AWS Mainframe Modernization crea una ejecución de trabajo por lotes independiente, que se puede monitorear. Con la consola web del servicio AWS Mainframe Modernization, puede buscar trabajos por lotes mediante nombre, proporcionar el lenguaje de control de trabajos (JCL), archivos de script y parámetros para los trabajos por lotes.

Con Control-M, puede definir, programar, administrar y monitorear los flujos de trabajo de aplicaciones y datos. Esto optimiza la visibilidad y la fiabilidad, lo que ayuda a mejorar los acuerdos de nivel de servicio (SLA). Control-M integra, automatiza y orquesta los flujos de trabajo en las instalaciones y en la nube de AWS para ayudarlo a prestar servicios empresariales a tiempo. Con una única vista gráfica unificada, los usuarios pueden orquestar todos sus flujos de trabajo, incluidas las transferencias de archivos, las aplicaciones, los orígenes de datos y la infraestructura, con una amplia biblioteca de complementos. Control-M, aprovisionado en una nube, utiliza las capacidades efímeras de la nube de AWS. Debido a que utilizan un enfoque de trabajo como código, con las API de REST, JSON y Python dentro de la cadena de herramientas de integración y entrega continuas (CI/CD), los flujos de trabajo de Control-M se pueden versionar, probar y mantener para que los desarrolladores y los ingenieros de DevOps puedan trabajar en colaboración.

Resultados empresariales específicos

Esta solución lo ayuda a lograr los siguientes objetivos:

- Amplíe su entorno de Control-M para administrar las cargas de trabajo en el servicio AWS Mainframe Modernization.
- Incorpore las API de REST de Control-M a los procesos con rapidez mediante el uso de asistentes.
- Orqueste, programe y monitoree todos los trabajos de carga de trabajo de sus aplicaciones en la vista unificada de Control-M.
- Aumente la escalabilidad y la agilidad empresarial.

Términos de esta guía

- Una aplicación de unidad central hace referencia a un conjunto de programas y subprogramas relacionados de la unidad central que llevan a cabo y facilitan un conjunto de procesos empresariales. Las aplicaciones de unidad central pueden ser sistemas de procesamiento por lotes o de transacciones en línea (OLTP).
- Un trabajo por lotes hace referencia a un programa planificado que se ha configurado para ejecutarse sin requerir la interacción del usuario. En AWS Mainframe Modernization, tanto los archivos de JCL de los trabajos por lotes como los binarios de los trabajos por lotes se preparan en un bucket de Amazon Simple Storage Service (Amazon S3), y la ubicación de ambos se encuentra en el archivo de definición de la aplicación.
- [AWS Mainframe Modernization](#) es un servicio nativo en la nube para migrar, modernizar, ejecutar y operar aplicaciones de unidad central.
- [Amazon S3](#) es un almacenamiento de objetos escalable con alta durabilidad, disponibilidad y rendimiento.
- [Amazon CloudWatch](#) es un servicio de monitoreo y observabilidad que se ha creado para ingenieros de DevOps, desarrolladores, ingenieros de confiabilidad del sitio (SRE), administradores de TI y propietarios de productos.
- Control-M Web es una solución para administrar las cargas de trabajo empresariales, incluidos los trabajos por lotes de unidad central, durante todo el ciclo de vida de creación, prueba, implementación, programación y monitoreo de los flujos de trabajo de las aplicaciones empresariales. Un amplio conjunto de funciones gráficas y programáticas proporciona a todos los usuarios acceso a las funciones de Control-M de la forma en que se sientan más cómodos.

- Application Integrator de Control-M es un diseñador de bajo código basado en la Web que se utiliza para crear tipos de trabajo que permiten la integración entre aplicaciones o servicios en la nube con Control-M. Una vez que se crean los tipos de trabajo de Application Integrator, funcionan exactamente igual que todos los demás trabajos de Control-M, con lo que se exponen todas las características y funciones de Control-M para la integración de destino. En esta guía, la integración se realiza con AWS Mainframe Modernization.
- El dominio de planificación es un conjunto de capacidades para administrar la creación y actualización de los flujos de trabajo.
- El dominio de monitoreo es un conjunto de capacidades para administrar los flujos de trabajo. Aquí puede ver todos los trabajos en ejecución y su estado. Si se producen errores, se generan alertas. Puede realizar acciones operativas para ver el estado del trabajo, reaccionar ante retrasos o errores, analizar los problemas y tomar medidas correctivas.
- Los agentes de Control-M se instalan en diferentes equipos administrados por un servidor de Control-M. Los trabajos se asignan a un nombre de agente que identifica un host. El agente se comunica con el servidor mediante el protocolo HTTPS (puerto 443).

Supuestos

Los ejemplos y diagramas de esta guía reflejan las siguientes suposiciones:

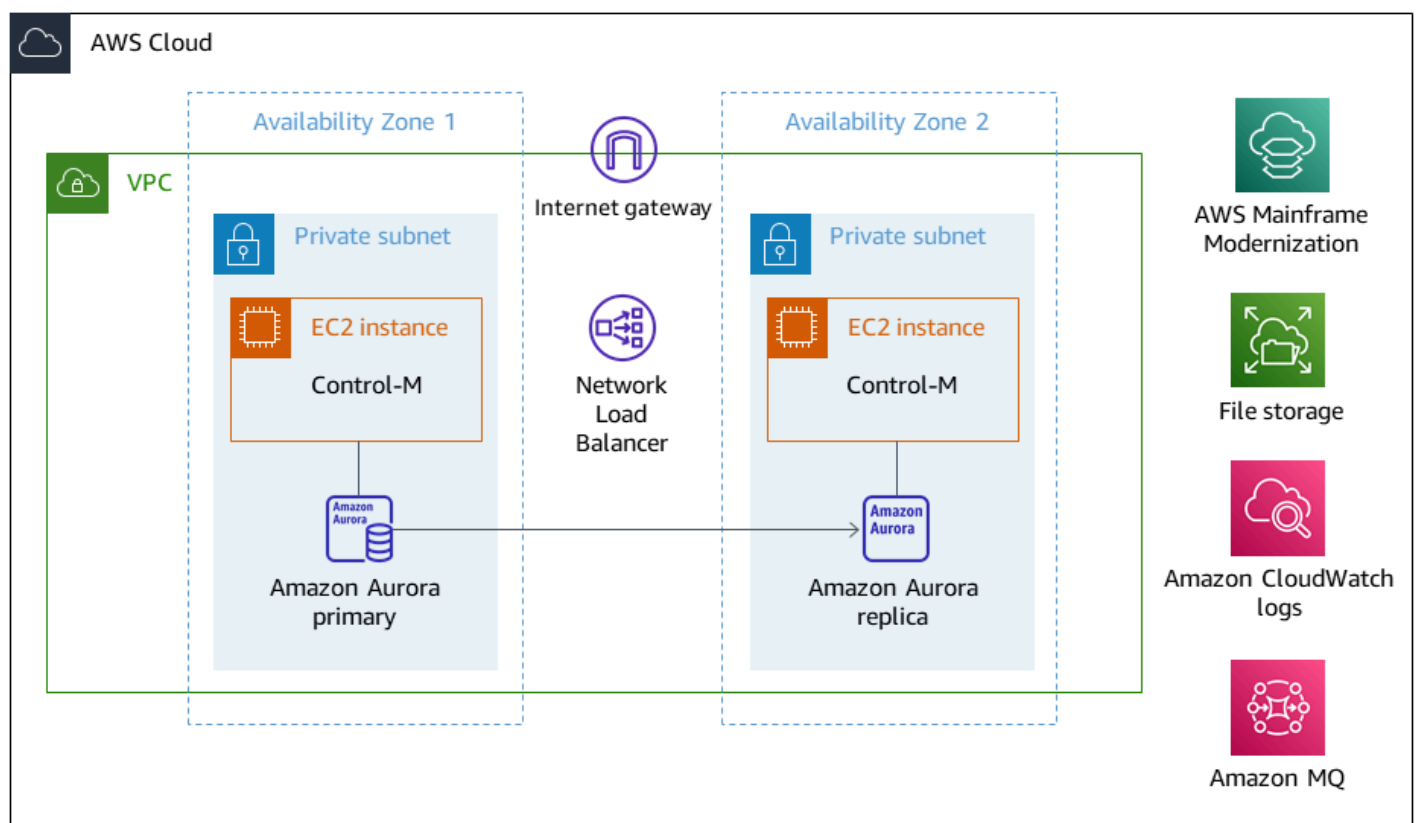
- La aplicación de unidad central que se migra puede ejecutar un solo programa o varios programas. Para simplificar, los diagramas de esta guía muestran un solo programa y varios subprogramas para cada aplicación.
- La aplicación de unidad central se migra y se ejecuta en un entorno de tiempo de ejecución administrado por AWS Mainframe Modernization con varios trabajos por lotes definidos. Para este piloto, configure la aplicación de ejemplo BankDemo en AWS Mainframe Modernization mediante las instrucciones de [Tutorial: Tiempo de ejecución administrado para Micro Focus](#).
- Una [instalación de Control-M](#) completa se encuentra disponible con todos los componentes de Control-M, incluidos los complementos y extensiones de la aplicación.
- El [agente de Control-M](#) es responsable de administrar los trabajos. Para distribuir la carga de trabajo, los agentes se pueden instalar en varios equipos. Esto puede mejorar el rendimiento y la resistencia.
- [Control-M Automation API](#) expone la funcionalidad de Control-M a través de los servicios web Restful (API de REST). Los artefactos como los trabajos, los perfiles de conexión, los usuarios y roles, y los estándares del sitio pueden escribirse en JSON y proporcionarse como entrada a

varios servicios o generarlos como salida. Se puede acceder a los servicios directamente a través de solicitudes HTTPS mediante cURL o funciones similares o las que proporciona la interfaz de la línea de comandos (CLI) de ctm.

Arquitectura

El tiempo de ejecución completamente administrado de AWS Mainframe Modernization cuenta con componentes compatibles con la unidad central de [Micro Focus](#) preinstalados. Estos incluyen administradores de transacciones, herramientas de asignación de datos, lectores de pantalla y mapas, y entornos de ejecución por lotes. Puede utilizar estos componentes para ejecutar aplicaciones con cambios mínimos en el código fuente.

En el siguiente diagrama, se muestra la integración del flujo de trabajo en la que Control-M se encuentra alojado en una instancia de Amazon Elastic Compute Cloud (Amazon EC2) con alta disponibilidad y orquesta los trabajos por lotes y los datos de las aplicaciones en el servicio AWS Mainframe Modernization. Control-M utiliza una base de datos de Amazon Aurora para mantener las configuraciones necesarias a fin de ejecutar trabajos por lotes.



Escenarios de integración de flujos de trabajo

En esta sección, se describe la configuración paso a paso necesaria para la integración entre Control-M Scheduler y el entorno de AWS Mainframe Modernization, y se describen los diferentes tipos de flujos de trabajo de integración:

- [Implementar un tipo de trabajo de AWS Mainframe Modernization en Control-M Application Integrator](#)
- [Crear un perfil de conexión de Control-M para AWS Mainframe Modernization](#)
- [Crear trabajos y programas en Control-M Planning](#)
- [Monitorear trabajos](#)

Implementar un tipo de trabajo de AWS Mainframe Modernization en Control-M Application Integrator

La implementación del tipo de trabajo suele ser una actividad que se realiza una sola vez. Si ya cuenta con un tipo de trabajo existente que desea utilizar, omita este paso y continúe con el siguiente para [crear un perfil de conexión](#).

El tipo de trabajo de ejemplo [AIJOB.ctmai](#) se suministra en el repositorio de git. Para implementarlo, debe realizar los siguientes pasos mediante [Application Integrator](#):

- Clone el repositorio de GitHub [aws-mainframe-modernization-controlm-integration](#) y descargue el archivo `AIJOB.ctmai` en una ubicación del sistema de archivos a la que pueda acceder Application Integrator.
- Inicie sesión en Application Integrator.
- En la pestaña Inicio, seleccione Importar el tipo de trabajo desde un archivo y seleccione la ubicación de `AIM2JOB.ctmai`.
- Si desea realizar alguna modificación en el ejemplo suministrado, familiarícese con Application Integrator.
- Implemente el tipo de trabajo al seguir las instrucciones de la [Documentación de Control-M](#).

Crear un perfil de conexión de Control-M para AWS Mainframe Modernization

Los perfiles de conexión definen los atributos de conexión y las credenciales de seguridad para una instancia específica de una aplicación. Varios trabajos pueden hacer referencia a cada perfil de conexión. Puede tener perfiles independientes para cada combinación única de una aplicación y credenciales.

Para definir los perfiles de conexión, puede utilizar la interfaz gráfica de usuario (GUI) disponible en el dominio de configuración de la interfaz web de Control-M, o puede utilizar JSON. El siguiente código es un ejemplo de uso de JSON.

```
{
  "JOB": {
    "Type": "ConnectionProfile:ApplicationIntegrator:AI JOB",
    "AI-Access Key Id": "AWS Access Key ",
    "AI-Secret Access Key": "Secret Access Key value",
    "AI-RESTHost": "AWS M2 Endpoint host",
    "AI-region": "AWS region",
    "Description": "This is a connection profile for AWS Mainframe Modernization",
    "Centralized": true
  }
}
```

Cree un archivo JSON similar al del ejemplo e impleméntelo mediante el [Servicio de implementación de Control-M Automation API](#). Por ejemplo, si el código JSON se guarda en un archivo denominado [cp-job.json](#), la sintaxis de la CLI ctm para implementar este perfil de conexión es la siguiente.

```
ctm deploy cp-JOB.json
```

La respuesta de Control-M Automation API será similar a la siguiente.

```
{
  "deploymentFile": "cp_JOB.json",
  "deploymentState": "DEPLOYED_CONNECTION_PROFILES",
  "deploymentStatus": "ENDED_OK",
  "successfulFoldersCount": 0,
  "successfulSmartFoldersCount": 0,
  "successfulSubFoldersCount": 0,
}
```

```
"successfulJobsCount": 0,  
"successfulConnectionProfilesCount": 1,  
"successfulDriversCount": 0,  
"isDeployDescriptorValid": false,  
"deployedConnectionProfiles": [  
  "JOB"  
]  
}
```

Crear trabajos y programas en Control-M Planning

Ahora que ha implementado el tipo de trabajo y un perfil de conexión para la conectividad de AWS, podemos comenzar a crear y ejecutar trabajos.

Cada trabajo del servicio AWS Mainframe Modernization consta de un conjunto de atributos divididos en cuatro secciones. Cada sección puede tener muchos atributos. En la siguiente lista, se muestran algunos de los atributos que se utilizan con más frecuencia.

- General:
 - El nombre del trabajo
 - La aplicación y la subaplicación a las que pertenece el trabajo
 - El JCL a enviar
 - Un enlace a la documentación
- Programación:
 - Meses y días en los que se puede ejecutar este trabajo
 - Calendarios como periodos contables empresariales, feriados u otras fechas especiales que no se pueden definir de forma algorítmica
 - Ventanas de tiempo
 - Comportamiento cíclico
- Requisitos previos:
 - Dependencias ascendentes (por lo general trabajos, que se deben completar de forma correcta antes de que este trabajo pueda ejecutarse)
 - Recursos que podrían necesitarse
 - Acción del usuario que podría necesitarse
- Acciones que realiza Control-M al finalizar el trabajo:

- Determinar el éxito o el fracaso del trabajo (por lo general, en función de los códigos de finalización del trabajo, pero puede anular esa configuración para utilizar el texto de salida o comprobar un estado específico)
- Notificaciones, como por correo electrónico, en caso de error o éxito
- Estado de publicación de las dependencias descendentes

Al igual que con los perfiles de conexión, los trabajos se pueden crear e implementar [en la GUI](#) o [en JSON](#), e implementar mediante Control-M Automation API.

En las siguientes secciones, se examinan algunos escenarios de flujo de trabajo comunes:

- [Iniciar los trabajos en función del código de estado de los trabajos anteriores](#)
- [Automatizar la ejecución de los trabajos con una frecuencia programada](#)
- [El trabajo base se ejecuta en función de eventos](#)

Iniciar los trabajos en función del estado de los trabajos anteriores

Cree un flujo de trabajos denominado flujo de trabajo. Los trabajos del flujo de trabajo se encuentran interconectados y dependen de la finalización exitosa de los trabajos anteriores.

Uso de la GUI

1. En el dominio de planificación, agregue un espacio de trabajo nuevo. Esto abre un lienzo con un objeto de carpeta vacía.
2. Seleccione el tipo de trabajo de AWS Mainframe Modernization (si utiliza la plantilla de tipo de trabajo proporcionada, que se denomina [M2JOB](#)) y arrástrelo a la carpeta.
3. Cuando el color del tipo de trabajo cambie a verde, puede soltarlo. El panel de la derecha contiene las secciones General, Programación, Requisitos previos y Acciones. Consulte [aquí](#) para obtener [instrucciones de creación de trabajos](#) de Control-M estándar.
4. Luego, necesitará los valores del nombre del trabajo para configurar los tipos de trabajo, que puede encontrar en la pantalla de Definición de aplicaciones de la consola de AWS Mainframe Modernization o al ejecutar la [API ListBatchJobDefinitions](#). En el caso de este piloto, repita los pasos 2 a 4 para cinco trabajos, al nombrar cada uno de ellos de Job1 a Job5.
5. Conecte estos trabajos al flujo deseado al seleccionar el triángulo de condición en el objeto del trabajo y arrastrarlo hasta al siguiente trabajo. Por ejemplo, seleccione el triángulo de condición en

Job1 y arrástrelo hasta Job2. Esta acción convierte a Job1 en el predecesor de Job2 y, de forma predeterminada, Job1 debe completarse de manera correcta antes de que Job2 pueda ejecutarse.

En el siguiente diagrama, se muestra la vista de Control-M Planning del flujo de trabajo básico.

+

Q

+

+

✂

📄

🗑

↶

🔗

+

81%

Hierarchy

☰

☰

M2_Jobs

Job1

Job2

Job3

Job4

Job5

Summary

Edit

Notes

➔

Job type

AI M2JOB

Job name

Job1

Where

Control-M server

ip-10-0-3-77.us-west

Host/Host group

ip-10-0-3-77.us-west-2.compute.internal

What

Connection Profile

M2JOB

M2 Application Id

6f3juxz75vgibndemj6nkloqt4

Job Name

JOB1

Status Polling Frequency

5

Parent folder

M2_Jobs

Application

bankdemo

Sub application

M2

Created by

emuser

Scheduling

Schedule

Every day

Maximum reruns

99

Time zone

PST (GMT-08:00)

Prerequisites

➤ In Conditions

Name	Order Date	AND/OR	Remove
File_Transfer_Job_1...	Order date	AND	Yes

Actions

➤ Out Conditions

Validations (0)

Changes (0)

Notes (0)

Authorizations (0)

Order Reports (0)

⬆

Imagen proporcionada por cortesía de BMC Software, Inc. ©2022

Mediante JSON

El mismo flujo se puede codificar en JSON y tiene este aspecto:

```
{
  "Defaults": {
    "RunAs": "JOB",
    "Application": "bankdemo",
```

```

"SubApplication": "M2",
"Description": "Banking job flow run on AWS Mainframe Modernization",
"Host": "ip-10-0-3-77.us-west-2.compute.internal",
"ControlmServer": "ip-172-31-60-16.us-w",
"OrderMethod": "Manual",
"Job": {
  "actionIfError": {
    "Type": "If",
    "CompletionStatus": "NOTOK",
    "mailTeam": {
      "Type": "Mail",
      "Message": "Job %%JOBNAME failed",
      "Subject": "Error occurred",
      "To": "support@amazon.com"
    }
  }
},
"M2_Jobs": {
  "Type": "Folder",
  "JOB1": {
    "Type": "Job:ApplicationIntegrator:AI M2JOB",
    "ConnectionProfile": "JOB",
    "AI-M2 Application Id": "coczwktgb5ddxhfjkxelah7tzi",
    "AI-Job Name": "JOB1",
    "AI-Status Polling Frequency": "5"
  },
  "JOB2": {
    "Type": "Job:ApplicationIntegrator:AI M2JOB",
    "ConnectionProfile": "JOB",
    "AI-M2 Application Id": "coczwktgb5ddxhfjkxelah7tzi",
    "AI-Job Name": "SYNTAXERR",
    "AI-Status Polling Frequency": "5"
  },
  "JOB3": {
    "Type": "Job:ApplicationIntegrator:AI M2JOB",
    "ConnectionProfile": "JOB",
    "AI-M2 Application Id": "coczwktgb5ddxhfjkxelah7tzi",
    "AI-Job Name": "JOB3",
    "AI-Status Polling Frequency": "5"
  },
  "JOB4": {
    "Type": "Job:ApplicationIntegrator:AI M2JOB",

```

```

    "ConnectionProfile": "JOB",
    "AI-M2 Application Id": "coczwktgb5ddxhfjkxelah7tzi",
    "AI-Job Name": "JOB4",
    "AI-Status Polling Frequency": "5"
  },
  "JOB5": {
    "Type": "Job:ApplicationIntegrator:AI M2JOB",
    "ConnectionProfile": "JOB",
    "AI-M2 Application Id": "coczwktgb5ddxhfjkxelah7tzi",
    "AI-Job Name": "JOB5",
    "AI-Status Polling Frequency": "5"
  },
  "flow01": {
    "Type": "Flow",
    "Sequence": [
      "JOB1",
      "JOB2",
      "JOB3",
      "JOB5"
    ]
  },
  "flow02": {
    "Type": "Flow",
    "Sequence": [
      "JOB2",
      "JOB4",
      "JOB5"
    ]
  }
}

```

Utilice el servicio de implementación para implementar este flujo.

Automatizar la ejecución de los trabajos con una frecuencia programada

Con el flujo que creó en los pasos anteriores, puede agregar la programación básica y de tiempo de ejecución.

- La programación básica define los días en que se puede ejecutar un trabajo (por ejemplo, todos los días de la semana, solo los días laborables, al final del mes o al final del trimestre).

- La programación de tiempo de ejecución determina cuándo se ejecutará un trabajo en el día en que se puede ejecutar (por ejemplo, cada hora, una vez que se encuentren disponibles los recursos específicos o solo después de una confirmación manual).

Puede configurar la programación básica y de tiempo de ejecución en la pestaña de Programación.

El trabajo base se ejecuta en función de eventos

Control-M Managed File Transfer (MFT) es un cliente y servidor FTP/SFTP que puede utilizar para ver y transferir archivos entre un host local y uno remoto. Para obtener más información sobre la definición de un trabajo de File Transfer, consulte la [Documentación de Control-M](#).

En este piloto, se utiliza el trabajo de File Transfer para ver un evento de creación de archivos con la extensión .poc en la carpeta /bmcfile en un bucket de S3 con el nombre bmc-poc-bucket. Cuando se produce ese evento, se inicia el trabajo de Control-M para ejecutar el siguiente trabajo. Si lo desea, puede pasar la ruta completa, incluido el nombre del bucket.

Monitorear trabajos

Puede monitorear y verificar los trabajos en el dominio de Control-M Monitoring y en la consola de administración de AWS.

Control-M Monitoring

Los envíos y ejecuciones de trabajos se pueden monitorear en el Dominio de Control-M Monitoring. De forma predeterminada, los trabajos del servicio AWS Mainframe Modernization aparecerán junto con todos los demás trabajos de Control-M. Si desea ver solo los trabajos del servicio AWS Mainframe Modernization sin ninguna otra carga de trabajo (ni ningún otro requisito de filtrado), puede crear un Punto de vista.

Los puntos de vista muestran no solo la información del trabajo, sino que también las relaciones con las dependencias ascendentes y descendentes. Además, si su flujo de trabajo incluye AWS Mainframe Modernization y otros tipos de trabajos de Control-M, podrá ver y administrar todo el flujo en el dominio de Monitoring.

Puede seguir los pasos detallados al consultar la [Sección de puntos de vista de Monitoring](#) en la documentación de Control-M.

En la siguiente captura de pantalla, se muestran los resultados de dos flujos de trabajo. En el lado izquierdo, el flujo de trabajo se ha completado de forma correcta con los cinco trabajos en verde. En el lado derecho, el flujo de trabajo solo se ha completado de forma correcta en parte porque el Job2 devolvió el estado Error, y el flujo de trabajo se detuvo allí, dejando a Job3-Job5 con el estado Esperar programación.

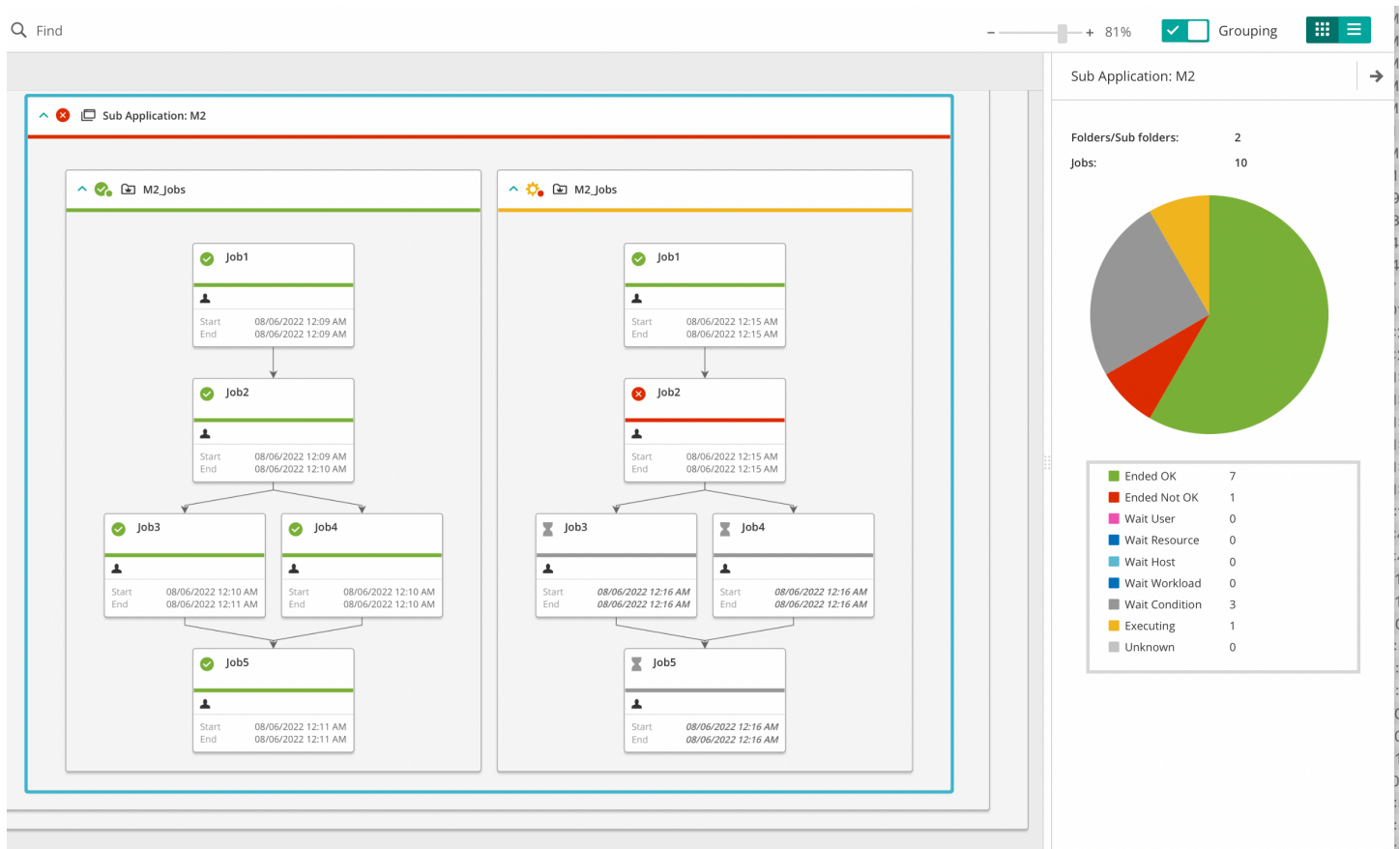


Imagen proporcionada por cortesía de BMC Software, Inc. ©2022

La consola de administración de AWS

También puede ver la información del trabajo y registro en la [Consola de AWS Mainframe Modernization](#). Esta vista no incluye las dependencias ni ninguna carga de trabajo que no se encuentre administrada por el servicio AWS Mainframe Modernization.

Prácticas recomendadas

Recomendamos que siga las siguientes prácticas recomendadas para las etapas iniciales de planificación e integración:

- En la fase de planificación inicial de la integración o migración, establezca una conectividad privada dedicada entre su red en las instalaciones y la región de AWS a la que se migrará su carga de trabajo. Esta es una práctica recomendada para una topología de red segura, pero no es obligatoria.
- Recomendamos utilizar perfiles de conexión centralizados porque este enfoque reduce la cantidad de objetos que se deben administrar y simplifica la implementación elástica de los agentes de Control-M.
- Cuando sea posible, realice la migración de la unidad central de forma gradual para reducir la complejidad y el riesgo. Al realizar una migración gradual, los equipos de migración pueden proporcionar comentarios con mayor rapidez sobre el progreso de la migración. Las empresas pueden utilizar esos comentarios para optimizar los procesos internos y acelerar el ritmo de la migración.
- A fin de evitar trabajo adicional, considere utilizar las plantillas que se proporcionaron para el tipo de trabajo y el perfil de conexión de las etapas iniciales.

Recursos relacionados

- [Micro Focus](#)
- [Documentación de BMC](#)
- [Control-M](#)
- [Prueba de Control-M](#)
- [Control-M Application Integrator](#)
- [Documentación de Control-M](#)
- [Modernización de la unidad central: DevOps en AWS con Micro Focus](#) (patrón de Recomendaciones de AWS)

Historial de documentos

En la siguiente tabla, se describen cambios significativos de esta guía. Si quiere recibir notificaciones de futuras actualizaciones, puede suscribirse a las [notificaciones RSS](#).

Cambio	Descripción	Fecha
Publicación inicial	—	16 de noviembre de 2022

Glosario de las Recomendaciones de AWS

Los siguientes son términos de uso común en las estrategias, guías y patrones que se ofrecen en las Recomendaciones de AWS. Para sugerir entradas, utilice el enlace [Enviar comentarios](#) al final del glosario.

Términos de DevOps

branch

Área contenida de un repositorio de código. La primera rama que se crea en un repositorio es la rama principal. Puede crear una rama nueva a partir de una rama existente y, a continuación, desarrollar características o corregir errores en la rama nueva. Una rama que se genera para crear una característica se denomina comúnmente rama de característica. Cuando la característica se encuentra lista para su lanzamiento, se vuelve a combinar la rama de característica con la rama principal. Para obtener más información, consulte [Acerca de las ramas](#) (documentación de GitHub).

repositorio de código

Una ubicación donde el código fuente y otros activos, como documentación, muestras y scripts, se almacenan y actualizan mediante procesos de control de versiones. Los repositorios en la nube más comunes incluyen GitHub o AWS CodeCommit. Cada versión del código se denomina rama. En una estructura de microservicios, cada repositorio se encuentra dedicado a una única funcionalidad. Una sola canalización de CI/CD puede utilizar varios repositorios.

integración y entrega continuas (CI/CD)

El proceso de automatización de las etapas de origen, compilación, prueba, presentación y producción del proceso de lanzamiento del software. La CI/CD se describe comúnmente como una canalización. La CI/CD puede ayudarlo a automatizar los procesos, mejorar la productividad, mejorar la calidad del código y entregar con mayor rapidez. Para obtener más información, consulte [Beneficios de la entrega continua](#). CD también puede significar implementación continua. Para obtener más información, consulte [Entrega continua frente a implementación continua](#).

implementación

El proceso de hacer que una aplicación, características nuevas o correcciones de código se encuentren disponibles en el entorno de destino. La implementación abarca implementar

cambios en una base de código y, a continuación, crear y ejecutar esa base en los entornos de la aplicación.

entorno de desarrollo

Consulte [environment](#).

environment

Una instancia de una aplicación en ejecución. Los siguientes son los tipos de entornos más comunes en la computación en la nube:

- entorno de desarrollo: instancia de una aplicación en ejecución que solo se encuentra disponible para el equipo principal responsable del mantenimiento de la aplicación. Los entornos de desarrollo se utilizan para probar los cambios antes de promocionarlos a los entornos superiores. Este tipo de entorno a veces se denomina entorno de prueba.
- entornos inferiores: todos los entornos de desarrollo de una aplicación, como los que se utilizan para las compilaciones y pruebas iniciales.
- entorno de producción: instancia de una aplicación en ejecución a la que pueden acceder los usuarios finales. En una canalización de CI/CD, el entorno de producción es el último entorno de implementación.
- entornos superiores: todos los entornos a los que pueden acceder usuarios que no sean del equipo de desarrollo principal. Esto puede incluir un entorno de producción, entornos de preproducción y entornos para las pruebas de aceptación por parte de los usuarios.

Flujo de trabajo de Gitflow

Un enfoque en el que los entornos inferiores y superiores utilizan diferentes ramas en un repositorio de código fuente. El flujo de trabajo de Gitflow se considera heredado, y [flujo de trabajo basado en enlaces troncales](#) es el enfoque moderno preferido.

rama de característica

Consulte [branch](#).

hotfix

Una solución urgente para un problema crítico en un entorno de producción. Debido a su urgencia, el hotfix suele realizarse fuera del flujo de trabajo típico de las versiones de DevOps.

infraestructura

Todos los recursos y activos que se encuentran en el entorno de una aplicación.

infraestructura como código (IaC)

Proceso de aprovisionamiento y administración de la infraestructura de una aplicación mediante un conjunto de archivos de configuración. La IaC se ha diseñado para ayudarlo a centralizar la administración de la infraestructura, estandarizar los recursos y escalar con rapidez a fin de que los entornos nuevos sean repetibles, fiables y consistentes. Para obtener más información, consulte [Infraestructura como código](#) (Documento técnico de AWS).

entornos inferiores

Consulte [environment](#).

rama principal

Consulte [branch](#).

entorno de producción

Consulte [environment](#).

versión

En un proceso de implementación, el acto de promover cambios en un entorno de producción.

entorno de prueba

Consulte [environment](#).

flujo de trabajo basado en enlaces troncales

Un enfoque en el que los desarrolladores crean y prueban características de forma local en una rama de característica y, a continuación, combinan esos cambios en la rama principal. Luego, la rama principal se adapta a los entornos de desarrollo, preproducción y producción, de forma secuencial.

entornos superiores

Consulte [environment](#).

control de versión

Procesos y herramientas que realizan un seguimiento de los cambios, como los cambios en el código fuente de un repositorio.

Términos de migración

Las 7 R

Siete estrategias de migración comunes para trasladar aplicaciones a la nube. Estas estrategias se basan en las 5 R que Gartner identificó en 2011 y consisten en lo siguiente:

- **Refactorizar/rediseñar:** traslade una aplicación y modifique su arquitectura mediante el máximo aprovechamiento de las características nativas en la nube para mejorar la agilidad, el rendimiento y la escalabilidad. Por lo general, esto implica trasladar el sistema operativo y la base de datos. Ejemplo: Migre la base de datos de Oracle en las instalaciones a Amazon Aurora PostgreSQL-Compatible Edition.
- **Redefinir la plataforma (transportar y redefinir):** traslade una aplicación a la nube e introduzca algún nivel de optimización para aprovechar las capacidades de la nube. Ejemplo: Migre la base de datos Oracle en las instalaciones a Amazon Relational Database Service (Amazon RDS) para Oracle en la nube de AWS.
- **Recomprar (readquirir):** cambie a un producto diferente, lo cual se suele llevar a cabo al pasar de una licencia tradicional a un modelo SaaS. Ejemplo: Migre el sistema de administración de las relaciones con los clientes (CRM) a Salesforce.com.
- **Volver a alojar (migrar mediante lift-and-shift):** traslade una aplicación a la nube sin realizar cambios para aprovechar las capacidades de la nube. Ejemplo: Migre su base de datos de Oracle en las instalaciones a Oracle en una instancia de EC2 en la nube de AWS.
- **Reubicar (migrar el hipervisor mediante lift and shift):** traslade la infraestructura a la nube sin comprar equipo nuevo, reescribir aplicaciones o modificar las operaciones actuales. Este escenario de migración es específico de VMware Cloud en AWS, que permite la compatibilidad de máquinas virtuales (VM) y la portabilidad de cargas de trabajo entre el entorno en las instalaciones y de AWS. Puede utilizar las tecnologías de VMware Cloud Foundation desde los centros de datos en las instalaciones al migrar una infraestructura a VMware Cloud en AWS. Ejemplo: Reubicar el hipervisor que aloja la base de datos de Oracle a VMware Cloud en AWS.
- **Retener (revisitar):** conserve las aplicaciones en el entorno de origen. Estas pueden incluir las aplicaciones que requieren una refactorización importante, que desee posponer para más adelante, y las aplicaciones heredadas que desee retener, ya que no hay ninguna justificación empresarial para migrarlas.
- **Retirar:** retire o elimine las aplicaciones que ya no sean necesarias en un entorno de origen.

migración activa-activa

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas (mediante una herramienta de replicación bidireccional o mediante operaciones de escritura doble) y ambas bases de datos gestionan las transacciones de las aplicaciones conectadas durante la migración. Este método permite la migración en lotes pequeños y controlados, en lugar de requerir una transición única. Es más flexible, pero requiere más trabajo que una [migración activa-pasiva](#).

migración activa-pasiva

Método de migración de bases de datos en el que las bases de datos de origen y destino se mantienen sincronizadas, pero solo la base de datos de origen gestiona las transacciones de las aplicaciones conectadas, mientras los datos se replican en la base de datos de destino. La base de datos de destino no acepta ninguna transacción durante la migración.

cartera de aplicaciones

Recopilación de información detallada sobre cada aplicación que utiliza una organización, incluido el costo de creación y mantenimiento de la aplicación y su valor empresarial. Esta información es clave para [el proceso de detección y análisis de la cartera](#) y ayuda a identificar y priorizar las aplicaciones que se van a migrar, modernizar y optimizar.

operaciones de inteligencia artificial (AIOps)

El proceso de utilizar técnicas de machine learning para resolver problemas operativos, reducir los incidentes operativos y la intervención humana, y mejorar la calidad del servicio. Para obtener más información sobre cómo se utiliza AIOps en la estrategia de migración de AWS, consulte la [Guía de integración de operaciones](#).

AWS Cloud Adoption Framework (AWS CAF)

Marco de directrices y prácticas recomendadas de AWS para ayudar a las organizaciones a desarrollar un plan eficiente y eficaz a fin de migrar con éxito a la nube. AWS CAF organiza la orientación en seis áreas de enfoque llamadas perspectivas: empresarial, humana, gobernanza, plataforma, seguridad y operaciones. Las perspectivas empresariales, humanas y de gobernanza se centran en las habilidades y los procesos empresariales; las perspectivas de plataforma, seguridad y operaciones se centran en las habilidades y los procesos técnicos. Por ejemplo, la perspectiva humana se dirige a las partes interesadas que se ocupan de los Recursos Humanos (RR. HH.), las funciones del personal y la administración de las personas. Desde esta perspectiva, AWS CAF brinda orientación para el desarrollo, la capacitación y la comunicación

de las personas, con el fin de ayudar a preparar la organización para una adopción exitosa de la nube. Para obtener más información, consulte la [Página web de AWS CAF](#) y el [Documento técnico de AWS CAF](#).

AWS Workload Qualification Framework (AWS WQF)

Herramienta que evalúa las cargas de trabajo de migración de bases de datos, recomienda estrategias de migración y brinda estimaciones de trabajo. AWS WQF se incluye con AWS Schema Conversion Tool (AWS SCT). Analiza los esquemas de bases de datos y los objetos de código, el código de las aplicaciones, las dependencias y las características de rendimiento y proporciona informes de evaluación.

planificación de la continuidad del negocio (BCP)

Plan que aborda el posible impacto de un evento disruptivo, como una migración a gran escala en las operaciones y permite a la empresa reanudar las operaciones rápidamente.

captura de datos de cambio (CDC)

Proceso de seguimiento de los cambios en un origen de datos, como una tabla de base de datos, y registro de los metadatos relacionados con el cambio. Puede utilizar los CDC para diversos fines, como auditar o replicar los cambios en un sistema de destino para mantener la sincronización.

Centro de excelencia en la nube (CCoE)

Equipo multidisciplinario que impulsa los esfuerzos de adopción de la nube en toda la organización, incluido el desarrollo de las prácticas recomendadas en la nube, la movilización de recursos, el establecimiento de plazos de migración y la dirección de la organización durante las transformaciones a gran escala. Para obtener más información, consulte las [Publicaciones del CCoE](#) en el Blog de estrategia empresarial en la nube de AWS.

etapas de adopción de la nube

Las siguientes son las cuatro fases por las que suelen pasar las organizaciones cuando migran a la nube de AWS:

- Proyecto: ejecución de algunos proyectos relacionados con la nube con fines de prueba de concepto y aprendizaje
- Fundamento: realización de inversiones fundamentales para escalar la adopción de la nube (p. ej., crear una zona de aterrizaje, definir un CCoE, establecer un modelo de operaciones)
- Migración: migración de aplicaciones individuales

- Reinención: optimización de productos y servicios e innovación en la nube

Stephen Orban definió estas etapas en la publicación del blog [The Journey Toward Cloud-First & the Stages of Adoption](#) (El camino hacia la nube como prioridad y las etapas de adopción) en el Blog de estrategia empresarial en la nube de AWS. Para obtener información sobre cómo se relacionan con la estrategia de migración de AWS, consulte la [Guía de preparación para la migración](#).

base de datos de administración de configuración (CMDB)

Repositorio que almacena y administra información sobre una base de datos y su entorno de TI, incluidos los componentes de hardware y software y sus configuraciones. Por lo general, los datos de una CMDB se utilizan en la etapa de detección y análisis de la cartera de productos durante la migración.

epopeya

En las metodologías ágiles, son categorías funcionales que ayudan a organizar y priorizar el trabajo. Las epopeyas brindan una descripción detallada de los requisitos y las tareas de implementación. Por ejemplo, las epopeyas de seguridad de AWS CAF incluyen la administración de identidades y accesos, los controles de detección, la seguridad de la infraestructura, la protección de datos y la respuesta a incidentes. Para obtener más información sobre las epopeyas en la estrategia de migración de AWS, consulte la [Guía de implementación del programa](#).

migración relámpago

Método de migración de bases de datos que utiliza la replicación continua de datos mediante [captura de datos de cambio \(CDC\)](#) para migrar los datos en el menor tiempo posible, en lugar de utilizar un enfoque gradual. El objetivo es reducir al mínimo el tiempo de inactividad.

migración heterogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que utilice un motor de base de datos diferente (por ejemplo, de Oracle a Amazon Aurora). La migración heterogénea suele ser parte de un esfuerzo de rediseño de la arquitectura y convertir el esquema puede ser una tarea compleja. [AWS ofrece AWS SCT](#), lo cual ayuda con las conversiones de esquemas.

migración homogénea de bases de datos

Migración de la base de datos de origen a una base de datos de destino que comparte el mismo motor de base de datos (por ejemplo, Microsoft SQL Server a Amazon RDS para SQL Server).

La migración homogénea suele formar parte de un esfuerzo para volver a alojar o redefinir la plataforma. Puede utilizar las utilidades de bases de datos nativas para migrar el esquema.

periodo de hiperatención

Periodo, inmediatamente después de la transición, durante el cual un equipo de migración administra y monitorea las aplicaciones migradas en la nube para solucionar cualquier problema. Por lo general, este periodo dura de 1 a 4 días. Al final del periodo de hiperatención, el equipo de migración suele transferir la responsabilidad de las aplicaciones al equipo de operaciones en la nube.

aplicación inactiva

Aplicación que utiliza un promedio de CPU y memoria de entre 5 y 20 por ciento durante un periodo de 90 días. En un proyecto de migración, es habitual retirar estas aplicaciones o mantenerlas en las instalaciones.

migración gradual

Estrategia de transición en la que se migra la aplicación en partes pequeñas en lugar de realizar una transición única y completa. Por ejemplo, puede trasladar inicialmente solo unos pocos microservicios o usuarios al nuevo sistema. Tras comprobar que todo funciona correctamente, puede trasladar microservicios o usuarios adicionales de forma gradual hasta que pueda retirar su sistema heredado. Esta estrategia reduce los riesgos asociados a las grandes migraciones.

biblioteca de información de TI (ITIL)

Conjunto de prácticas recomendadas para ofrecer servicios de TI y alinearlos con los requisitos empresariales. La ITIL proporciona la base para la ITSM.

administración de servicios de TI (ITSM)

Actividades asociadas con el diseño, la implementación, la administración y el soporte de los servicios de TI para una organización. Para obtener información sobre la integración de las operaciones en la nube con las herramientas de ITSM, consulte la [Guía de integración de operaciones](#).

zona de aterrizaje

Una zona de aterrizaje es un entorno de AWS correctamente diseñado, con varias cuentas, que es escalable y seguro. Este es un punto de partida desde el cual las organizaciones pueden lanzar e implementar rápidamente cargas de trabajo y aplicaciones con confianza en su entorno de seguridad e infraestructura. Para obtener más información sobre las zonas de aterrizaje, consulte [Configuración de un entorno de AWS seguro y escalable con varias cuentas](#).

migración grande

Migración de 300 servidores o más.

Programa de aceleración de la migración (MAP)

Programa de AWS que brinda soporte de consultoría, capacitación y servicios para ayudar a las organizaciones a construir una base operativa sólida para migrar a la nube y ayudar a compensar el costo inicial de las migraciones. El MAP incluye una metodología de migración para ejecutar las migraciones antiguas de forma metódica y un conjunto de herramientas para automatizar y acelerar los escenarios de migración más comunes.

Migration Portfolio Assessment (MPA)

Herramienta en línea que brinda información a fin de validar los argumentos comerciales necesarios para migrar a la nube de AWS. La MPA ofrece una evaluación detallada de la cartera (adecuación del tamaño de los servidores, precios, comparaciones del costo total de propiedad, análisis de los costos de migración), así como una planificación de la migración (análisis y recopilación de datos de aplicaciones, agrupación de aplicaciones, priorización de la migración y planificación de oleadas). La [herramienta MPA](#) (requiere inicio de sesión) está disponible de forma gratuita para todos los consultores de AWS y los consultores asociados de APN.

Evaluación de la preparación para la migración (MRA)

Proceso que consiste en obtener información sobre el estado de preparación de la nube de una organización, identificar los puntos fuertes y débiles, y elaborar un plan de acción para cerrar las brechas identificadas, mediante AWS CAF. Para obtener más información, consulte la [Guía de preparación para la migración](#). La MRA es la primera fase de la [estrategia de migración de AWS](#).

migración a escala

Proceso de transferencia de la mayoría de la cartera de aplicaciones a la nube en oleadas, con más aplicaciones desplazadas a un ritmo más rápido en cada oleada. En esta fase, se utilizan las prácticas recomendadas y las lecciones aprendidas en las fases anteriores para implementar una fábrica de migración de equipos, herramientas y procesos con el fin de agilizar la migración de las cargas de trabajo mediante la automatización y la entrega ágil. Esta es la tercera fase de la [estrategia de migración de AWS](#).

fábrica de migración

Equipos multifuncionales que agilizan la migración de las cargas de trabajo mediante enfoques automatizados y ágiles. Los equipos de la fábrica de migración suelen incluir operaciones,

analistas y propietarios de negocios, ingenieros de migración, desarrolladores y profesionales de DevOps que trabajan en tiempo y forma. Entre el 20 y el 50 por ciento de la cartera de aplicaciones empresariales se compone de patrones repetidos que pueden optimizarse mediante un enfoque de fábrica. Para obtener más información, consulte la [discusión sobre las fábricas de migración](#) y la [Guía de fábricas de migración a la nube](#) en este contenido.

metadatos de migración

Información sobre la aplicación y el servidor que se necesita para completar la migración. Cada patrón de migración requiere un conjunto diferente de metadatos de migración. Algunos ejemplos de metadatos de migración son las subredes de destino, los grupos de seguridad y las cuentas de AWS.

patrón de migración

Tarea de migración repetible que detalla la estrategia de migración, el destino de la migración y la aplicación o el servicio de migración utilizados. Ejemplo: Volver a alojar la migración en Amazon EC2 con AWS Application Migration Service.

estrategia de migración

Enfoque utilizado para migrar una carga de trabajo a la nube de AWS. Para obtener más información, consulte la entrada [Las 7 R](#) en este glosario y también [Movilizar la organización para acelerar las migraciones a gran escala](#).

migración fuera de línea

Método de migración en el que la carga de trabajo de origen se elimina durante el proceso de migración. Este método implica un tiempo de inactividad prolongado y, por lo general, se utiliza para cargas de trabajo pequeñas y no críticas.

migración en línea

Método de migración en el que la carga de trabajo de origen se copia al sistema de destino sin que se desconecte. Las aplicaciones que están conectadas a la carga de trabajo pueden seguir funcionando durante la migración. Este método implica un tiempo de inactividad nulo o mínimo y, por lo general, se utiliza para cargas de trabajo de producción críticas.

acuerdo de nivel operativo (OLA)

Acuerdo que aclara lo que los grupos de TI operativos se comprometen a ofrecerse entre sí, para respaldar un acuerdo de nivel de servicio (SLA).

integración de operaciones (OI)

Proceso de modernización de las operaciones en la nube, que implica la planificación de la preparación, la automatización y la integración. Para obtener más información, consulte la [Guía de integración de las operaciones](#).

administración del cambio organizacional (OCM)

Marco para administrar las transformaciones empresariales importantes y disruptivas desde la perspectiva de las personas, la cultura y el liderazgo. La OCM ayuda a las organizaciones a prepararse para nuevos sistemas y estrategias y a realizar la transición a ellos, al acelerar la adopción de cambios, abordar los problemas de transición e impulsar cambios culturales y organizacionales. En la estrategia de migración de AWS, este marco se denomina aceleración de personas, debido a la velocidad de cambio requerida en los proyectos de adopción de la nube. Para obtener más información, consulte la [Guía de OCM](#).

manual de estrategias

Conjunto de pasos predefinidos que capturan el trabajo asociado a las migraciones, como la entrega de las funciones de operaciones principales en la nube. Un manual puede adoptar la forma de scripts, manuales de procedimientos automatizados o resúmenes de los procesos o pasos necesarios para operar un entorno modernizado.

evaluación de cartera

Proceso de detección, análisis y priorización de la cartera de aplicaciones para planificar la migración. Para obtener más información, consulte la [Evaluación de la preparación para la migración](#).

matriz responsable, confiable, consultada e informada (RACI)

Matriz que define y asigna roles y responsabilidades en un proyecto. Por ejemplo, puede crear una RACI con el fin de definir la propiedad del control de seguridad o identificar las funciones y responsabilidades para tareas específicas de un proyecto de migración.

manual de procedimientos

Conjunto de procedimientos manuales o automatizados necesarios para realizar una tarea específica. Por lo general, se diseñan para agilizar las operaciones o los procedimientos repetitivos con altas tasas de error.

acuerdo de nivel de servicio (SLA)

Acuerdo que aclara lo que un equipo de TI se compromete a ofrecer a los clientes, como el tiempo de actividad y el rendimiento del servicio.

lista de tareas

Herramienta que se utiliza para hacer un seguimiento del progreso mediante un manual de procedimientos. La lista de tareas contiene una descripción general del manual de procedimientos y una lista de las tareas generales que deben completarse. Para cada tarea general, se incluye la cantidad estimada de tiempo necesario, el propietario y el progreso.

flujo de trabajo

Grupos funcionales de un proyecto de migración que son responsables de un conjunto específico de tareas. Cada flujo de trabajo es independiente, pero respalda a los demás flujos de trabajo del proyecto. Por ejemplo, el flujo de trabajo de la cartera es responsable de priorizar las aplicaciones, planificar las oleadas y recopilar los metadatos de migración. El flujo de trabajo de la cartera entrega estos recursos al flujo de trabajo de migración, que luego migra los servidores y las aplicaciones.

aplicación zombi

Aplicación que utiliza un promedio de CPU y memoria menor al 5 por ciento. En un proyecto de migración, es habitual retirar estas aplicaciones.

Términos de modernización

capacidad empresarial

Lo que hace una empresa para generar valor (por ejemplo, ventas, servicio al cliente o marketing). Las arquitecturas de microservicios y las decisiones de desarrollo pueden estar impulsadas por las capacidades empresariales. Para obtener más información, consulte la sección [Organizado en torno a las capacidades empresariales](#) del documento técnico [Ejecutar microservicios en contenedores en AWS](#).

diseño basado en el dominio

Un enfoque para desarrollar un sistema de software complejo mediante la conexión de sus componentes a dominios en evolución, o a los objetivos empresariales principales, a los que

sirve cada componente. Este concepto lo introdujo Eric Evans en su libro, *Diseño impulsado por el dominio: abordando la complejidad en el corazón del software* (Boston: Addison-Wesley Professional, 2003). Para obtener información sobre cómo utilizar el diseño basado en dominios con el patrón de higos estranguladores, consulte [Modernización gradual de los servicios web antiguos de Microsoft ASP.NET \(ASMX\) mediante contenedores y Amazon API Gateway](#).

microservicio

Un servicio pequeño e independiente que se comunica a través de API bien definidas y que, por lo general, es propiedad de equipos pequeños e independientes. Por ejemplo, un sistema de seguros puede incluir microservicios que se adapten a las capacidades empresariales, como las de ventas o marketing, o a subdominios, como las de compras, reclamaciones o análisis. Los beneficios de los microservicios incluyen la agilidad, la escalabilidad flexible, la facilidad de implementación, el código reutilizable y la resiliencia. Para obtener más información, consulte [Integración de microservicios mediante servicios sin servidor de AWS](#).

arquitectura de microservicios

Un enfoque para crear una aplicación con componentes independientes que ejecutan cada proceso de la aplicación como un microservicio. Estos microservicios se comunican a través de una interfaz bien definida mediante API ligeras. Cada microservicio de esta arquitectura se puede actualizar, implementar y escalar para satisfacer la demanda de funciones específicas de una aplicación. Para obtener más información, consulte [Implementación de microservicios en AWS](#).

modernización

Transformar una aplicación obsoleta (antigua o monolítica) y su infraestructura en un sistema ágil, elástico y de alta disponibilidad en la nube para reducir los costos, aumentar la eficiencia y aprovechar las innovaciones. Para obtener más información, consulte [Estrategia para modernizar las aplicaciones en la Nube de AWS](#).

evaluación de la preparación para la modernización

Evaluación que ayuda a determinar la preparación para la modernización de las aplicaciones de una organización; identifica los beneficios, los riesgos y las dependencias; y determina qué tan bien la organización puede soportar el estado futuro de esas aplicaciones. El resultado de la evaluación es un esquema de la arquitectura objetivo, una hoja de ruta que detalla las fases de desarrollo y los hitos del proceso de modernización y un plan de acción para abordar las brechas identificadas. Para obtener más información, consulte [Evaluación de la preparación para la modernización de las aplicaciones en la Nube de AWS](#).

aplicaciones monolíticas (monolitos)

Aplicaciones que se ejecutan como un único servicio con procesos estrechamente acoplados. Las aplicaciones monolíticas presentan varios inconvenientes. Si una característica de la aplicación experimenta un aumento en la demanda, se debe escalar toda la arquitectura. Agregar o mejorar las características de una aplicación monolítica también se vuelve más complejo a medida que crece la base de código. Para solucionar problemas con la aplicación, puede utilizar una arquitectura de microservicios. Para obtener más información, consulte [Descomposición de monolitos en microservicios](#).

persistencia polígloa

Elegir de forma independiente la tecnología de almacenamiento de datos de un microservicio en función de los patrones de acceso a los datos y otros requisitos. Si sus microservicios tienen la misma tecnología de almacenamiento de datos, pueden enfrentarse a desafíos de implementación o experimentar un rendimiento deficiente. Los microservicios se implementan más fácilmente y logran un mejor rendimiento y escalabilidad si utilizan el almacén de datos que mejor se adapte a sus necesidades. Para obtener más información, consulte [Habilitación de la persistencia de datos en los microservicios](#).

modelo de dividir y sembrar

Un patrón para escalar y acelerar los proyectos de modernización. A medida que se definen las nuevas funciones y los lanzamientos de los productos, el equipo principal se divide para crear nuevos equipos de productos. Esto ayuda a ampliar las capacidades y los servicios de su organización, mejora la productividad de los desarrolladores y apoya la innovación rápida. Para obtener más información, consulte [Enfoque gradual para modernizar las aplicaciones en la Nube de AWS](#).

patrón de higo estrangulador

Un enfoque para modernizar los sistemas monolíticos mediante la reescritura y el reemplazo gradual de las funciones del sistema hasta que se pueda dismantelar el sistema heredado. Este patrón utiliza la analogía de una higuera que crece hasta convertirse en un árbol estable y, finalmente, se apodera y reemplaza a su host. El patrón fue [presentado por Martin Fowler](#) como una forma de gestionar el riesgo al reescribir sistemas monolíticos. Para ver un ejemplo con la aplicación de este patrón, consulte [Modernización gradual de los servicios web antiguos de Microsoft ASP.NET \(ASMX\) mediante contenedores y Amazon API Gateway](#).

equipo de dos pizzas

Un pequeño equipo de DevOps al que puedes alimentar con dos pizzas. Un equipo formado por dos integrantes garantiza la mejor oportunidad posible de colaboración en el desarrollo de software. Para obtener más información, consulte la sección [El equipo de dos pizzas](#) del documento técnico [Introducción a DevOps en AWS](#).