AWS Mainframe Modernization과 Control-M 워크플로 오케스트레이터 통합 사용

AWS 권장 가이드

AWS 권장 가이드: AWS Mainframe Modernization과 Control-M 워크플로 오케스트레이터 통합 사용

Table of Contents

소개 1
개요 1
목표 결과 3
이 가이드의 용어 4
가정 4
아키텍처 6
Control-M과의 관리형 AWS Mainframe Modernization 통합 8
Control-M 리소스 배포 8
AWS Mainframe Modernization Control-M 플러그인 배포 8
Control-M Application Integrator에서 AWS Mainframe Modernization 작업 유형 배포 9
연결 프로파일 생성 9
작업 및 일정 생성 10
워크플로를 사용하여 작업 시작 12
일정에 따라 작업 자동화 15
이벤트에서 기본 작업 실행 16
작업 모니터링 16
Control-M Monitoring 16
콘솔에서 모니터링 17
AWS Mainframe Modernization Amazon EC2와 Control-M의 통합 18
리소스 배포 18
Control-M 에이전트 배포 19
Micro Focus 플러그인 배포 8
작업 유형 배포 19
연결 프로파일 생성 20
작업 및 일정 생성 21
워크플로를 사용하여 작업 시작 22
일정에 따라 작업 자동화 26
작업 모니터링 26
Control-M Monitoring 26
모범 사례 28
리소스 29
기여자 30
문서 기록 31
..... xxxii

와 Control-M 워크플로 오케스트레이터 통합 사용 AWS Mainframe Modernization

Amazon Web Services 및 BMC Software, Inc.(기여자)

2024년 2월(문서 기록)

비즈니스 및 디지털 트랜스포메이션의 요구 사항을 충족하기 위해 고객은 메인프레임 애플리케이션을 현대화하여 민첩성을 높이고 비용을 절감하며 혁신을 가속화하고 있습니다. re:Invent 2021에서 Amazon Web Services(AWS)는 고객이 메인프레임 워크로드를 현대화하는 데 도움이 되는 AWS Mainframe Modernization 서비스를 발표했습니다.는 클라우드 네이티브 관리형 고가용성 런타임 환경을 AWS Mainframe Modernization 제공합니다 AWS.

이 가이드에서는 BMC Helix Control-M 워크플로 오케스트레이션을 AWS Mainframe Modernization 서비스와 통합하여 메인프레임 현대화 여정 중에 배치 애플리케이션을 현대화할 수 있는 추가 옵션을 제공하는 방법에 대한 자세한 정보를 제공합니다. Control-M의 기능을 통해 조직은 엔드 투 엔드 뷰를 제공하는 단일 도구를 사용하여 기존 워크로드와 마이그레이션된 워크로드로 구성된 환경을 합리화할 수 있습니다.

개요

Control-M 및 AWS Mainframe Modernization Replatform을 Micro Focus와 통합하면 다음과 같은 이점이 있습니다.

- 인재 격차를 줄이는 데 도움이 됩니다.
- 민첩한 DevOps 접근 방식으로 빠른 혁신을 지원합니다.
- 큰 변경 없이 애플리케이션과 데이터에 더 쉽게 액세스할 수 있습니다.
- 애플리케이션 실행 또는 확장 비용을 최적화합니다.
- 비즈니스 민첩성을 극대화하고 비용을 절감합니다.

Micro Focus를 사용한 AWS 리플랫포밍 서비스는 메인프레임 애플리케이션을 AWS 클라우드 네이티브 관리형 런타임 환경으로 현대화하는 데 도움이 됩니다. 마이그레이션과 현대화를 계획하고 구현하는 데 도움이 되는 도구와 리소스를 제공합니다. 사용자는 배치 작업을 제출 또는 취소하고 배치 작업 실행의 세부 정보를 검토할 수 있습니다. 사용자가 배치 작업을 제출할 때마다 Micro Focus를 사용한 AWS 리플랫폼 서비스는 모니터링할 수 있는 별도의 배치 작업 실행을 생성합니다. AWS Mainframe

Modernization 서비스 웹 콘솔을 사용하여 이름별로 배치 작업을 검색하고, 작업 제어 언어(JCL), 스크립트 파일 및 파라미터를 배치 작업에 제공할 수 있습니다.

Control-M을 사용하면 애플리케이션 및 데이터 워크플로를 정의, 예약, 관리 및 모니터링할 수 있습니다. 이를 통해 가시성과 안정성이 향상되어 서비스 수준에 관한 계약(SLA)을 개선하는 데 도움이 됩니다. Control-M은 온프레미스 및에서 워크플로를 통합, 자동화 및 오케스트레이션 AWS 클라우드 하여 비즈니스 서비스를 적시에 제공할 수 있도록 지원합니다. 사용자는 단일 통합 그래픽 뷰를 통해 풍부한 플러그인 라이브러리를 사용하여 파일 전송, 애플리케이션, 데이터 소스 및 인프라를 포함한 모든 워크플로를 조정할 수 있습니다. 클라우드에 프로비저닝된 Control-M은의 임시 기능을 사용합니다 AWS 클라우드. Control-M 워크플로는 지속적 통합 및 지속적 전달(CI/CD) 도구 체인 내에서 REST API, JSON, Python을 사용하여 코드형 작업 방식을 사용하기 때문에 개발자와 DevOps 엔지니어가 공동으로 작업할 수 있도록 버전을 관리하고, 테스트하고, 유지 관리할 수 있습니다.

> **Note**
>
> AWS Mainframe Modernization AWS Blu Age를 사용한 리팩터링도 지원됩니다. 자세한 내용은 AWS 권장 가이드 가이드를 참조하세요.

목표 비즈니스 성과

이 솔루션은 다음 목표를 달성하는 데 도움이 됩니다.

비즈니스 이점

- 원활한 마이그레이션 및 워크플로 오케스트레이션 - Control-M을 사용하면 메인프레임 워크로드를 로 원활하게 전환할 수 있습니다 AWS. 복잡한 워크플로를 오케스트레이션하고 마이그레이션 중 및 마이그레이션 후 배치 처리의 연속성을 보장합니다.
- 하이브리드 운영 관리 - Control-M을 사용하면 로 전환하는 동안 레거시 메인프레임 프로세스와 새로운 클라우드 기반 애플리케이션을 효율적으로 관리할 수 있습니다 AWS.
- 비용 절감 및 리소스 사용 최적화 - Control-M AWS 을 사용하여 로 마이그레이션하면 최적화된 클라우드 리소스 사용 및 확장을 통해 운영 비용이 절감됩니다.
- 민첩성 및 혁신 향상 - 로 전환 AWS 하면 회사가 시장 변화에 빠르게 적응하고 혁신을 위해 클라우드 네이티브 서비스를 사용할 수 있습니다.
- 규정 준수, 보안 및 효율적인 모니터링 - Control-M은 지속적인 규정 준수 및 보안을 보장하는 데 도움이 됩니다. 또한 Control-M은 클라우드 환경에서 향상된 실시간 모니터링 및 보고 기능을 제공합니다.
- 인재 격차 감소 -는 레거시 플랫폼에서 숙련된 메인프레임 전문가를 AWS 찾고 유지하는 것과 관련된 문제를 해결합니다.

기술적 이점

이 솔루션은 다음과 같은 목표를 달성하는 데 도움이 됩니다.

- Control-M 환경을 확장하여 서비스의 워크로드를 AWS Mainframe Modernization 관리합니다.
- 마법사를 사용하여 Control-M REST APIs를 프로세스에 빠르게 통합하여 빠른 혁신을 지원합니다.
- Control-M 통합 뷰에서 모든 애플리케이션 워크로드 작업을 오케스트레이션, 예약 및 모니터링합니다.
- 변화하는 시장 상황과 고객 요구에 보다 효과적으로 대응할 수 있도록 조직의 확장성과 비즈니스 민첩성을 높입니다.
- 메인프레임 워크로드를 현대화하고 클라우드 네이티브 서비스의 이점을 활용하여 애플리케이션을 실행하거나 확장하는 비용을 최적화합니다.

이 가이드의 용어

- 메인프레임 애플리케이션은 일련의 비즈니스 프로세스를 수행하고 지원하는 일련의 관련 메인프레임 프로그램 및 하위 프로그램입니다. 메인프레임 애플리케이션은 배치 처리 시스템 또는 온라인 트랜잭션 처리(OLTP) 시스템일 수 있습니다.
- 배치 작업은 사용자 개입 없이 실행되도록 구성된 예약된 프로그램입니다. Micro Focus를 사용한 AWS 리플랫폼에서 배치 작업 JCL 파일과 배치 작업 바이너리는 모두 Amazon Simple Storage Service(Amazon S3) 버킷에서 준비되며, 두 파일의 위치는 애플리케이션 정의 파일에 제공됩니다.
- AWS Mainframe Modernization는 메인프레임 애플리케이션을 마이그레이션, 현대화, 실행 및 운영하기 위한 클라우드 네이티브 서비스입니다.
- Amazon S3는 높은 내구성, 가용성 및 성능을 갖춘 확장 가능한 객체 스토리지입니다.
- Amazon CloudWatch는 DevOps 엔지니어, 개발자, 사이트 신뢰성 엔지니어(SRE), IT 관리자 및 제품 소유자를 위해 구축된 모니터링 및 관찰성 서비스입니다.
- Control-M Web은 비즈니스 애플리케이션 워크플로의 구축, 테스트, 배포, 일정 수립 및 모니터링의 전체 수명 주기 동안 메인프레임 배치 작업을 비롯한 엔터프라이즈 워크로드를 관리하기 위한 솔루션입니다. 광범위한 그래픽 및 프로그래밍 기능을 통해 편안하게 Control-M 시설을 이용할 수 있습니다.
- Control-M의 Application Integrator는 Control-M과 애플리케이션 또는 클라우드 서비스 간의 통합을 지원하는 작업 유형을 생성하는 데 사용되는 웹 기반 로우코드 디자이너입니다. Application Integrator 작업 유형을 빌드한 후 다른 모든 Control-M 작업과 동일하게 작동하여 대상 통합을 위한 모든 Control-M 기능을 노출합니다. 이 가이드에서 통합은 Micro Focus를 사용하는 AWS Replatform과 함께 이루어집니다.
- Planning 도메인은 작업 흐름의 생성 및 업데이트를 관리하는 기능 세트입니다.
- Monitoring 도메인은 작업 흐름을 관리하는 기능 세트입니다. 여기에서 실행 중인 모든 작업과 해당 상태를 볼 수 있습니다. 오류가 발생하면 알림이 생성됩니다. 작업 상태를 확인하고, 지연 또는 장애에 대응하고, 문제를 분석하고, 수정 조치를 취하는 운영 작업을 수행할 수 있습니다.
- Control-M 에이전트는 Control-M Server에서 관리하는 다양한 컴퓨터에 설치됩니다. 호스트를 식별하는 에이전트 이름에 작업이 할당됩니다. Agent는 HTTPS 프로토콜(포트 443)을 사용하여 서버와 통신합니다.

가정

이 가이드의 예와 다이어그램은 다음과 같은 가정을 반영합니다.

- 마이그레이션되는 메인프레임 애플리케이션은 단일 프로그램 또는 여러 프로그램을 실행할 수 있습니다. 간소화를 위해 이 가이드의 다이어그램에는 각 애플리케이션에 대한 단일 프로그램과 여러 하위 프로그램이 표시됩니다.
- 메인프레임 애플리케이션은 여러 개의 정의된 배치 작업이 있는 Micro Focus 관리형 런타임 환경의 AWS 리플랫폼에서 마이그레이션되고 실행됩니다. 이 파일럿의 경우 자습서: Micro Focus용 관리형 런타임의 지침에 따라 Micro Focus를 사용하여 AWS Replatform에서 BankDemo 예제 애플리케이션을 설정합니다.
- 애플리케이션 플러그인 및 추가 기능을 포함한 모든 Control-M 구성 요소와 함께 전체 Control-M 설치가 가능합니다.
- Control-M Agent가 작업 관리를 담당합니다. 워크로드를 분산하기 위해 에이전트를 여러 컴퓨터에 설치할 수 있습니다. 이를 통해 성능과 복원력을 향상시킬 수 있습니다.
- Control-M Automation API는 Restful Web Services(REST API)를 통해 Control-M 기능을 노출합니다. 작업, 연결 프로파일, 사용자 및 역할, 사이트 표준 등의 아티팩트를 JSON으로 작성하고 다양한 서비스에 대한 입력으로 제공하거나 출력으로 생성할 수 있습니다. 서비스는 curl 또는 유사한 시설 또는 제공된 ctm 명령줄 인터페이스(CLI)를 사용하여 HTTPS 요청을 통해 직접 액세스할 수 있습니다.

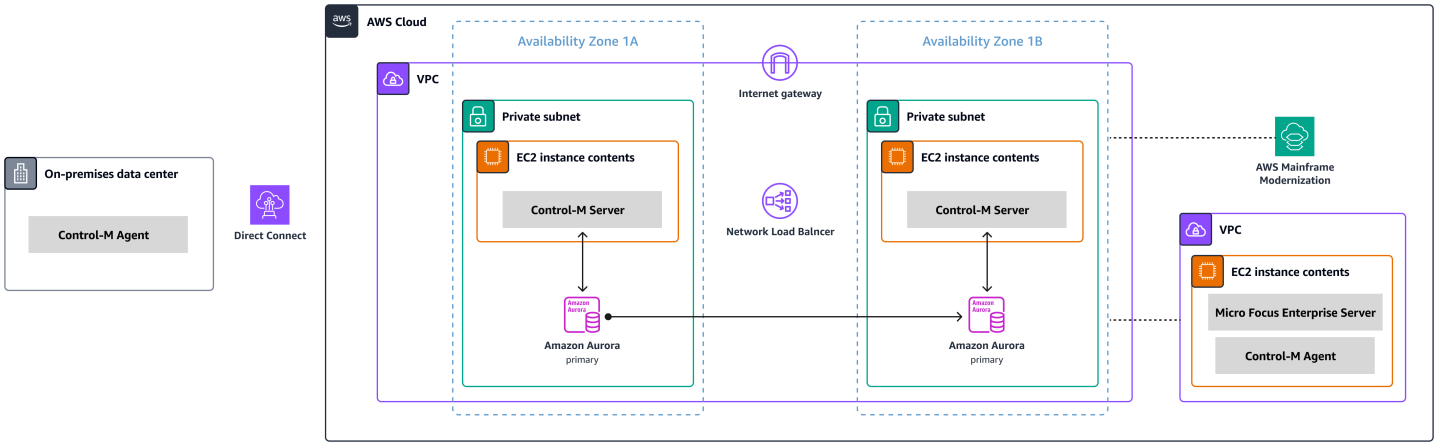
아키텍처

AWS Mainframe Modernization Micro Focus를 사용한 리플랫폼은 두 가지 모드로 제공됩니다.

- AWS Micro Focus를 사용한 리플랫폼은 Micro Focus 백엔드와 함께 동적으로 배포되고에서 완전히 관리하는 서버리스 관리형 런타임 환경입니다 AWS. AWS Micro Focus를 사용한 리플랫폼은 Micro Focus와 상호 작용하기 위한 클라우드 네이티브 API 계층을 제공합니다. 이 관리형 접근 방식에서는 Micro Focus만 리플랫포밍에 사용할 수 있습니다. UniKix 솔루션을 사용할 수 없습니다.
- AWS Micro Focus on Amazon Elastic Compute Cloud(Amazon EC2)를 사용하는 리플랫폼은 선택한 EC2 인스턴스 유형에서 시작하는 사전 설치된 Micro Focus 환경의 Amazon Machine Image(AMI)로 제공됩니다. 이 사용자 지정 배포는 네이티브 Micro Focus를 직접 노출합니다.

두 모드 모두 트랜잭션 관리자, 데이터 매핑 도구, 화면 및 맵 리더, 배치 작업 실행 환경을 포함합니다. 두 모드 중 하나를 사용하여 소스 코드를 최소한으로 변경하여 분산 서버에서 메인프레임 애플리케이션을 실행할 수 있습니다.

다음 다이어그램은 Control-M이 Amazon EC2 인스턴스에서 호스팅되는 워크플로 통합을 보여줍니다. Amazon Aurora 데이터베이스는 배치 작업을 관리하고 실행하는 데 필요한 데이터를 유지 관리하는 데 사용됩니다. 아키텍처는 고가용성을 위한 다중 가용 영역(Multi-AZ) 배포입니다. 애플리케이션의 배치 작업 및 데이터는 Micro Focus 런타임 환경의 AWS 리플랫폼에서 오케스트레이션됩니다. 다이어그램은 Amazon EC2에서 완전 관리형 및 사용자 지정이라는 Micro Focus 모드의 AWS 리플랫폼을 모두 보여줍니다.



다이어그램은 다음 리소스를 보여줍니다.

1. 온프레미스 환경에서 Control-M Agent는 IBM Z/OS 또는 기타 워크로드에서 계속 실행되는 워크로드를 제어하도록 설치됩니다. x86에서 실행되는 워크로드는 AWS Direct Connect를 통해 AWS 환경에 연결됩니다.
2. Control-M Server는 고가용성 및 재해 복구를 위해 다중 AZ 환경의 액티브-패시브 모드에서 EC2 인스턴스 페어에 설치됩니다.
3. Control-M에서 사용하는 Amazon Aurora 백엔드 데이터베이스(EC2 인스턴스에서 실행)는 고가용성 및 재해 복구를 위해 보조 가용 영역에 복제본과 함께 배포됩니다.
4. 별도의 VPC에는 Micro Focus가 사전 설치된 Micro Focus 환경의 AMI로 제공되는 AWS Replatform이 있는 EC2 인스턴스가 포함되어 있습니다. Control-M Agent는 확장 작업 관리 기능을 제공하는 Micro Focus 유틸리티와 상호 작용하기 위해이 인스턴스에 설치됩니다.

마이그레이션 프로젝트 중에도 메인프레임 및 분산 서버 모두에서 AWS 위치가 아닌 워크로드를 관리할 수 있습니다. 표시된 아키텍처는 규범적이 아니라 일반적인 방향을 제공하기 위한 것입니다. 재해 복구 옵션을 포함한 자세한 구성은 Control-M 구현의 일부로 구성하는 것이 좋습니다.

Control-M과의 관리형 AWS Mainframe Modernization 통합

이 섹션에서는 Control-M이 Micro Focus 런타임 엔진으로 배포된 관리형 AWS Mainframe Modernization 환경에서 실행되는 배치 작업과 통합되고 지원하는 방법을 설명합니다. Amazon EC2에서 Micro Focus 환경을 사용하는 사용자 지정 AWS Replatform을 구현하는 경우 AWS Mainframe Modernization Amazon EC2와 Control-M의 통합 섹션을 참조하세요.

이 섹션에서는 다음 사전 조건을 가정합니다.

- 활성. AWS 계정
- 메인프레임 애플리케이션은 여러 개의 정의된 배치 작업이 있는 Micro Focus 관리형 런타임 환경의 AWS 리플랫폼에서 마이그레이션되고 실행됩니다.
- 이 파일럿의 경우 BankDemo 예제 애플리케이션이에 설정되어 있습니다 AWS Mainframe Modernization. 설정 지침은 자습서: Micro Focus용 관리형 런타임을 참조하세요.

다음 주제에서는 Control-M 스케줄러와 다양한 유형의 통합 워크플로 AWS Mainframe Modernization 환경 간의 통합에 필요한 step-by-step 설정을 설명합니다.

- Control-M 리소스 배포
- AWS Mainframe Modernization을 위한 Control-M 연결 프로파일 생성
- Control-M Planning에서 작업 및 일정 생성
- 작업 모니터링

Control-M 리소스 배포

AWS Mainframe Modernization을 Control-M과 통합하는 첫 두 단계는 플러그인과 작업 유형을 배포하는 것입니다.

AWS Mainframe Modernization Control-M 플러그인 배포

플러그인은 Control-M이 오케스트레이션하는 애플리케이션 및 서비스에 대한 통합 및 지원을 제공합니다. 관리형 AWS Mainframe Modernization 서비스의 경우 플러그인을 배포합니다 AWS Mainframe Modernization .

플러그인 배포는 자주 발생하지 않는 활동입니다. 플러그인을 처음 설치하는 경우 Control-M 설명서의 단계를 따릅니다. 사용하려는 기존 플러그인이 있는 경우이 단계를 건너뛰고 연결 프로필을 생성합니다.

Control-M Application Integrator에서 AWS Mainframe Modernization 작업 유형 배포

작업 유형 배포는 일반적으로 일회성 활동입니다. 사용하려는 기존 작업 유형이 이미 있는 경우 이 단계를 건너뛰고 연결 프로파일을 생성하는 다음 단계로 이동합니다.

예제 작업 유형 AIJOB.ctmai는 git 리포지토리에 제공됩니다. 작업 유형을 배포하려면 Application Integrator를 사용하여 다음 단계를 수행해야 합니다.

1. aws-mainframe-modernization-controlm-integration GitHub 리포지토리를 복제하고 Application Integrator에서 액세스할 수 있는 파일 시스템 위치에 `AIJOB.ctmai` 파일을 다운로드합니다.
2. Application Integrator에 로그인합니다.
3. Home 탭에서 Import job type from file을 선택하고 `AIJOB.ctmai`의 위치를 선택합니다.
4. 제공된 예제를 수정하려면 Application Integrator를 숙지해야 합니다.
5. Control-M 설명서의 지침에 따라 작업 유형을 배포합니다.

에 대한 Control-M 연결 프로필 생성 AWS Mainframe Modernization

연결 프로파일은 애플리케이션의 특정 인스턴스에 대한 연결 속성과 보안 인증을 정의합니다. 여러 작업에서 각 연결 프로파일을 참조할 수 있습니다. 애플리케이션과 보안 인증의 고유한 조합마다 별도의 프로파일을 가질 수 있습니다.

연결 프로파일을 정의하려면 Control-M 웹 인터페이스의 구성 도메인에서 사용할 수 있는 그래픽 사용자 인터페이스(GUI)를 사용하거나 JSON을 사용할 수 있습니다. AWS Mainframe Modernization 플러그인의 연결 프로파일에 대한 자세한 내용은 Control-M 설명서를 참조하세요.

다음 코드는 JSON 사용의 예입니다.

```
{
  "MANAGED-M2-REPLATFORM": {
    "Type": "ConnectionProfile:AWS Mainframe Modernization",
    "Mainframe Modernization URL": "https://m2.{{AwsRegion}}.amazonaws.com",
    "Connection Timeout": "30",
    "AWS Region": "us-west-2",
    "Authentication": "NoSecret",
```

```
    "IAM Role": "--- IAM Role name ---",
    "AWS Logs URL": "https://logs.{{AwsRegion}}.amazonaws.com",
    "Description": "",
    "Centralized": true
  }
}
```

예제와 유사한 JSON 파일을 생성하고 Control-M Automation API 배포 서비스를 사용하여 배포합니다. 예를 들어 JSON 코드가 라는 파일에 저장되는 경우이 연결 프로파일을 배포하는 `cp-MANAGED-M2-REPLATFORM.json` `ctm` CLI 구문은 다음과 같습니다.

```
ctm deploy cp-MANAGED-M2-REPLATFORM.json
```

Control-M Automation API의 응답은 다음과 유사합니다.

```
[
  {
    "deploymentFile": "cp-Managed-M2-REPLATFORM.json",
    "deploymentState": "DEPLOYED_CONNECTION_PROFILES",
    "deploymentStatus": "ENDED_OK",
    "successfulFoldersCount": 0,
    "successfulSmartFoldersCount": 0,
    "successfulSubFoldersCount": 0,
    "successfulJobsCount": 0,
    "successfulConnectionProfilesCount": 1,
    "successfulDriversCount": 0,
    "isDeployDescriptorValid": false,
    "deployedConnectionProfiles": [
      "MANAGED-M2-REPLATFORM"
    ]
  }
]
```

Control-M Planning에서 작업 및 일정 생성

이제 작업 유형이 배포되고 AWS 연결을 위한 연결 프로파일이 있으므로 작업 생성 및 실행을 시작할 수 있습니다.

각 AWS Mainframe Modernization 서비스 작업은 4개의 섹션으로 구성된 속성 세트로 구성됩니다. 각 섹션에는 많은 속성이 있을 수 있습니다. 다음 목록은 가장 일반적으로 사용되는 속성 중 일부를 보여줍니다.

- 일반
  - 작업 이름
  - 작업이 속한 애플리케이션 및 하위 애플리케이션
  - 제출할 JCL
  - 설명서 링크
- 일정:
  - 이 작업을 실행할 수 있는 월 및 일
  - 알고리즘으로 정의할 수 없는 비즈니스 회계 기간, 공휴일 또는 기타 특별한 날짜와 같은 달력
  - 시간 범위
  - 주기적 행동
- 사전 조건:
  - 업스트림 종속성(일반적으로 이 작업을 실행할 수 있게 되기 전에 성공적으로 완료되어야 하는 작업)
  - 필요할 수 있는 리소스
  - 필요할 수 있는 사용자 작업
- 작업 완료 시 Control-M에서 수행하는 작업:
  - 작업 성공 또는 실패 결정(일반적으로 작업의 완료 코드를 기반으로 하지만 해당 설정을 재정의하여 출력 텍스트 또는 특정 상태 확인을 사용할 수 있음)
  - 실패 또는 성공에 대한 알림(예: 이메일)
  - 다운스트림 종속성에 대한 게시 상태

연결 프로파일과 마찬가지로 작업을 GUI 또는 JSON에서 생성 및 배포하고 Control-M Automation API를 사용하여 배포할 수 있습니다.

다음 섹션에서는 몇 가지 일반적인 워크플로 시나리오를 살펴봅니다.

- 이전 작업의 상태 코드를 기반으로 작업 시작
- 예약된 빈도로 작업 실행 자동화
- 이벤트에서 기본 작업 실행

이전 작업의 상태를 기반으로 작업 시작

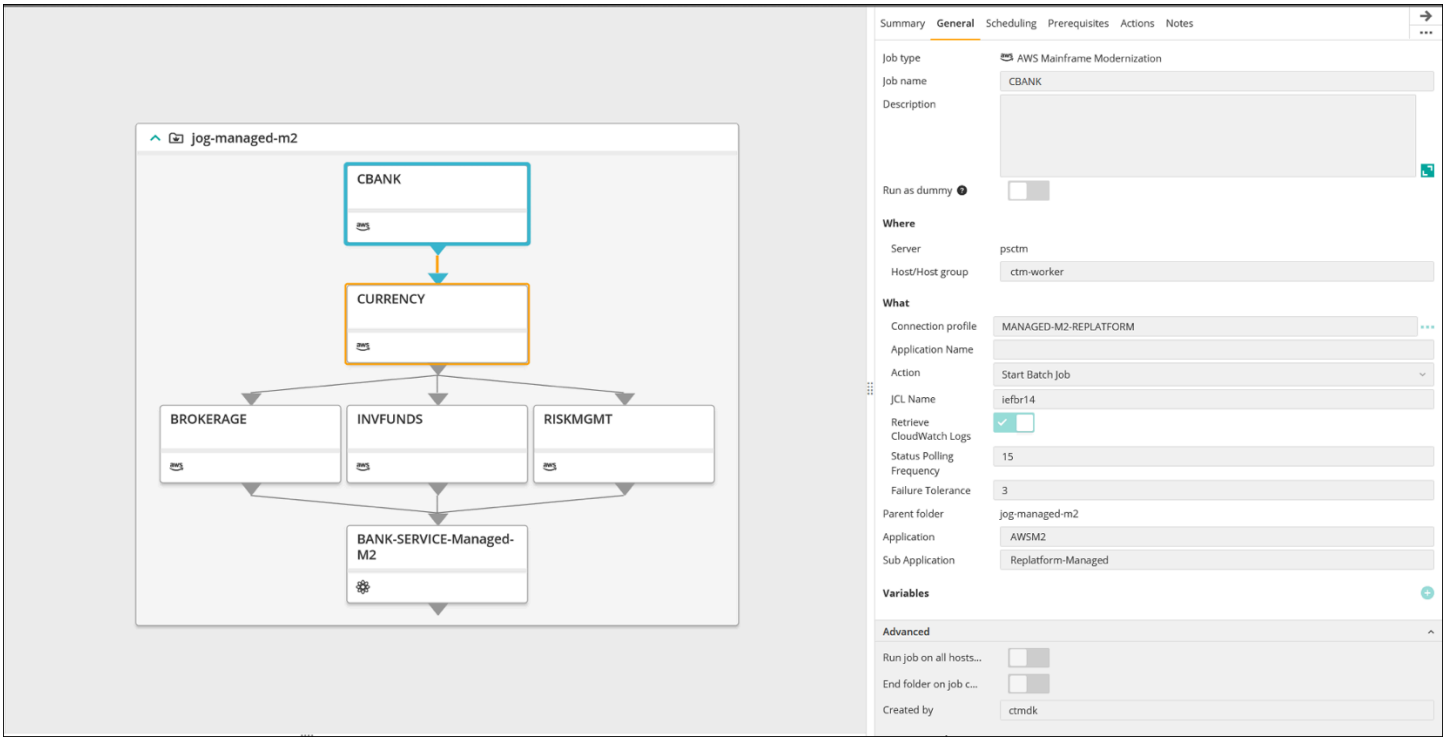
워크플로라는 작업 흐름을 생성합니다. 워크플로의 작업은 이전 작업이 성공적으로 완료되면 종속성과 상호 연결됩니다.

Control-M 웹 GUI 사용

Control-M 사용자 인터페이스에서 작업을 시작하려면 다음을 수행합니다.

1. 계획 도메인에 새 Workspace를 추가합니다. 그러면 빈 폴더 객체가 있는 캔버스가 열립니다.
2. AWS Mainframe Modernization 작업 유형(제공된 작업 유형 템플릿을 사용하는 경우 이를 M2JOB라고 함)을 선택하고 폴더로 끕니다.
3. 작업 유형의 색상이 녹색으로 바뀌면 삭제합니다. 오른쪽 창에는 일반, 예약, 사전 조건 및 작업 섹션이 포함되어 있습니다. 작업을 생성하려면 Control-M 설명서의 표준 지침을 참조하세요.
4. 다음으로 작업 유형을 구성하려면 작업 이름 값이 필요합니다. 콘솔의 AWS Mainframe Modernization 애플리케이션 정의 화면에서 또는 ListBatchJobDefinitions API를 실행하여 작업 이름 값을 찾을 수 있습니다. 이 파일럿의 경우 여러 작업에 대해 2~4단계를 반복하여 각 작업의 이름을 선택한 이름으로 지정합니다. 예제 이름은 `CBANK`, `CURRENCY`, `INVFUNDS`, `BROKERAGE`, 및 `RISKMGMT`입니다`BANK-SERVICE-Managed-M2`.
5. 이러한 작업을 원하는 흐름에 연결하려면 작업 객체에서 조건 삼각형을 선택하고 다음 작업으로 끕니다. 예를 들어, 아래에서 조건 삼각형을 선택하고 로 `CBANK`드래그합니다`CURRENCY`. 이 작업은 `CBANK`의 이전 버전을 만듭니다`CURRENCY`. 기본적으로 `CURRENCY`가 실행 가능 상태가 되기 전에 가 성공적으로 완료되어야 `CBANK` 합니다.

다음 스크린샷은 기본 작업 흐름의 Control-M 계획 보기를 보여줍니다.



이미지 제공: BMC Software, Inc. ©2022

JSON 사용

동일한 흐름을 JSON으로 코딩할 수 있습니다.

```
{
  "Defaults": {
    "Application": "AWSM2",
    "SubApplication": "Replatform-Managed",
    "Job": {
      "Host": "ctm-worker",
      "Output": {}
    }
  },
  "jog-managed-m2": {
    "Type": "Folder",
    "ControlmServer": "psctm",
    "OrderMethod": "Manual",
    "SiteStandard": "_z_DemoBusinessFlows",
    "CBANK": {
      "Type": "Job:AWS Mainframe Modernization",
      "ConnectionProfile": "MANAGED-M2-REPLATFORM",
      "JCL Name": "iefbr14",
```

```
      "Retrieve CloudWatch Logs": "checked",
      "Action": "Start Batch Job",
      "Application Version": "1"
    },
    "CURRENCY": {
      "Type": "Job:AWS Mainframe Modernization",
      "ConnectionProfile": "MANAGED-M2-REPLATFORM",
      "JCL Name": "iefbr14",
      "Retrieve CloudWatch Logs": "checked",
      "Action": "Start Batch Job",
      "Application Version": "1"
    },
    "BROKERAGE": {
      "Type": "Job:AWS Mainframe Modernization",
      "ConnectionProfile": "MANAGED-M2-REPLATFORM",
      "JCL Name": "iefbr14",
      "Retrieve CloudWatch Logs": "checked",
      "Action": "Start Batch Job",
      "Application Version": "1"
    },
    "INVFUNDS": {
      "Type": "Job:AWS Mainframe Modernization",
      "ConnectionProfile": "MANAGED-M2-REPLATFORM",
      "JCL Name": "iefbr14",
      "Retrieve CloudWatch Logs": "checked",
      "Action": "Start Batch Job",
      "Application Version": "1"
    },
    "RISKMGMT": {
      "Type": "Job:AWS Mainframe Modernization",
      "ConnectionProfile": "MANAGED-M2-REPLATFORM",
      "JCL Name": "iefbr14",
      "Retrieve CloudWatch Logs": "checked",
      "Action": "Start Batch Job",
      "Application Version": "1"
    },
    "BANK-SERVICE-Managed-M2": {
      "Type": "Job:SLAManagement",
      "ServiceName": "Bank Service - Managed M2",
      "RunAs": "ctmagent",
      "CompleteBy": {
        "Time": "12:00",
        "Days": "0"
      }
```

```
    },
    "leftbranch": {
      "Type": "Flow",
      "Sequence": [
        "CURRENCY",
        "RISKMGMT",
        "BANK-SERVICE-Managed-M2"
      ]
    },
    "middlebranch": {
      "Type": "Flow",
      "Sequence": [
        "CBANK",
        "CURRENCY",
        "INVFUNDS",
        "BANK-SERVICE-Managed-M2"
      ]
    },
    "rightbranch": {
      "Type": "Flow",
      "Sequence": [
        "CURRENCY",
        "BROKERAGE",
        "BANK-SERVICE-Managed-M2"
      ]
    }
  }
}
```

이 흐름을 배포하려면 배포 서비스를 사용합니다.

```
ctm deploy folder-MANAGED-M2-REPLATFORM.json
```

예약된 빈도로 작업 실행 자동화

이전 단계에서 생성한 흐름을 사용하여 기본 일정 예약과 런타임 일정 예약을 추가할 수 있습니다.

- 기본 일정 예약은 작업을 실행할 수 있는 날(예: 매주 평일, 근무일, 월말 또는 분기말에만)를 정의합니다.
- 런타임 일정은 실행이 가능한 날(예: 매시간, 지정된 리소스를 사용할 수 있는 후 또는 수동 확인 후에만)에 작업이 실행되는 시기를 결정합니다.

일정 예약 탭에서 기본 및 런타임 일정 예약을 설정할 수 있습니다.

이벤트에서 기본 작업 실행

Control-M Managed File Transfer(MFT)는 로컬 호스트와 원격 호스트 간에 파일을 감시하고 전송하는 데 사용할 수 있는 FTP/SFTP 클라이언트 및 서버입니다. File Transfer 작업 정의에 대한 자세한 내용은 Control-M 설명서를 참조하세요.

이 파일럿에서는 File Transfer 작업을 사용하여 `bmc-poc-bucket`이라는 S3 버킷의 `/bmcfile` 폴더에서 확장명이 .poc인 파일의 파일 생성 이벤트를 감시합니다. 해당 이벤트가 발생하면 다음 작업을 실행하기 위해 Control-M 작업이 시작됩니다. 선택적으로 버킷 이름을 포함한 전체 경로를 전달할 수 있습니다.

작업 모니터링

Control-M 모니터링 도메인 내에서 그리고를 통해 작업 처리를 모니터링하고 검증하여 두 플랫폼 모두에서 포괄적인 관찰 및 확인을 AWS Management Console보장할 수 있습니다.

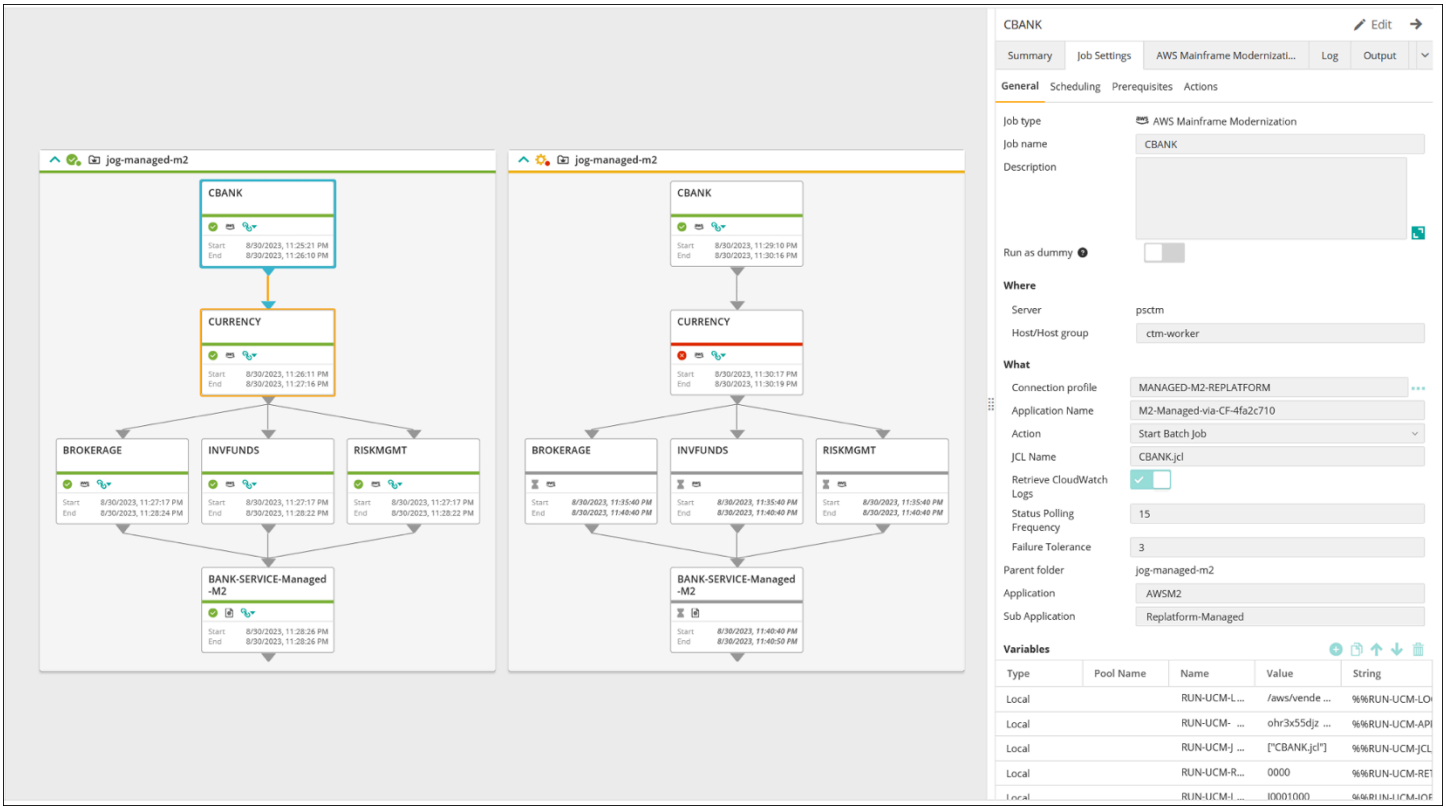
Control-M Monitoring

Control-M Monitoring 영역에서 작업 제출 및 실행을 모니터링할 수 있습니다. 기본적으로 AWS Mainframe Modernization 서비스 작업은 다른 모든 Control-M 작업과 함께 표시됩니다. 다른 워크로드(또는 다른 필터링 요구 사항) 없이 AWS Mainframe Modernization 서비스 작업만 보려면 뷰포인트를 생성할 수 있습니다.

Viewpoint는 작업 정보뿐만 아니라 업스트림 및 다운스트림 종속성과의 관계도 보여줍니다. 또한 워크플로에 AWS Mainframe Modernization 및 기타 Control-M 작업 유형이 포함된 경우 모니터링 도메인에서 전체 흐름을 보고 관리할 수 있습니다.

자세한 단계를 수행하려면 Control-M 설명서의 모니터링의 뷰포인트 섹션을 참조하세요.

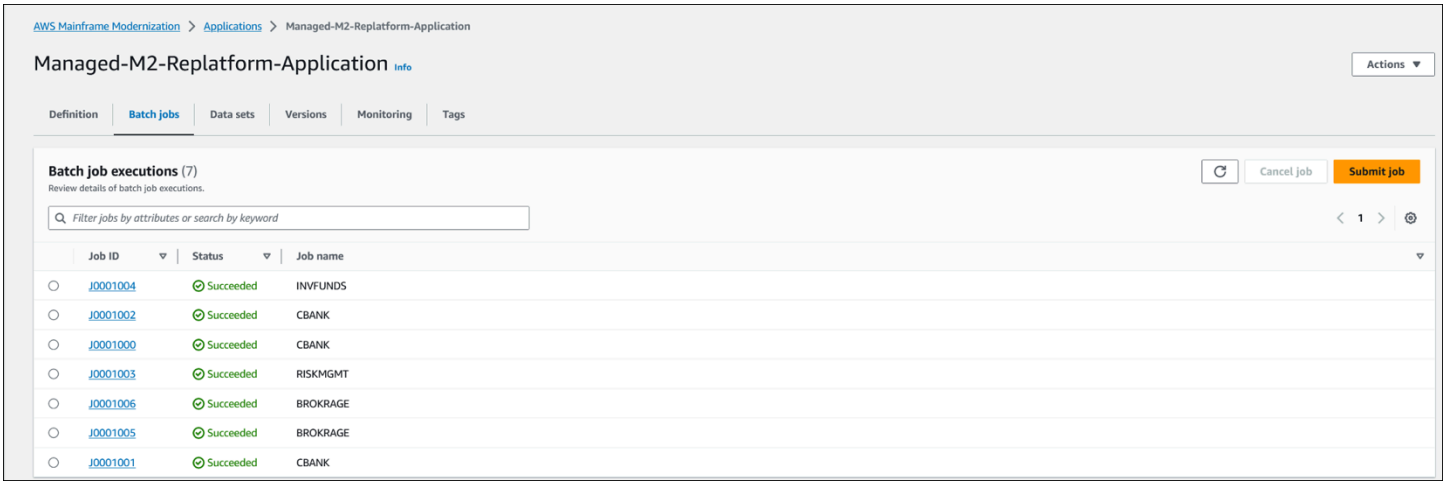
다음 스크린샷은 두 워크플로의 결과를 보여줍니다. 왼쪽에서 워크플로는 모든 작업을 녹색으로 표시하여 성공적으로 완료됩니다. 오른쪽에서는 작업이 빨간색으로 표시된 실패 상태를 `CURRRENCY` 반환했기 때문에 워크플로가 부분적으로만 성공합니다. 워크플로가 중지되어 나머지 작업은 대기 일정 상태로 유지됩니다.



이미지 제공: BMC Software, Inc. ©2022

콘솔에서 모니터링

에서 작업 및 로그 정보를 보려면에 AWS로그인한 AWS Management Console다음 AWS Mainframe Modernization 콘솔로 이동합니다.



이 보기에는 종속성이나 AWS Mainframe Modernization 서비스에서 관리하지 않는 워크로드가 포함되지 않습니다.

AWS Mainframe Modernization Amazon EC2와 Control-M의 통합

이 섹션에서는 Control-M이 EC2 인스턴스에 배포된 사용자 지정 AWS Mainframe Modernization 런타임 환경에서 실행되는 배치 작업과 통합되고 지원하는 방법을 설명합니다. Micro Focus 런타임 환경을 사용하여 완전 관리형 AWS Replatform을 구현하는 경우 Control-M과의 관리형 AWS Mainframe Modernization 통합 섹션을 참조하세요.

이 섹션에서는 다음 사전 조건을 가정합니다.

- 활성 상태의 AWS 계정.
- EC2 인스턴스가 생성될 Virtual Private Cloud(VPC)입니다.
- 메인프레임 애플리케이션은 EC2 인스턴스의 Micro Focus 환경이 있는 AWS Replatform에서 마이그레이션되고 실행되며 여러 개의 정의된 배치 작업으로 Micro Focus 런타임 엔진을 지원합니다. 이 파일럿의 경우 Micro Focus를 사용하여 애플리케이션 리플랫포밍의 지침을 따릅니다. 이 설명서에는 Amazon EC2에서 Micro Focus 런타임 환경을 사용하여 AWS Replatform을 구성하고 운영하는 방법에 대한 모든 작업과 추가 정보가 포함되어 있습니다.

다음 주제에서는 Control-M과 AWS Replatform with Micro Focus 환경 간의 통합에 필요한 설정을 다룹니다.

- Control-M 및 Micro Focus 리소스 배포
- Control-M 연결 프로필 생성
- Control-M Planning에서 작업 및 일정 생성
- 모니터링을 사용하여 Control-M에서 작업 실행 관리

Control-M 및 Micro Focus 리소스 배포

AWS Mainframe Modernization을 Control-M과 통합하는 첫 두 단계는 Control-M 에이전트를 배포한 다음 에이전트에 플러그인을 배포하는 것입니다. 세 번째 단계는 Control-M Application Integrator에 작업 유형을 배포하는 것입니다.

EC2 인스턴스에 Control-M Agent 배포

Amazon EC2 런타임 환경에서 Micro Focus와 함께 사용자 지정 AWS Replatform을 사용하는 경우 `MFBSIFJCL` 유틸리티가 호출됩니다. 유틸리티는의 Micro Focus Enterprise Server 제안을 사용하여 시작된 호스트에서 작동합니다 AWS Marketplace. `MFBSIFJCL` 유틸리티를 실행하려면 해당 호스트에 Control-M 에이전트도 배포해야 합니다. 지침은 Control-M 설명서를 참조하세요.

> **Note**
>
> BMC 전자 제품 다운로드 사이트에서 필요한 설치 미디어를 다운로드할 수 있습니다.

Control-M Agent에 Micro Focus 플러그인 배포

플러그인은 Control-M이 오케스트레이션하는 애플리케이션 및 서비스에 대한 통합 및 지원을 제공합니다.

플러그인 배포는 자주 발생하지 않는 활동입니다. 사용하려는 기존 플러그인이 이미 있는 경우이 단계를 건너뛰고 연결 프로파일을 생성합니다.

Amazon EC2에서 Micro Focus 서비스가 포함된 AWS Mainframe Modernization 리플랫폼은 Micro Focus 엔진을 노출합니다. 통합을 위해 AWS Mainframe Modernization Control-M은 Micro Focus 플러그인을 사용합니다. 자세한 내용은 Control-M 설명서를 참조하세요.

Micro Focus Enterprise Server가 실행 중인 호스트에 설치된 에이전트에이 플러그인을 배포해야 합니다.

Control-M Application Integrator에서 AWS Mainframe Modernization 작업 유형 배포

작업 유형 배포는 일반적으로 일회성 활동입니다. 사용하려는 기존 작업 유형이 이미 있는 경우 이 단계를 건너뛰고 연결 프로파일을 생성하는 다음 단계로 이동합니다.

예제 작업 유형 AIJOB.ctmai는 git 리포지토리에 제공됩니다. 작업 유형을 배포하려면 Application Integrator를 사용하여 다음 단계를 수행해야 합니다.

- aws-mainframe-modernization-controlm-integration GitHub 리포지토리를 복제하고 Application Integrator에서 액세스할 수 있는 파일 시스템 위치에 `AIJOB.ctmai` 파일을 다운로드합니다.
- Application Integrator에 로그인합니다.

- Home 탭에서 Import job type from file을 선택하고 `AIM2JOB.ctmai`의 위치를 선택합니다.
- 제공된 예제를 수정하려면 Application Integrator를 숙지해야 합니다.
- Control-M 설명서의 지침에 따라 작업 유형을 배포합니다.

Control-M 연결 프로필 생성

연결 프로파일은 애플리케이션의 특정 인스턴스에 대한 연결 속성과 보안 인증을 정의합니다. 여러 작업에서 각 연결 프로파일을 참조할 수 있습니다. 애플리케이션과 보안 인증의 고유한 조합마다 별도의 프로파일을 가질 수 있습니다.

연결 프로필을 정의하려면 Control-M 웹 인터페이스의 구성 도메인을 사용하거나 JSON을 사용할 수 있습니다. 다음 코드는 JSON 사용의 예입니다.

```
{
  "MICROFOCUS-WINDOWS": {

  "Type": "ConnectionProfile:Micro Focus Windows",
  "Centralized": true,
  "Description": "Micro Focus on Windows Connection Profile - file locations refer to
 the Enterprise Server host", "MFBSI Config Path": "C:\\microfocus\\ES\\mfbsi\\MFWIN\
\mfbsi.cfg",
  "MFBSI Directory Path": "c:\\microfocus\\es\\mfbsi\\MFWIN",
  "Runtime Environment": "\"C:\\Program Files (x86)\\Micro Focus\\Enterprise Developer\
\createenv.bat\"", "Run As": "dbauser",
  "RunAs-Pass": "*****"
  }
}
```

예제 코드는 GitHub 리포지토리의 파일에 `ConnectionProfile-Custom-M2-Replatform.json` 있습니다. 코드를 배포하려면 Control-M Automation API 배포 서비스를 사용합니다.

```
ctm deploy ConnectionProfile-Custom-M2-Replatform.json
```

Control-M Automation API의 응답은 다음과 유사합니다.

```
[
 {
    "deploymentFile": "cp-JOG-MF-WINDOWS.json",
    "deploymentState": "DEPLOYED_CONNECTION_PROFILES",
```

```
    "deploymentStatus": "ENDED_OK",
    "successfulFoldersCount": 0,
    "successfulSmartFoldersCount": 0,
    "successfulSubFoldersCount": 0,
    "successfulJobsCount": 0,
    "successfulConnectionProfilesCount": 1,
    "successfulDriversCount": 0,
    "isDeployDescriptorValid": false,
    "deployedConnectionProfiles": [ " MICROFOCUS-WINDOWS " ]
  }
]
```

Control-M Planning에서 작업 및 일정 생성

이제 플러그인을 배포하고 Micro Focus 연결을 위한 연결 프로필을 배포했으므로 작업 생성 및 실행을 시작할 수 있습니다.

각 Control-M for Micro Focus 작업은 4개의 섹션으로 구성된 속성 세트로 구성됩니다. 각 섹션에는 많은 속성이 있을 수 있습니다. 다음 목록에는 가장 일반적으로 사용되는 속성 중 일부가 나와 있습니다.

- 일반
  - 작업 이름
  - 작업이 속한 애플리케이션 및 하위 애플리케이션
  - 제출할 JCL
  - 사이트에서 제공하는 운영 설명서 링크
- 일정:
  - 이 작업을 실행할 수 있는 월 및 일
  - 비즈니스 회계 기간, 공휴일 또는 알고리즘으로 정의할 수 없는 기타 특수 날짜와 같은 달력
  - 시간 범위
  - 매시간 실행과 같은 주기적 동작
- 사전 조건
  - 업스트림 종속성(일반적으로 이 작업을 실행할 수 있게 되기 전에 성공적으로 완료되어야 하는 작업)
  - 필요할 수 있는 리소스
  - 필요할 수 있는 사용자 작업
- 작업 완료 시 Control-M에서 수행하는 작업:

- 작업 성공 또는 실패 결정(일반적으로 작업의 완료 코드를 기반으로 하지만 해당 설정을 재정의하여 출력 텍스트 또는 특정 상태 확인을 사용할 수 있음)
- 실패 또는 성공에 대한 알림(예: 이메일)
- 다운스트림 종속성에 대한 게시 상태

연결 프로파일과 마찬가지로 Control-M 웹을 사용하여 작업을 생성 및 배포하거나 JSON으로 작성하고 Control-M Automation API를 사용하여 배포할 수 있습니다.

다음 섹션에서는 몇 가지 일반적인 워크플로 시나리오를 살펴봅니다.

- 워크플로를 사용하여 작업 시작
- 예약된 빈도로 작업 실행 자동화

워크플로를 사용하여 작업 시작

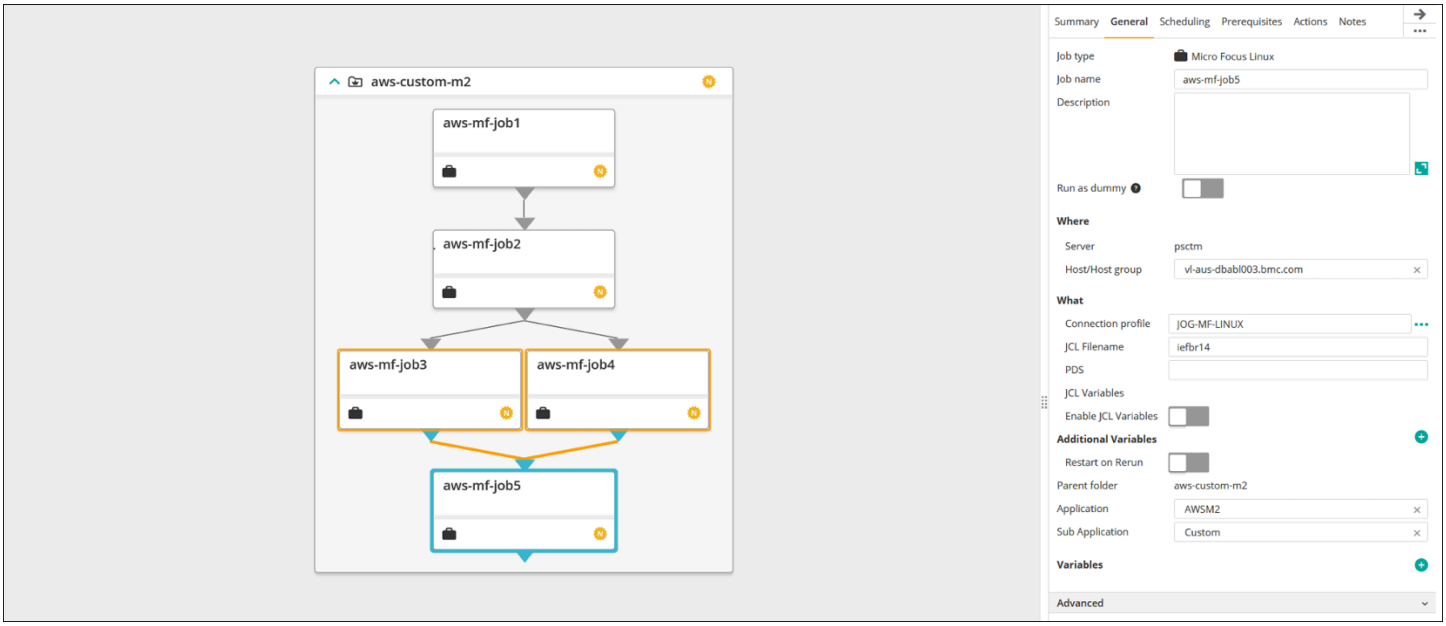
워크플로라는 작업 흐름을 생성합니다. 워크플로의 작업은 이전 작업이 성공적으로 완료되면 종속성과 상호 연결됩니다.

Control-M 웹 사용

Control-M 사용자 인터페이스에서 작업을 시작하려면 다음을 수행합니다.

1. Planning 영역에서 새 작업 영역을 추가합니다. 그러면 빈 폴더 객체가 있는 캔버스가 열립니다.
2. 메뉴 모음에서 작업 추가를 선택하고 Micro Focus Windows 또는 Micro Focus Linux 작업을 선택한 다음 폴더의 얼굴로 끕니다.
3. 작업의 색상이 녹색으로 바뀌면 삭제합니다. 오른쪽 창에는 일반, 예약, 사전 조건 및 작업 섹션이 포함되어 있습니다. 작업을 생성하려면 Control-M 설명서의 표준 지침을 참조하세요.
4. Micro Focus용 Control-M 작업 빌드에 대한 지침은 Control-M 설명서를 참조하세요. 이 파일럿의 경우 5개 작업에 대해 2~4단계를 반복하여에서 `aws-mf-job1`로 이름을 지정합니다`aws-mf-job5`.
5. 이러한 작업을 원하는 흐름에 연결하려면 작업 객체에서 조건 삼각형을 선택하고 다음 작업으로 끕니다. 예를 들어, 아래에서 조건 삼각형을 선택하고 로 `aws-mf-job1`끕니다`aws-mf-job2`. 이 작업은 `aws-mf-job1`의 이전 버전을 만듭니다`aws-mf-job2`. 기본적으로가 실행 자격을 갖추기 전에 `aws-mf-job2`가 성공적으로 완료되어야 `aws-mf-job1` 합니다.

다음 다이어그램에서는 기본 작업 흐름의 Control-M Planning 뷰를 보여줍니다.



이미지 제공: BMC Software, Inc. ©2022

JSON 사용

동일한 흐름을 JSON으로 코딩할 수 있습니다. GitHub 리포지토리에서 예제 코드는 파일에 있습니다`Folder-Custom-M2-Replatform.json`.

```
{
  "aws-custom-m2": {
    "Type": "Folder",
    "ControlmServer": "psctm",
    "OrderMethod": "Manual",
    "Application": "AWSM2",
    "SubApplication": "Replatform-Custom",
    "aws-mf-job1": {
      "Type": "Job:Micro Focus Windows",
      "ConnectionProfile": "MICROFOCUS-WINDOWS",
      "Enable JCL Variables": "unchecked",
      "Restart on Rerun": "unchecked",
      "Recapture ABEND Codes": "Ignore",
      "Recapture COND Codes": "Ignore",
      "Auto Adjust Restart": "Ignore",
      "Set MF_UCC11": "Ignore",
      "Restart with Modified JCL": "No",
      "Application": "AWSM2",
      "SubApplication": "Replatform-Custom",
      "Host": "microfocus-es-host",
```

```
      "Output": {}
    },
    "aws-mf-job2": {
      "Type": "Job:Micro Focus Windows",
      "ConnectionProfile": "MICROFOCUS-WINDOWS",
      "Enable JCL Variables": "unchecked",
      "Restart on Rerun": "unchecked",
      "Recapture ABEND Codes": "Ignore",
      "Recapture COND Codes": "Ignore",
      "Auto Adjust Restart": "Ignore",
      "Set MF_UCC11": "Ignore",
      "Restart with Modified JCL": "No",
      "Application": "AWSM2",
      "SubApplication": "Replatform-Custom",
      "Host": "microfocus-es-host",
      "Output": {}
    },
    "aws-mf-job3": {
      "Type": "Job:Micro Focus Windows",
      "ConnectionProfile": "MICROFOCUS-WINDOWS",
      "Enable JCL Variables": "unchecked",
      "Restart on Rerun": "unchecked",
      "Recapture ABEND Codes": "Ignore",
      "Recapture COND Codes": "Ignore",
      "Auto Adjust Restart": "Ignore",
      "Set MF_UCC11": "Ignore",
      "Restart with Modified JCL": "No",
      "Application": "AWSM2",
      "SubApplication": "Replatform-Custom",
      "Host": "microfocus-es-host",
      "Output": {}
    },
    "aws-mf-job4": {
      "Type": "Job:Micro Focus Windows",
      "ConnectionProfile": "MICROFOCUS-WINDOWS",
      "Enable JCL Variables": "unchecked",
      "Restart on Rerun": "unchecked",
      "Recapture ABEND Codes": "Ignore",
      "Recapture COND Codes": "Ignore",
      "Auto Adjust Restart": "Ignore",
      "Set MF_UCC11": "Ignore",
      "Restart with Modified JCL": "No",
      "Application": "AWSM2",
      "SubApplication": "Replatform-Custom",
```

```
      "Host": "microfocus-es-host",
      "Output": {}
    },
    "aws-mf-job5": {
      "Type": "Job:Micro Focus Windows",
      "ConnectionProfile": "MICROFOCUS-WINDOWS",
      "Enable JCL Variables": "unchecked",
      "Restart on Rerun": "unchecked",
      "Recapture ABEND Codes": "Ignore",
      "Recapture COND Codes": "Ignore",
      "Auto Adjust Restart": "Ignore",
      "Set MF_UCC11": "Ignore",
      "Restart with Modified JCL": "No",
      "Application": "AWSM2",
      "SubApplication": "Replatform-Custom",
      "Host": "microfocus-es-host",
      "Output": {}
    },
    "leftbranch": {
      "Type": "Flow",
      "Sequence": [
        "aws-mf-job1",
        "aws-mf-job2",
        "aws-mf-job3",
        "aws-mf-job5"
      ]
    },
    "rightbranch": {
      "Type": "Flow",
      "Sequence": [
        "aws-mf-job2",
        "aws-mf-job4",
        "aws-mf-job5"
      ]
    }
  }
}
```

이 흐름을 배포하려면 Control-M Automation API를 사용합니다.

```
ctm deploy Folder-Custom-M2-Replatform.json
```

예약된 빈도로 작업 실행 자동화

이전 단계에서 생성한 흐름을 사용하여 기본 일정 예약과 런타임 일정 예약을 추가할 수 있습니다.

- 기본 일정은 작업을 실행할 수 있는 일수를 정의합니다(예: 평일, 근무일, 월말 또는 분기말에만). 예약 탭에서 기본 예약을 설정할 수 있습니다.
- 런타임 예약은 실행이 가능한 날(예: 이전 작업이 완료된 후, 매시간, 지정된 리소스를 사용할 수 있는 후 또는 수동 확인 후에만) 작업이 실행되는 시기를 결정합니다.

일정 섹션에서 일부 런타임 일정을 정의하고 사전 조건 섹션에서 다른 런타임 일정을 정의할 수 있습니다.

작업 모니터링

Control-M 모니터링 도메인과 Micro Focus Enterprise Server Common Web Administration 사용자 인터페이스에서 작업을 모니터링하고 확인할 수 있습니다.

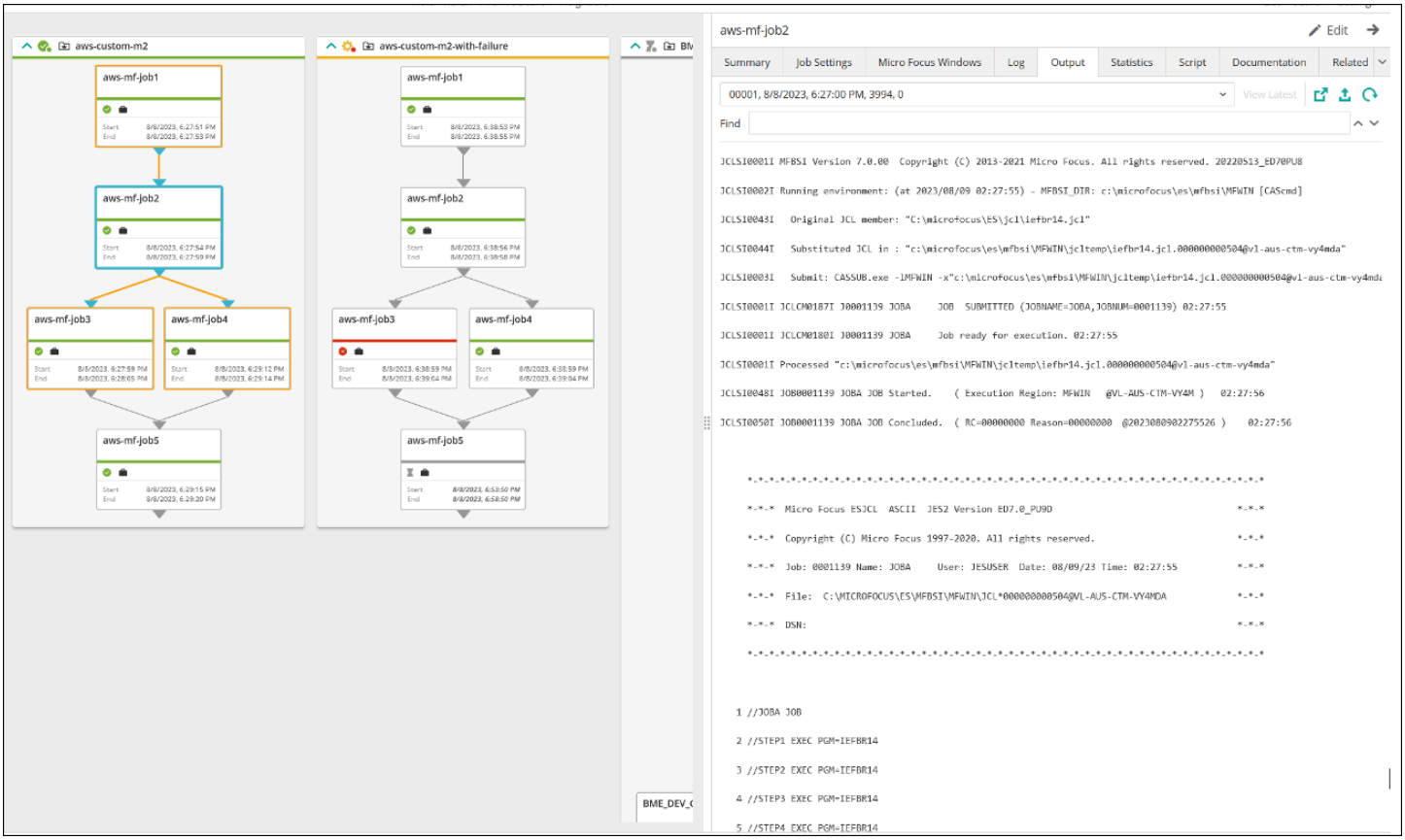
Control-M Monitoring

Control-M Monitoring 영역에서 작업 제출 및 실행을 모니터링할 수 있습니다. 기본적으로 AWS Mainframe Modernization 서비스 작업은 다른 모든 Control-M 작업과 함께 표시됩니다. 다른 워크로드(또는 다른 필터링 요구 사항) 없이 AWS Mainframe Modernization 서비스 작업만 보려면 뷰포인트를 생성할 수 있습니다.

Viewpoint는 작업 정보뿐만 아니라 업스트림 및 다운스트림 종속성과의 관계도 보여줍니다. 또한 워크플로에 AWS Mainframe Modernization 및 기타 유형의 Control-M 작업이 포함된 경우 모니터링 도메인에서 전체 흐름을 보고 관리할 수 있습니다.

Control-M 설명서의 모니터링의 뷰포인트 섹션을 방문하여 자세한 단계를 수행할 수 있습니다.

다음 스크린샷은 두 워크플로의 결과를 보여줍니다. 왼쪽에는 5개 작업이 모두 녹색으로 표시된 상태로 워크플로가 성공적으로 완료되었습니다. 오른쪽에서는가 실패 상태를 `aws-mf-job3` 반환하고 워크플로가 중지되어 대기 일정 상태로 남아 있기 때문에 워크플로가 부분적으로만 성공`aws-mf-job5`합니다.



이미지 제공: BMC Software, Inc. ©2022

모범 사례

초기 계획 및 통합 단계에서 다음과 같은 모범 사례를 따르는 것이 좋습니다.

- 통합하기 전에 마이그레이션하거나 자동화해야 하는 워크로드와 프로세스를 철저히 이해합니다. 이를 통해 마이그레이션에 가장 중요한 작업을 식별하고 Control-M을 사용하여 일정 및 자동화를 계획할 수 있습니다.
- 메인프레임 워크로드를 로 마이그레이션할 때 Control-M을 사용하여 처음부터 자동화를 AWS계획합니다. 클라우드 환경에서 작업 및 워크플로를 예약, 관리 및 모니터링하는 방법을 고려합니다.
- 중앙 집중식 연결 프로파일을 사용하면 관리할 객체 수가 줄어들고 Control-M Agent의 탄력적 배포가 간소화되므로 중앙 집중식 연결 프로파일을 사용하는 것이 좋습니다.
- 가능하면 메인프레임 마이그레이션을 점진적으로 수행하여 복잡성과 위험을 줄이세요. 증분 마이그레이션을 수행하면 마이그레이션 팀이 마이그레이션 진행 상황에 대해 더 빠르게 피드백을 제공할 수 있습니다. 기업은 이 피드백을 사용하여 내부 프로세스를 최적화하여 마이그레이션 속도를 가속화할 수 있습니다.
- 불필요한 작업을 피하려면 초기 단계의 작업 유형 및 연결 프로파일에 대해 제공된 템플릿을 사용하는 것이 좋습니다.

관련 리소스

참조

- Micro Focus
- Control-M
- Control-M 평가판
- Control-M Application Integrator
- Control-M 설명서
- 메인프레임 현대화: Micro Focus를 AWS 사용한의 DevOps(AWS 권장 가이드 패턴)

코드

- aws-mainframe-modernization-controlm-integration GitHub 리포지토리

기여자

기여자

다음은 이 문서의 기여자입니다.

- Sunil Bemarkar, Senior Partner Solutions Architect – DevOps, AWS
- Joe Goldberg, BMC Software, Inc.
- Pablo Alonso Prieto, 수석 메인프레임 아키텍트, AWS
- Vaidy Sankaran, Senior Modernization Architect, AWS
- Vij Balakrishn, CloudOps의 Senior Partner Development Manager AWS

문서 기록

아래 표에 이 가이드의 주요 변경 사항이 설명되어 있습니다. 향후 업데이트에 대한 알림을 받으려면 RSS 피드를 구독하십시오.

변경 사항	설명	날짜
섹션이 추가되었습니다.	새로운 섹션에서는 AWS Mainframe Modernization Control-M과의 Amazon EC2 통합에 대해 다룹니다.	2024년 2월 19일
최초 게시	—	2022년 11월 16일

기계 번역으로 제공되는 번역입니다. 제공된 번역과 원본 영어의 내용이 상충하는 경우에는 영어 버전이 우선합니다.