

기술 백서

퓨어스토리지 플래시블레이드(FlashBlade®)에서 버티카 이온 모드(Vertica Eon Mode) 구현

대규모 온-프레미스 환경에서 클라우드 수준의 간소함과 유연성을 확보할 수 있습니다.



목차

소개	3
이점 및 기능에 대한 요약	3
새로운 기능	4
기술 개요	5
버티카 이온 모드	5
퓨어스토리지 플래시블레이드	6
솔루션 개요	6
설계 토폴로지	7
핵심 솔루션 이점	8
확장성 및 용량	8
운영 효율성	9
데이터베이스 성능: 엔터프라이즈 모드 vs. 이온 모드	10
개발, 테스트 및 오프로드 솔루션	11
솔루션 개요	11
Fast CopyObject 기능	12
장애 시 일관성(crash-consistent)이 보장되는 DB 복제본의 신속한 생성	13
데이터베이스 복사본을 온라인으로 가져오기	14
재해 복구 및 백업 솔루션	17
플래시블레이드 오브젝트 복제	17
VBR 타겟 및 소스로 사용되는 플래시블레이드	18
재해 복구를 위한 플래시블레이드 복제	19
버티카 VBR 백업 및 복원	23
버티카 데이터 보호 및 오프로드 모범 사례	25
자동화 예시	25
요약	25
참고자료	26
부록: 플래시블레이드 스토리지를 통한 버티카 구현의 모범 사례	27



소개

버티카 분석 플랫폼(Vertica Analytics Platform)은 대규모 데이터 세트의 실시간 고급 분석을 지원하는 고급 SQL 데이터 웨어하우스입니다. 버티카를 분석 데이터베이스 시스템으로 구축하면 익숙한 SQL 쿼리로 페타바이트 규모의 데이터 세트를 즉시 분석할 수 있습니다.

지금까지 버티카 아키텍처는 분산 DAS(Direct-Attached Storage)를 지원해왔습니다. 이러한 접근 방식은 10~15년 전의 비즈니스와 IT 요구는 충족할 수 있었지만, 오늘날의 역동적인 IT 환경에서 구현을 확장하고 관리하는 데 필요한 민첩성은 부족합니다.

최근에 출시된 버티카 이온 모드는 컴퓨팅과 스토리지를 독립적으로 확장되는 풀로 분리한 새로운 아키텍처 및 구현 모델입니다. 기존 버티카 엔터프라이즈 모드와 달리, 버티카 이온 모드는 모든 사용자 데이터를 위한 공용 스토리지로 안정적이고 확장 가능한 AWS S3 호환 오브젝트 저장소를 필요로 합니다. 이를 위해 퓨어스토리지와 버티카가 파트너십을 맺고 퓨어스토리지 플래시블레이드에 기반한 버티카 이온 모드를 제공하게 되었습니다. 버티카는 원래 클라우드 네이티브 애플리케이션을 지원하도록 이온 모드를 설계했습니다. 2019년 버티카는 이온 모드에 퓨어스토리지의 초고속 통합 파일 및 오브젝트 저장소인 플래시블레이드를 결합하여 온-프레미스 버티카 환경에 향상된 확장성, 운영 효율성 및 동급 혹은 그 이상의 성능을 제공해왔습니다.

이점 및 기능에 대한 요약

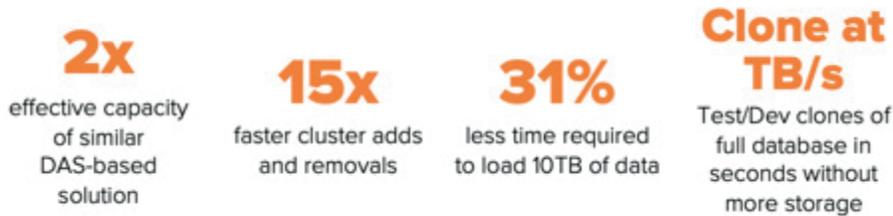
이 백서에서는 버티카 이온 모드와 퓨어스토리지 플래시블레이드를 간단하게 통합하는 방법을 설명하며, 다음과 같은 통합 솔루션의 이점, 새로운 기능 및 구현 모범 사례가 포함되어 있습니다.

- **간소화 및 확장성:** 컴퓨팅과 별도로 페타바이트 규모로 용량 확장 가능합니다.
 - 스토리지와 컴퓨팅이 분리되어, 노드를 추가하지 않고도 스토리지를 확장할 수 있습니다. 따라서 DAS 기반 시스템으로 확장하는 동시에 노드가 늘어나면서 복잡성이 기하급수적으로 증가하는 것을 방지할 수 있습니다.
 - 버티카의 컬럼 기반(columnar) 압축은 퓨어스토리지 플래시블레이드의 압축과 중앙화된 내장애성(fault-tolerant) 스토리지 시스템의 장점을 결합하여, 유사하게 구성된 DAS 기반의 버티카 엔터프라이즈 시스템보다 2배의 유효 스토리지 용량을 제공합니다.
- **운영 효율성:** 클러스터 노드 추가 및 제거 속도가 15배 빠릅니다.
 - 버티카 팀은 비즈니스 가치를 창출하는 작업에 집중할 수 있어야 합니다. 팀은 데이터와 비즈니스 분석가들이 사용할 수 있도록 올바른 데이터 소스를 가져와 ETL(추출, 변환, 로드)을 수행하고 집계하여 데이터를 포맷하길 원합니다. 또한 노드 추가, 패치 적용 및 업데이트와 관련된 유지 관리에 소요되는 시간을 최소화하고 싶어합니다.



- 시간과 리소스를 소비하며 클러스터를 리밸런싱하지 않고, 원하는 방식으로 클러스터를 확장 혹은 축소할 수 있습니다. 예를 들어, 플래시블레이드를 사용하는 버티카 이온 모드의 경우 기존 8 노드 클러스터에 8개의 노드를 추가하는 데 5분이 걸렸지만, 유사한 DAS 기반 시스템에서는 74분이 걸렸습니다.
- **데이터베이스 성능:** 더 탁월한 로드 성능을 확보하고 쿼리 성능을 향상할 수 있습니다.
 - 버티카의 엔드 유저는 필요한 분석을 수행할 수 있도록 일관되고 빠른 쿼리 성능을 기대합니다. 올플래시 스토리지 시스템인 버티카와 퓨어스토리지는 고객의 기대에 부응할 수 있는 유일한 솔루션입니다.
 - 테스트 결과, 이 솔루션은 10TB의 데이터를 로드하는 데 54분이 걸린 반면, DAS 기반 시스템은 71분이 걸렸습니다.
 - 4개의 동시 사용자 세션에서 동일한 쿼리 집합을 실행했을 때 버티카 이온 모드는 엔터프라이즈 모드와 비슷하거나 더 나은 성능 노드를 제공하는 것으로 나타났습니다.
- **복원성:** 재해 복구와 신속하고 공간 효율적인 복제본 생성을 통해 혜택을 얻을 수 있습니다.
 - 솔루션은 백업 및 복구를 위한 가속화된 워크플로우, 빠르고 공간 효율적인 복제를 통해 개발/테스트 팀을 위한 분리된 작업 환경을 제공하여 장애와 오류에 대한 복원성을 지원합니다.

퓨어스토리지 플래시블레이드에 구현된 버티카 이온 모드의 확장성, 간소화, 복원성 및 운영 효율성과 솔루션의 성능을 통해 현 상태를 유지하는데 그치지 않고 비즈니스의 가치를 더 높이는데 집중할 수 있습니다.



새로운 기능

버티카 10.x 및 퓨리티(Purity) 3.x 릴리스에서는 솔루션에 다음 기능들이 추가되었습니다.

- **재해 복구 및 백업 솔루션:** 퓨리티//FB 3을 사용해 플래시블레이드는 이제 비동기식 오브젝트 복제를 제공합니다. 버티카 백업 및 복구 유틸리티를 결합하면, 로컬 또는 원격 사이트에 데이터를 백업하여 버티카 데이터베이스를 보호하고 복구할 수 있는 빠른 워크플로우를 확보할 수 있습니다.
- **빠르고 공간 효율적인 복제본 생성 - Fast CopyObject 기능:** 놀라울 정도로 빠른 메타데이터 연산을 사용해 버티카 이온 모드 데이터베이스의 모든 오브젝트를 공간 효율적으로 복사할 수 있습니다. 복제 클론(replica clone)은 계산을 오프로드하거나, 개발/테스트를 위해 또는 재해 복구를 검증하는 데 사용할 수 있습니다. 17.2TB 데이터베이스(압축되지 않은 50TB)의 복사본 10개를 생성하는 테스트에서, 추가 스토리지를 사용하지 않고 각 복사본을 생성하는 데 약 5.20초가 걸렸습니다. 버티카 엔터프라이즈 모드에서는 유사한 연산을 수행하는 데 스토리지 요구 사항의 20배가 필요하며 몇 시간에서 최대 며칠이 소요될 수 있습니다.
- **개발, 테스트 및 쿼리 오프로드 솔루션:** 플래시블레이드를 활용하면 몇 분 안에 대용량 버티카 데이터 웨어하우스를 복제하고 사용할 수 있습니다. 퓨리티//FB 3 버전부터는 개발자, 데이터 과학자 및 테스트 환경을 위해 공간 효율적인 버티카 이온 데이터베이스 클론을 신속하게 생성할 수 있습니다.



기술 개요

버티카 이온 모드

버티카는 분석을 위한 컬럼 지향(column-oriented) 관계형 SQL 데이터베이스로, 처음에는 분산된 비공유(shared-nothing) 아키텍처에 구축되었습니다. 기존에는 버티카 엔터프라이즈라 불리며 분산 DAS(DDAS) 노드가 있는 버티카 데이터베이스를 구현해야 했습니다. 엔터프라이즈 모드에서는 각 클러스터의 상용 서버가 로컬에 연결된 디스크들에 여러 조각의 데이터를 저장합니다.

최근 버티카는 클러스터의 스토리지와 컴퓨팅을 별개의 풀로 분리하는 이온 모드로 버티카를 출시했습니다. 버티카 이온 모드를 실행하면, 컴퓨팅 노드와는 별도로 Amazon S3 호환 오브젝트 스토리지의 공용 위치에 데이터베이스의 스토리지 계층이 유지됩니다. 필요에 따라 온디스크 캐시와 유사한 임시 저장 위치인 이온 모드 “디포(depot)”에 있는 노드에 데이터를 유지할 수도 있습니다. 이러한 데이터 스토리지의 중간층은 각 노드에 로컬로 데이터의 빠른 복사본을 제공하는 동시에 공용 스토리지에서 모든 레코드를 일관성 있게 유지합니다. 디포에서는 쿼리에서 자주 사용되는 데이터가 우선순위를 가집니다. 또한 새로 기록된 데이터를 공용 스토리지에 저장하기 전에 디포에 저장할 수 있습니다.

기존 분산 DAS 아키텍처에서는 클러스터와 데이터 크기가 늘어나면 스토리지 관리가 복잡해집니다. 따라서 유지 관리 업그레이드, 노드 멤버십 같은 관리 문제가 운영 워크플로우를 장악할 수 있습니다. 이는 디포의 상태가 일시적인 분리 아키텍처와 대조 됩니다. 분리된 아키텍처는 컴퓨팅 노드를 최소 상태로 유지하여 컴퓨팅 노드에 대한 유지 관리 작업이 훨씬 빠르고 간편합니다. 버티카와 퓨어스토리지는 기존 분산 DAS 아키텍처의 운영과 관련된 도전 과제를 공동으로 해결하여, 기업이 고객 경험에 초점을 맞추고 민첩한 인프라 설치 공간을 유지할 수 있도록 지원합니다.

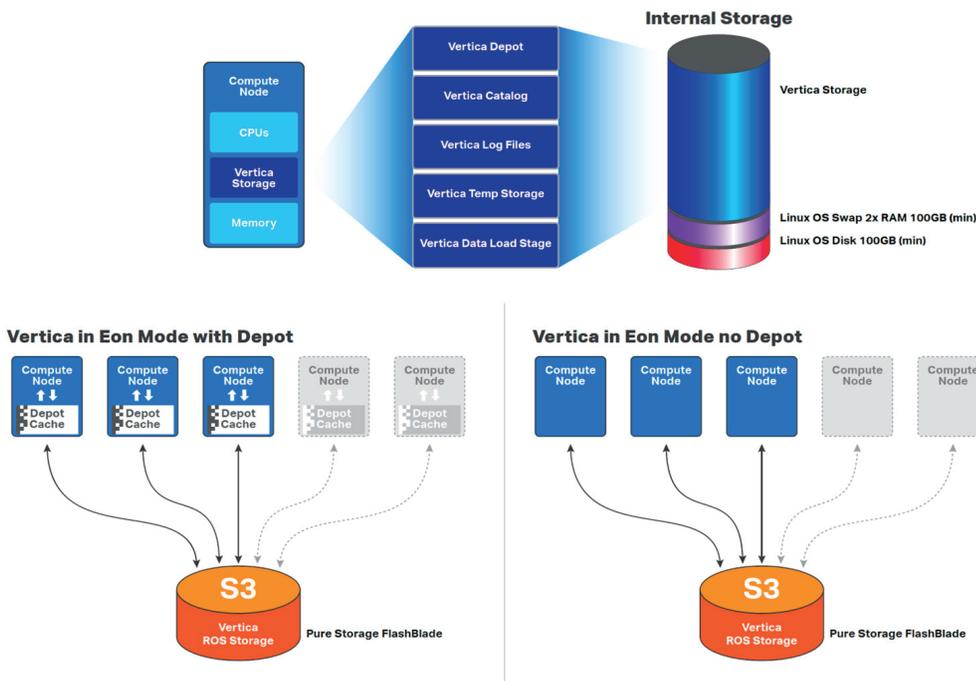


그림 1. 버티카 이온 모드 데이터베이스의 아키텍처



퓨어스토리지 플래시블레이드

퓨어스토리지는 현대적인 애플리케이션의 데이터 요구 사항을 충족하기 위해 클라우드에 최적화된 고성능 스토리지 플랫폼 플래시블레이드를 개발했습니다. 플래시블레이드는 비정형 데이터의 저장 및 처리에 최적화된 올플래시 시스템입니다. 플래시블레이드 시스템은 수천 개의 클라이언트를 위해 여러 파일 시스템과 멀티 테넌트(multi-tenant) 오브젝트 저장소를 동시에 호스팅할 수 있습니다. 성능, 용량 및 연결성을 위한 6가지 주요 혁신 기술이 플래시블레이드의 확장을 지원합니다.

- **고성능 스토리지:** 플래시블레이드는 데이터를 스토리지 장치에 저장하고, 기존의 회전식 디스크나 솔리드 스테이트 드라이브 같이 레이턴시가 높은 스토리지 미디어를 제거함으로써 올플래시 아키텍처의 이점을 극대화합니다. 확장 가능한 NVRAM을 각 스토리지 장치에 통합하면 성능 향상에 도움이 되고, 시스템에 새 블레이드를 추가함으로써 용량을 비례적으로 늘릴 수 있습니다.
- **통합 네트워크:** 플래시블레이드 시스템은 최대 100GB/초의 이더넷 링크를 통해 클라이언트와 내부 관리 호스트 간의 높은 통신 트래픽을 IPv4 및 IPv6 클라이언트 액세스 모듈을 지원하는 신뢰할 수 있는 단일 고성능 네트워크로 통합합니다.
- **퓨리티 운영 환경:** 퓨어스토리지의 대칭적인 운영 체제는 플래시블레이드의 패브릭 모듈에서 실행됩니다. 퓨리티는 플래시블레이드 스토리지의 블레이드들에 모든 클라이언트 연산 요청을 균등하게 분산하여 워크로드 밸런싱 문제를 최소화해줍니다.
- **파일 및 오브젝트를 위한 공통의 미디어 아키텍처 설계:** 플래시블레이드의 단일 미디어 아키텍처는 전체 플래시블레이드 구성에서 NFS, 고속 Amazon S3 오브젝트 스토리지, NFS over HTTP, SMB 등의 프로토콜을 통해 파일에 대한 동시 액세스를 지원합니다.
- **간편한 사용:** 플래시블레이드 스토리지의 퓨리티//FB는 일상적인 관리 작업을 자율적으로 수행함으로써 스토리지 운영을 간소화하고 시스템 관리 부담을 줄여줍니다. 견고한 운영 체제를 갖춘 플래시블레이드 스토리지는 구성 요소에 장애가 발생할 경우 자동으로 튜닝을 수행하고 시스템 알림을 제공합니다.
- **퓨어1(Pure1®):** 퓨어스토리지1 플랫폼은 전체 스택 분석, 퓨어1 메타(Meta™)의 AI 기반 역량을 통해 간소화된 클라우드 기반 관리와 간편한 예측 지원을 제공합니다. 퓨어1은 퓨어스토리지가 세계 최고 수준의 순고객추천지수(NPS)를 획득하는 데 많은 기여를 했습니다.

완전하게 구성된 플래시블레이드 시스템은 2개의 외부 패브릭 모듈(XFM)로의 고속 링크를 통해 상호 연결되는 최대 10개의 독립형 랙 마운트 새시로 구성됩니다. 각 새시의 후면에는 고속 이더넷을 통한 TCP/IP를 사용해 블레이드, 다른 새시 및 클라이언트를 상호 연결하기 위한 2개의 온보드 패브릭 모듈이 있습니다. 두 개의 패브릭 모듈은 서로 연결되어 있으며, 각각 컨트롤 프로세서와 이더넷 스위치 ASIC를 포함합니다. 안정성을 위해 각 새시에는 예비 전원 공급 장치와 냉각 팬이 장착되어 있습니다.

각 새시의 전면에는 데이터 연산과 저장을 위해 최대 15개의 블레이드가 장착됩니다. 각 블레이드 어셈블리는 프로세서, 통신 인터페이스 및 영구 데이터 저장을 위한 17TB 또는 52TB의 플래시 메모리를 갖춘 독립형 컴퓨팅 모듈입니다. 일반적인 플래시블레이드 시스템은 최소 7 x 17TB 구성으로 구축할 수 있으며 최대 150 x 52TB의 구성으로 페타바이트 규모까지 무중단으로 확장할 수 있습니다.

솔루션 개요

여기에 소개된 솔루션은 플래시블레이드에 기반한 버티카 분석 플랫폼으로 구성됩니다. 플래시블레이드는 다차원적 성능과 확장성을 갖추고 있기 때문에 버티카 데이터베이스를 배포하고 관리하는 데 이상적입니다. 테스트에서는 버티카 데이터베이스를 플래시블레이드에서 생성된 단일 Amazon S3 버킷에 배포했습니다. 컴퓨팅 및 스토리지 아키텍처가 분리되어 있기 때문에, 가장 작은 플래시블레이드 구성으로 시작해 필요에 따라 스토리지와 컴퓨팅 계층을 독립적으로 확장할 수 있습니다. 보편적인 엔터프라이즈 문제를 해결하기 위해 다음에 중점을 두고 시스템에 대한 테스트를 실시했습니다.



- 용량 절감
- 운영 효율성
- ETL 성능
- 쿼리 성능

설계 토폴로지

연구소에서 테스트한 솔루션에는 버티카 데이터베이스 노드를 호스팅하는 16대의 인텔 CPU 기반 서버와 데이터 스토리지 계층을 호스팅하는 15개의 블레이드가 장착된 퓨어스토리지 플래시블레이드 새시 1대가 포함되었습니다. 엔터프라이즈 모드 테스트에서는 각 서버에서 로컬 스토리지로 단일 SSD를 사용하고, 이온 모드 테스트에서는 동일한 SSD를 사용해 디포를 호스팅했습니다.

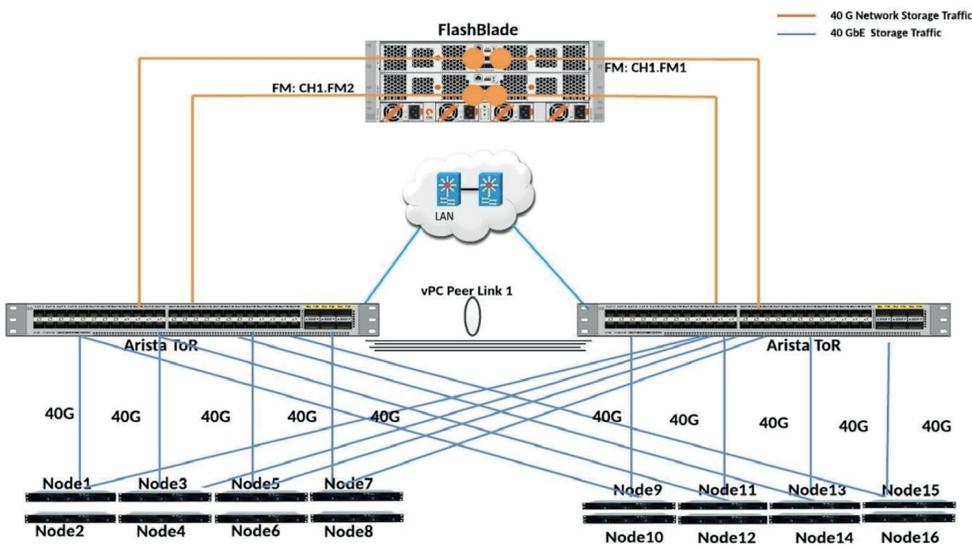


그림 2. 버티카 이온 모드 테스트 환경 - 물리적 토폴로지

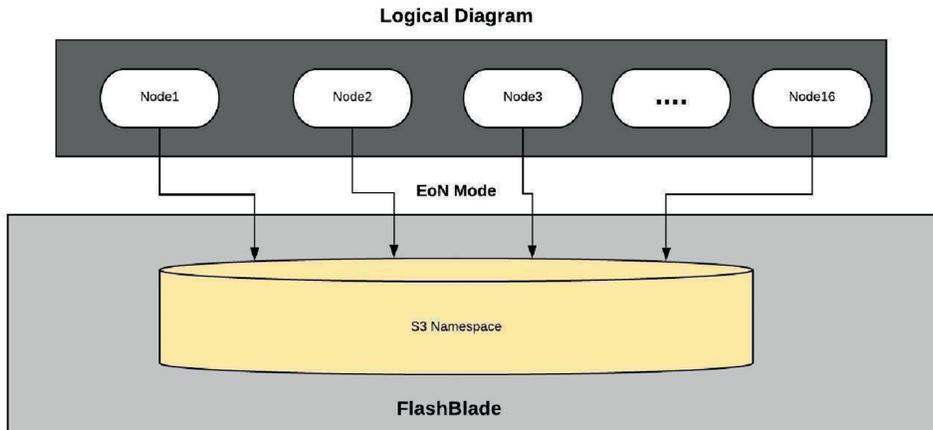


그림 3. 플래시블레이드 노드.

구성 요소	설명
플래시블레이드	15 x 17TB 블레이드
용량	원시 용량 240TB, 가용 용량 162.46TB(데이터 절감 안함)
연결	4 x 40Gb/초 이더넷(데이터), 2 x 1Gb/초 이중화; 이더넷(관리 포트)
물리적	4U
소프트웨어	퓨리티//FB 3.0

표 1. 플래시블레이드 구성

구성 요소	설명
버티카 노드	인텔 기반 서버 16대: 각각 인텔 제온 CPU 2개(48개 코어), 512GB 메모리, 1TB 로컬 SSD 포함
네트워크 인터페이스 카드	40Gbps 포트 2개
네트워크 스위치	랙 상단(TOR) 아리스타(Arista) 이더넷 스위치 2개

표 2. 서버 및 네트워크 구성

핵심 솔루션 이점

버티카 분석 솔루션과 퓨어스토리지 플래시블레이드로 구성된 솔루션은 스토리지와 컴퓨팅을 분리함으로써, 용량 절감, 운영 효율성 향상, 핵심 데이터베이스 성능 향상이라는 세 가지 주요 이점을 제공합니다.

확장성 및 용량

버티카는 데이터를 사전 압축된 형식으로 저장하여 스토리지 용량과 성능을 최적화하는 컬럼 지향 관계형 SQL 데이터베이스입니다. 버티카 데이터베이스의 엔터프라이즈 모드 구현에서는 노드 장애로부터 보호하기 위해 데이터를 둘 이상의 노드에 복제합니다. 복제본 수는 데이터베이스의 k-안전 계수로 정의됩니다. 기본 k-안전 계수는 1(각 조각(shard)의 복사본 두 개)이며, 1보다 높은 k-안전 계수를 사용하여 둘 이상의 동시 노드 장애로부터 보호할 수 있습니다.

k-안전 계수가 높을수록 엔터프라이즈 모드 구현 모델의 노드 안정성이 향상되지만, 이 때 스토리지의 용량과 쓰기 성능 요구 사항도 증가합니다. 이와는 대조적으로, 스토리지와 컴퓨팅이 분리된 버티카 이온 모드 모델의 아키텍처에서는 k-안전 계수 설정에 관계없이 Amazon S3 스토리지에 데이터 복사본을 하나만 저장합니다.

그림 4는 이온 및 엔터프라이즈 모드에서 동일한 10TB 데이터베이스에 의해 소비되는 스토리지를 기본 k-안전 계수 '1'로 비교한 것입니다. 10TB(TPC-DS 유사) 데이터베이스는 버티카의 고급 컬럼 기반 압축 기능 덕분에 3.8TB로 사전 압축됩니다. 이온 모드를 사용하면 복제가 필요하지 않아 스토리지를 추가로 절약할 수 있습니다.



이와 비교해 엔터프라이즈 모드는 데이터 복사본 두 개를 저장해야 하므로 스토리지 요구 사항이 최대 8TB까지 증가합니다. 이는 일부 구현 시나리오에서 이온 모드가 제공하는 또 다른 중요한 이점입니다. 백엔드 스토리지에 여러 개의 복사본을 저장할 필요가 없으므로, 더 높은 쓰기 성능을 위한 요구 사항을 크게 줄일 수 있습니다.

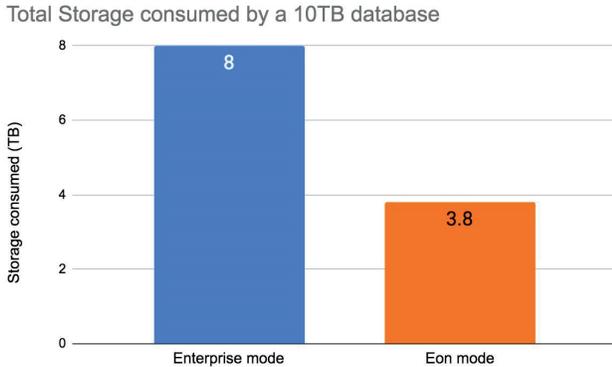


그림 4. 이온 모드를 통한 용량 절감

또한, 스토리지와 컴퓨팅이 분리되어, 노드를 추가하지 않고도 스토리지를 확장할 수 있습니다. 따라서 대규모 노드 증가로 인한 복잡성이 기하급수적으로 증가하는 것을 방지할 수 있습니다.

운영 효율성

분리된 컴퓨팅 및 스토리지 아키텍처의 두 번째 이점은 운영 효율성입니다. 여기에는 얼마나 쉽고 빠르게 환경을 확장하여 새로운 비즈니스 도전 과제를 충족하고 유지 관리 릴리스에 패치를 적용할 수 있는지 등이 포함됩니다. 그림 5는 라이브 10TB 데이터베이스에 있는 8개의 컴퓨팅 노드에서 16개의 노드로 버티카 데이터베이스 클러스터를 확장하는 데 걸리는 시간을 분단위로 보여 줍니다. 엔터프라이즈 모드 구현 모델의 경우, 추가 컴퓨팅을 활용하려면 새 노드를 추가한 후에 데이터를 물리적으로 재배포해야 합니다. 온라인 상태이지만 쿼리를 실행하지 않는 데이터베이스에서, 노드를 8개 더 추가한 후 시작된 데이터베이스 리밸런싱 작업을 완료하는 데 74분이 걸렸습니다. 데이터베이스에 대해 활성 쿼리가 실행되고 있는 경우 시간이 오래 걸릴 수 있습니다.

반면, 이온 모드 구현 모델에서는 같은 크기의 데이터베이스를 가상으로 재배포하는 데 5분밖에 걸리지 않았습니다. 이온 모드를 사용하면, 이는 실제 데이터 이동이 필요 없는 간단한 논리적 매핑 작업이 됩니다. 그 결과 클러스터 환경 확장에 걸리는 시간이 15배 단축되었으며, 데이터 재배포 시간은 데이터베이스 크기와 k-안전 계수에 맞게 확장될 것으로 예상됩니다. 운영 데이터베이스의 경우, 데이터 리밸런싱은 쿼리와 데이터 로드 성능에 영향을 미칠 수 있습니다. 많은 기업에서 버티카를 사용하여 크기가 수백 또는 수천 테라바이트에 달하고 k-안전 계수가 1보다 큰 데이터베이스를 실행하고 있습니다. 따라서 추가된 노드들에서 데이터베이스를 완전하게 리밸런싱하려면 성능에 영향을 미치고 며칠이 걸릴 수 있습니다.

그림 5의 오른쪽에 있는 차트는 8개 노드와 16개 노드 사이의 쿼리 성능을 비교합니다. 노드 수를 16개로 늘리면 쿼리 런타임이 예상대로 거의 50% 감소됩니다. 이러한 결과는 퓨어스토리지 플래시블레이드와 버티카 이온 모드를 사용해 얻을 수 있는 이점과 유연성을 보여줍니다. 중요한 점은 엔터프라이즈 모드 구현 모델에서는 데이터베이스 크기가 커지면 데이터베이스를 리밸런싱하는 데 필요한 시간이 증가하는 반면, 이온 모드 구현 환경에서는 일정하게 유지된다는 것입니다. 이를 통해 변화하는 요구 사항에 따라 손쉽게 컴퓨팅을 확장할 수 있습니다. 컴퓨팅 노드를 확장할 수 있을 뿐만 아니라 사용량이 적은 시간에는 줄일 수 있으므로 구현에 드는 전력 및 냉각 효율성을 크게 향상할 수 있습니다.



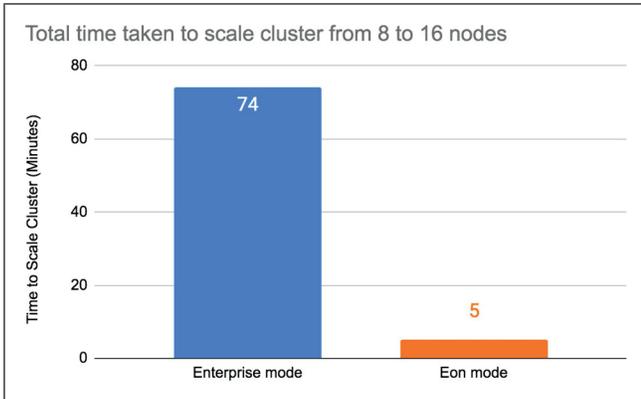
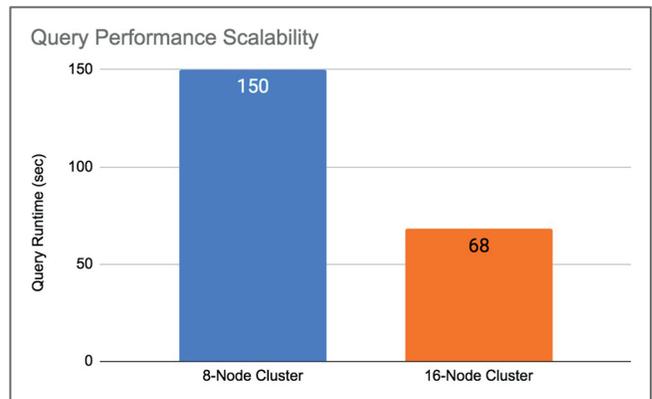


그림 5. 엔터프라이즈 모드와 이온 모드의 운영 효율성 비교



데이터베이스 성능: 엔터프라이즈 모드 vs. 이온 모드

이온 모드가 운영 효율성을 높여 주긴 하지만 Amazon S3 스토리지 프로토콜의 성능 역량에 대한 우려를 야기할 수 있습니다. Amazon S3 스토리지를 ‘저렴하고 깊은’ 스토리지와 동일시할 수도 있겠지만, 이제는 그렇지 않습니다. 처음 시작할 때는 퍼블릭 클라우드에서 저렴하고 집적도가 높은 스토리지 옵션이었던지는 몰라도, 이제 Amazon S3 스토리지는 최신 분석 애플리케이션을 위해 선택되는 프로토콜로 빠르게 진화하고 있습니다.

퓨어스토리지 플래시블레이드는 최신 애플리케이션 스택과 요구 사항을 염두에 두고 설계된 업계 최초의 초고속 통합 파일 및 오브젝트(Unified Fast File and Object, UFFO) 스토리지 어플라이언스입니다. 퓨어스토리지와 버티카는 협력을 통해 플래시블레이드 스토리지의 다차원적 성능과 확장성을 활용할 수 있도록 버티카 데이터베이스의 액세스 패턴을 최적화했습니다. 플래시블레이드 스토리지는 고대역폭 애플리케이션에 걸맞게 최적화되어 있으며 버티카의 ROS(Read-Optimized-Store) 데이터 형식에 적합합니다. 모든 테스트 결과에 따르면, 이온 모드 데이터베이스에서 버티카의 성능은 엔터프라이즈 모드와 동등하거나 우수하다는 사실을 알 수 있습니다. (부록 A에 설명된 적절한 튜닝 필요)

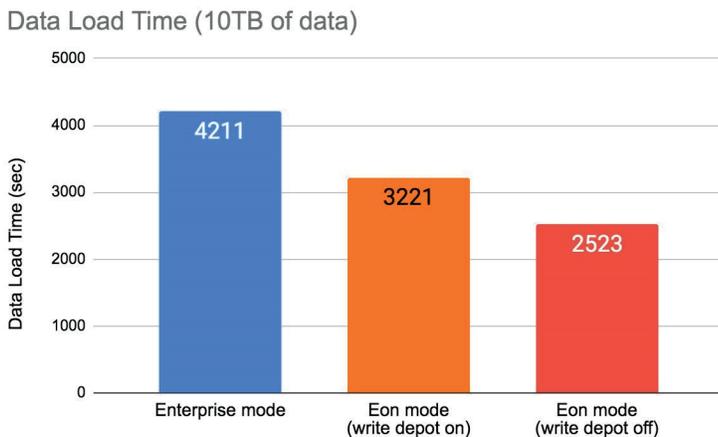


그림 6. 데이터베이스 로드 시간 - 일반적인 ETL 작업

그림 6은 엔터프라이즈 모드와 이온 모드에서 10TB 버티카 데이터베이스의 데이터 로드 시간을 비교한 것입니다. 이 테스트에서는 데이터베이스 로드에 미치는 영향을 수치화하기 위해 쓰기 디포 설정도 다양하게 변화시켜 봤습니다. UseDepotForWrites 기능이 활성화되면 Amazon S3 스토리지로 플러시(flush)되기 전에 데이터가 디포에 기록됩니다. 데이터에 따르면 데이터베이스가 노드 장애로부터 보호하기 위해 최소 2개의 데이터 복사본을 생성해야 하기 때문에, 엔터프라이즈 모드에서 10TB의 데이터를 인제스트하는 데 가장 오랜 시간(71분)이 걸린 것으로 나타났습니다.



k-안전 계수를 늘리면 엔터프라이즈 모드 데이터베이스의 데이터 로드 시간이 더욱 증가할 수 있습니다. 쓰기 디포가 활성화 및 비활성화된 이온 모드에서 동일한 10TB 데이터는 각각 54분과 43분 만에 로드되었습니다.

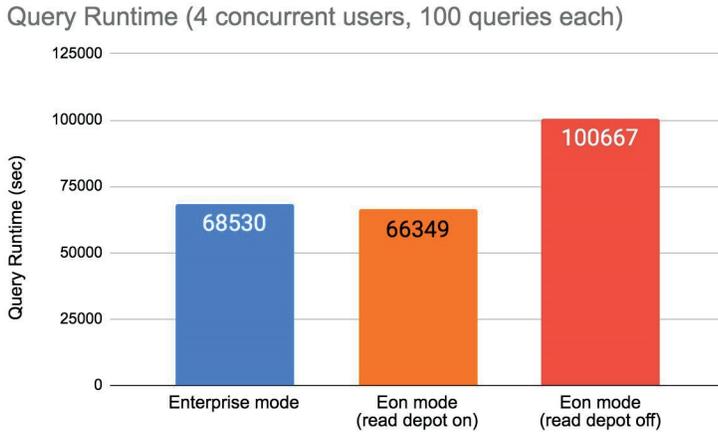


그림 7. 데이터베이스 쿼리 성능 - 엔터프라이즈 모드 vs. 이온 모드

그림 7은 10TB 버티카 엔터프라이즈 데이터베이스의 데이터베이스 쿼리 실행 시간으로 이온 모드에 구현된 동일한 크기의 데이터베이스와 비교한 것입니다. 동시 데이터베이스 액세스를 모방하기 위해, 4개의 동시 사용자 세션에서 동일한 쿼리 집합을 실행했습니다. 또한 이온 모드 테스트에서 읽기 디포 설정을 전환하여 성능에 미치는 영향을 확인해보았습니다. 테스트 결과, 읽기 디포를 활성화한 상태에서 버티카 이온 모드가 엔터프라이즈 모드보다 약간 더 나은 성능을 보였습니다. 읽기 디포의 영향은 데이터 액세스 패턴 및 데이터베이스 크기에 따라 크게 달라질 수 있습니다. 데이터베이스에서 이러한 설정을 테스트함으로써, 환경에 적합한 값을 결정할 수 있습니다.

개발, 테스트 및 오프로드 솔루션

일반적인 도전 과제 중 하나는 운영 데이터 웨어하우스의 복사본을 스프린업하기 위해 몇 시간(심지어 며칠)을 기다려야 한다는 것입니다. 가장 오래 기다린 시간은 얼마인가요? 추가 리소스, 클라우드 스토리지 비용 및 직원 시간 측면에서 비용이 얼마나 소요되었나요?

퓨어스토리지 플래시블레이드를 위한 버티카 이온 모드는 다음과 같은 뛰어난 솔루션을 제공합니다. 퓨리티//FB3부터는 버티카 이온 데이터베이스의 공간 효율적인 클론을 신속하게 생성할 수 있습니다. 이 섹션에서는 플래시블레이드를 활용해 몇 분 안에 대용량 버티카 데이터 웨어하우스를 복제하고 사용하는 방법을 알아봅니다. 이러한 데이터 웨어하우스 클론은 최신 데이터에서 새로운 코드를 개발하고 테스트하는 데 유용합니다. 클론을 사용하면 운영 환경에 영향을 미치지 않고 실제 데이터베이스의 콘텐츠에 대한 스키마나 데이터의 변환을 테스트할 수 있습니다. 또한 데이터베이스의 재해 복구(DR) 복사본은 추가 쿼리를 2차 위치로 오프로드하는 데 사용될 수 있는 액티브 복사본으로 변환하는 데 사용할 수 있습니다. 재해 복구 복사본을 무효하게 할 수 있는 잠재적인 변경 사항에 대해 우려할 필요가 없습니다.

솔루션 개요

다음 각 활용 사례에 대해 자세히 살펴보겠습니다.

- 쿼리 오프로드:** 많은 조직이 데이터 웨어하우스의 DR 또는 2차 복사본을 유지 관리하는 데 투자해 왔습니다. 수집되는 인사이트가 비즈니스에 더욱 중요해지고 있기 때문에, 데이터 웨어하우스의 신속한 복구와 비즈니스 연속성을 보장하는 것이 필요합니다. 그러나 데이터베이스의 또 다른 복사본과 두 번째 클러스터 노드 세트가 유휴 상태로 있는 것은 리소스 낭비입니다. 또한 조직은 DR 데이터베이스가 양호한 상태인지



확신할 수 있어야 합니다. DR 데이터베이스의 클론을 사용하면 2차 사이트에서 두 가지 모두를 안전하게 지원할 수 있습니다.

- 개발자 및 데이터 과학자의 DB 복사본:** 개발자와 데이터 과학자들은 실질적인 최신 데이터와 스키마가 포함된 데이터베이스의 개발 복사본에 액세스할 수 있을 때 훨씬 더 생산적일 수 있습니다. 데이터가 망가질 것을 염려하지 않아도 되는 데이터베이스 복사본, 즉, 데이터베이스를 수정 및 쿼리할 수 있는 복사본이 필요합니다. 이상적으로는 각 사용자가 하나의 복사본을 가지는 것이 좋을 것입니다. 플래시블레이드 오브젝트 클론은 공간 효율성이 뛰어나고, 버티카는 일반적으로 각 오브젝트를 한 번만 쓰기 때문에 이제 가능한 일입니다.
- DB의 테스트 복제본:** 오늘날 기술적으로 성숙된 조직은 자동화된 테스트, 연속 통합 및 연속 배포(CI-CD) 방식을 사용합니다. 실질적인 최신 데이터와 스키마를 사용해 테스트를 해야 가장 확실한 결과가 나옵니다. 플래시블레이드 오브젝트 클론은 빠르기 때문에 수백 테라 바이트를 몇 분 안에 보관하는 데이터베이스의 클론을 효율적으로 생성할 수 있습니다. 이는 대부분의 자동화된 테스트와 CI-CD 프레임워크의 시간 제약 조건을 충족하며 더 짧은 시간 내에 더 높은 품질의 결과를 얻을 수 있게 합니다.

Fast CopyObject 기능

플래시블레이드는 놀라울 정도로 빠른 메타데이터 작업을 사용해 버티카 이온 모드 데이터베이스의 모든 오브젝트를 공간 효율적으로 복사할 수 있습니다. Amazon S3 API 호출을 받아 오브젝트를 동일한 버킷 내의 기존 오브젝트의 복사본으로 PUT할 때 플래시블레이드는 보다 최적의 접근 방식을 취할 수 있다는 사실을 이해합니다. 소스 오브젝트와 관련된 모든 것을 새 위치로 복사하는 대신, 플래시블레이드는 기존 소스 오브젝트의 데이터를 가리키는 새로운 내부 메타데이터 엔트리를 빠르게 만들 수 있습니다. 메타데이터 복사는 거의 즉각적으로 이뤄지며, 플래시블레이드에서 추가 공간을 거의 사용하지 않는다는 이점이 있습니다. 이러한 오브젝트는 또한 새 오브젝트로 덮어쓸 수도 있으므로, 데이터베이스 복사본은 업데이트(PUT)와 쿼리(GET)를 지원합니다.

데이터 세트	복사	s5cmd cp		s5cmd rm		s5cmd du		플래시블레이드 GUI		
		초	초	오브젝트	바이트	오브젝트	물리적	논리	DRR	
SF=50000	0			44590	18.4T	44.59K	17.12T	18.40T	1.1	
SF=50000	1	4.54	6.50	89180	36.7T	89.18K	17.12T	36.79T	2.1	
SF=50000	2	4.83	6.54	133770	55.1T	133.77K	17.12T	55.18T	3.2	
SF=50000	3	5.23	6.56	178360	73.4T	178.36K	17.12T	73.57T	4.3	
SF=50000	4	5.28	6.65	222950	91.8T	222.95K	17.12T	93.27T	5.4	
SF=50000	5	5.40	6.63	267540	110.2T	267.54K	17.12T	110.4T	6.4	
SF=50000	6	5.32	6.67	312130	128.5T	312.13K	17.12T	128.52T	7.5	
SF=50000	7	5.55	6.69	356720	146.9T	356.72K	17.12T	146.88T	8.6	
SF=50000	8	5.03	6.77	401310	165.2T	401.31K	17.12T	165.23T	9.7	
SF=50000	9	5.41	6.68	445900	183.6T	445.90K	17.12T	183.59T	10.7	
SF=50000	10	5.47	6.63	490490	202.0T	490.49K	17.12T	201.95T	11.8	
Average		5.20	6.63							



표 3. 오브젝트 연산 시간 및 데이터 효율성

먼저 50TB의 원시 데이터로 시작해 버티카의 고급 컬럼 기반 압축을 사용하여 약 18.4TB의 논리적 스토리지로 압축했습니다. 이 데이터는 플래시블레이드의 44,590개 오브젝트에 저장되었으며, 이후 약 17.12TB의 물리적 스토리지로 압축되었습니다. 이는 약 1.074:1의 데이터 절감률(DRR)에 해당됩니다. 이 테스트에서는 데이터베이스 복사본 10개를 만들었습니다. 각 복사본을 생성하는데 평균 약 5.20초 정도 걸렸습니다. 유효 복사 속도는 논리적 스토리지의 경우 평균 약 3.54TB/초, 원래 원시 데이터의 경우 약 9.62TB/초입니다. 이는 플래시블레이드가 버티카 데이터베이스를 얼마나 빠르게 복제할 수 있는지를 보여 줍니다.

두 번째 주목할 것은 플래시블레이드에서 사용되는 물리적 스토리지가 17.12TB로 일정하게 유지된다는 점입니다. 데이터베이스의 사본이 추가되면, 오브젝트 수, 논리적 용량, 데이터 절감률이 예측 가능한 간격으로 증가합니다. 이는 플래시블레이드의 효율성을 강조해 보여주며, 동일한 데이터베이스의 여러 작업 복사본을 오프로드, 개발 및 테스트를 위해 사용할 수 있도록 해줍니다.

마지막 요소는 이러한 데이터베이스 클론을 제거하는 속도입니다. 자동화된 테스트와 검증에 사용하려면, 복제된 데이터베이스를 신속하게 제거하고 다음 단계로 넘어갈 수 있어야 합니다. 복제된 데이터베이스 오브젝트를 모두 제거하는 데 평균 6.63초가 걸렸습니다. 유효 속도로 보면 논리적 스토리지는 약 2.78TB/초, 원시 데이터는 7.54TB/초입니다. 이러한 속도라면, 페타바이트 규모의 데이터베이스도 몇 분 내에 복제하고 삭제할 수 있습니다.

장애 시 일관성(crash-consistent)이 보장되는 신속한 DB복제본 생성

장애 시에도 일관성이 보장되는 버티카 이온 데이터베이스 복사본을 만드는 작업은 모든 기저 오브젝트를 복사하는 것만큼 간단합니다. 데이터베이스가 버킷 기반이 아닌 특정 접두사로 시작되는 경우에 유용합니다. 가장 좋은 예는 데이터베이스에 대한 공용 스토리지 경로를 s3://vertica/dw01이 아니라 s3://vertica/prod/dw01과 같이 구성하는 것입니다. 이렇게 하면 동일한 버킷 내의 다른 접두사(예: s3://vertica/test23/dw01)로 간단히 복사할 수 있습니다.

AWS 버킷에서 스토리지 작업을 수행하는 데 사용할 수 있는 몇 가지 툴이 있습니다. 가장 자주 사용하는 툴은 s5cmd입니다. 이 툴은 **매우 빠르며, 간단한 스크립트를 사용해 고급 작업을** 수행할 수 있는 유연한 구문을 제공합니다. s5cmd를 사용하여 버티카 데이터베이스를 복제하는 명령은 다음과 같습니다.

```
s5cmd --endpoint-url=10.1.2.3 --log=error cp 's3://vertica/prod/dw01/*' s3://vertica/test23/dw01/
```

또한 \$HOME/.aws/credentials에서 플래시블레이드 버킷에 대한 자격증명을 정의해야 합니다. 소스 경로를 감싸는 작은 따옴표는 s5cmd에 매개 변수를 전달하기 전에 쉘이 와일드카드를 확인하지 않도록 하기 위한 것입니다. 기본 로깅 수준은 복사된 모든 오브젝트에 대해 상태 줄을 생성하는 info입니다. --log=error를 설정하면 메시지가 억제됩니다. 더 이상 필요하지 않은 복사본을 제거하려면 다음 명령을 사용합니다.

```
s5cmd --endpoint-url=10.1.2.3 --log=error rm 's3://vertica/test23/dw01/*'
```

데이터베이스의 일관성 보장

S3 공용 스토리지에 있는 버티카 이온 데이터베이스는 항상 장애 시 일관성이 보장되어야 하며, 이전 에포크(epoch)로부터 복구가 가능해야 합니다. 그러나 데이터베이스를 복제하기 전에 카탈로그를 동기화하고 공용 스토리지에 최근 변경 사항을 기록하는 것이 유용할 수 있습니다. 이를 위해서는 다음 명령을 사용합니다.



```
vsq1 -qt -c "SELECT hurry_service('System','TxnLogSyncTask'); SELECT sync_catalog();"
vsq1 -qt -c "SELECT count(*) from system_services where service_name='TxnLogSyncTask' and
last_run_end is null;" | sed -e '/^$/d' -e 's/ //g'
```

마지막 명령의 결과는 모든 노드에서 동기화 작업이 완료되었음을 나타내는 0(zero)이어야 합니다.

Fast CopyObject 기능에 대한 주의 사항

원래 오브젝트와 동일한 버킷에서 오브젝트 복사를 요청하는 데 사용되는 경우, 플래시블레이드는 항상 AWS S3 CopyObject API를 최적화합니다. 버킷 전반에서 오브젝트를 복사하거나, UploadPartCopy API를 통해 멀티파트 카피(multipart-copy)를 사용해 오브젝트를 복사하거나, 호스트에서 오브젝트의 복사본을 업로드할 때만 완전한 복사본이 생깁니다. 플래시블레이드에는 AWS처럼 CopyObject에 5GB 오브젝트 크기 제한이 없습니다. 단일 CopyObject API 호출을 사용하여 지원되는 모든 크기(현재 최대 64TB)의 오브젝트를 복사할 수 있습니다.

s5cmd cp 명령에는 CopyObject API의 사용을 보장하는 데 사용할 수 있는 --concurrency (-c) 및 --part-size (-p) 매개 변수가 포함되어 있습니다. 오브젝트 크기가 우려되면, 데이터베이스 접두사를 복제하는 데 사용되는 명령을 수정하여 --concurrency를 1로 설정하고 --part-size를 67108864(64TB)처럼 매우 큰 값으로 설정하면 됩니다.

```
s5cmd --endpoint-url=10.1.2.3 --log=error cp --concurrency=1 --part-size=67108864
```

Amazon S3 버킷으로 작업하는 다른 툴에도 유사한 옵션이 있을 수 있으므로 선택한 툴의 사용법을 확인하기 바랍니다. ~/.aws/config 파일 내에서 [multipart_threshold_parameter](#)를 설정할 수도 있습니다. 이는 AWS S3 SDK를 사용하는 모든 툴에 의해 픽업이 됩니다.

데이터베이스 복사본을 온라인으로 가져오기

복사된 데이터베이스를 온라인 상태로 가져오려면, 다른 클러스터에서 데이터베이스를 복원해 공용 스토리지 경로를 해당 복사본에 이르는 지점으로 설정 하면 됩니다. 간단하지만 몇 가지 주의 사항이 있습니다.

1. 대상 클러스터의 노드 수는 소스 클러스터와 같아야 합니다.
2. 복사된 오브젝트에는 cluster_config.json이 포함되며, 여기에는 향후 최대 15분 동안 클러스터 리스 타임 스탬프(cluster lease timestamp)가 포함됩니다. 복원 명령은 현재 시간이 기록된 리스(lease) 시간을 경과할 때까지는 작동하지 않습니다.
3. s5cmd 명령은 여러 TCP 연결을 사용하여 오브젝트를 복사합니다. 기본적으로 이러한 연결 중 대부분은 time_wait 상태에서 교착될 수 있습니다. 클라이언트 호스트 또는 사용자가 연결을 빠르게 열고 다시 사용할 수 없는 경우, 복사 작업이 느려지고 오류가 발생할 수도 있습니다.
4. 기본적으로 5GB보다 큰 오브젝트는 현재 최적화되지 않은 Multipart Upload API를 사용하여 복사해야 합니다. 이로 인해 추가 스토리지를 사용하는 복사본 속도가 느려질 수 있습니다.

클러스터 노드 수

버티칼을 사용하려면 공용 스토리지에 표시된 것과 동일한 수의 노드를 사용하여 클러스터를 복원해야 합니다. 이렇게 해야 데이터 조각들을 노드에 간단하게 매핑하고 복원된 클러스터에 충분한 리소스가 확보됩니다. 사용자들이 2차/개발/테스트 클러스터에서 동일한 수의 노드를 가지는 것이 이상적입니다. 이것이 가능하지 않은 경우, 요구 사항을 충족하는 몇 가지 방법이 있습니다.



- 소스 클러스터 축소:** 데이터베이스를 복제하기 전에 대상 클러스터와 노드 수가 같도록 소스 클러스터를 축소합니다. 클론 생성이 완료되면 노드를 클러스터에 다시 추가합니다. 버티카 이온 모드를 사용하면 작업이 비교적 간단합니다. 모든 데이터가 Amazon S3 공용 스토리지에 있고 노드는 대부분 스테이트리스(stateless) 상태이기 때문입니다. 제거되는 노드가 쿼럼(quorum)에 영향을 주지 않는 2차 하위 클러스터 내에 모두 포함되어 있으면 훨씬 간단합니다. 전체 프로세스를 느린 기간 동안 예약하고 자동화할 수 있습니다. 그러나 작업을 완료하는 데 몇 분 정도 걸릴 수 있습니다. 이 때문에 클론을 생성하고 새로 고칠 수 있는 빈도가 제한됩니다. 버티카의 성능과 SLA도 이러한 작업 중에 영향을 받을 수 있으며 노드 수는 줄어듭니다.
- 대상 클러스터 가상화:** 일반적으로 대상 클러스터에서 동일한 수의 노드를 보유하는 것과 관련된 문제는 비용 뿐입니다. 2차 위치에서 동일한 수의 서버를 구입하여 운영하기에는 비용이 너무 많이 듭니다. 버티카는 동일한 수의 노드를 필요로 하지만 이러한 노드가 동일한 성능을 발휘할 필요는 없습니다. 적은 수의 물리적 서버를 충분한 수의 버티카 노드로 가상화하는 것은 허용됩니다. 소스 클러스터와 비교해 성능, 가상화 및 관리 오버헤드에도 영향을 줄 수 있습니다. 그러나 비용 절감이 주된 과제이므로 2차 사이트의 성능 저하가 허용되는 경우에는 실행가능한 옵션이 될 수 있습니다.

클러스터 리스 타임아웃(Cluster Lease Timeout)

버티카는 클러스터 구성을 공용 스토리지 메타데이터 오브젝트(cluster_config.json)에 유지합니다. 이 오브젝트에는 클러스터를 정의하고 복원하는 데 필요한 모든 정보가 있습니다. 트래킹되는 한 가지 세부 정보는 *클러스터 리스 만료(Cluster Lease Expiration)* 입니다. 공용 스토리지에 있는 각 버티카 데이터베이스는 향후 일정 기간 동안 특정 클러스터에 “리스(leased)”됩니다. 이는 둘 이상의 클러스터가 동일한 데이터베이스를 사용하지 못하게 하는 보호 메커니즘입니다. 오브젝트는 정기적으로 최신 상태로 업데이트되고 리스는 다른 기간(예: 버티카 10.0의 경우 기본 15분)동안 연장됩니다. 클러스터에 문제가 발생하면 리스가 만료되고, 데이터베이스가 다른 클러스터에서 다시 복원될 수 있습니다.

cluster_config.json 오브젝트는 다른 모든 데이터베이스 오브젝트와 함께 복제되며, 원래 클러스터 리스는 리스가 만료될 때까지 복제된 데이터베이스가 다시 복원되지 않도록 막습니다. 일반적으로 이는 문제가 되지 않습니다. 클론이 데이터베이스 백업 또는 재해 복구에 사용되는 경우, 리스가 만료되고 나서야 복원이 될 수 있습니다. 하지만 클론 생성의 목적이 몇 가지 자동화된 테스트를 빠르게 실행하는 것이라면, 프로세스가 느려지고 시간당 수행할 수 있는 테스트 수가 제한됩니다.

가장 간단한 해결 방법은 데이터베이스를 복제하기 전에 중지시키고 클론 생성 작업이 완료되면 다시 시작하는 것입니다. 데이터베이스가 중지되면, 리스 시간이 현재 시간으로 업데이트되고 데이터베이스가 즉시 복원될 수 있습니다. 작업에 미치는 영향이 적은 상태로 데이터베이스를 중지할 수 있는 경우, 이 방법이 선호되고 지원되는 방법입니다. 여러 클론을 생성하는 경우, 첫 번째 클론에 대해서만 데이터베이스를 중지하면 됩니다. 첫 번째 클론을 복제하여 추가 클론을 생성할 수 있습니다.

데이터베이스를 중지하는 것이 실질적으로 가능하지 않은 경우, 한 가지 해결 방법은 스크립트를 사용하여 클론 경로에 클러스터 리스 타임스탬프를 다시 쓰기 하는 것입니다. 예를 들어, 다음의 간단한 awk 코드는 원본 오브젝트를 입력으로 사용하고 수정된 타임스탬프가 있는 버전을 stdout에 쓰기합니다:

```
#!/bin/awk -f
{
  switch ($0) {
    case / "ClusterLeaseExpiration" /:
      print strftime(" \\"ClusterLeaseExpiration\\" : \"%F %T.000000\\",", systime(), 1)
      break
    default:
  }

  print $0
}
```



```

    }
}

```

코드가(경로에 있는) `grab_release_now`라는 실행 파일에 있는 경우, 다음과 같이 사용할 수 있습니다.

```

[dbadmin@node01]$ s5cmd --endpoint-url=10.1.2.3 cat
s3://vertica/test23/dw01/metadata/verticadb/cluster_config.json | grab_release_now > /tmp/cfg_new
[dbadmin@node01]$ s5cmd --endpoint-url=10.1.2.3 cp /tmp/cfg_new
s3://vertica/test23/dw01/metadata/verticadb/cluster_config.json

```

그러면 클로닝된 데이터베이스는 즉시 복원될 수 있습니다. 위와 같은 내용이 테스트 자동화의 일부로 포함되어야 합니다. 50TB의 버티카 이온 데이터베이스를 사용한 테스트에서는, Ansible playbook을 사용하여 데이터베이스를 복제하고 다른 클러스터에서 복원하고, 시작하고 중지한 다음 삭제했습니다. 전체 주기가 몇 분 밖에 걸리지 않았습니다. 대부분의 시간은 복제와 클론 생성보다는 데이터베이스 작업에 소비되었습니다.

충분한 TCP/IP 연결

Amazon S3 작업은 기본적으로 HTTP를 통한 REST API 호출입니다. 이러한 호출 중 대부분은 각 작업에 대해 TCP/IP 연결을 열고 닫아야 합니다. 이는 매우 빠르게 수행되지만 기본적으로 닫힌 TCP/IP 연결은 `time_wait` 상태로 전환되어, 지연된 패킷이 제대로 연결되어 적절히 처리되도록 합니다. 기본 `time_wait` 간격은 다양하지만 일반적으로 적어도 60초입니다. 이는 현대적인 데이터센터 네트워크에서는 너무 보수적인 경향이 있습니다. 또한 각 연결에는 호스트에 포트가 필요하며, 사용 가능한 포트 수는 제한되어 있습니다. 수십만 개의 오브젝트가 있는 데이터베이스의 클론을 생성하려고 할 때, 모든 포트가 한 번 이상 사용되고 `time_wait` 상태에 갇힐 수 있습니다. 이 경우, 사용 가능한 포트가 릴리즈되는 속도에 따라 클론 생성 속도가 제한됩니다.

다행히, 최신 운영 체제에서는 다양한 관련 매개변수를 조정할 수 있습니다. CentOS 7.7을 사용한 테스트에서는 다음의 조정작업이 효과적으로 작동했습니다.

```

net.ipv4.tcp_fin_timeout=20
net.ipv4.tcp_tw_reuse=1
net.ipv4.ip_local_port_range=16384 65535

```

이는 일반적으로 `/etc/sysctl.conf` 파일에 추가된 다음 `sysctl -p`(`sudo` 또는 `root` 모두)를 실행하여 활성화합니다. `net.ipv4.tcp_fin_timeout` 설정은 `time_wait` 기간을 20초로 낮춥니다. `net.ipv4.tcp_tw_reuse` 설정을 활용하면 `time_wait` 상태에서 중단된 연결을, 이를 필요로 하는 새로운 연결에서 재사용할 수 있습니다. 마지막으로, `net.ipv4.ip_local_port_range` 설정은 시작 포트를 16384로 낮춰 사용 가능한 포트 수를 확장합니다. 최적의 설정을 찾으려면 사용하는 OS와 네트워크를 다양하게 실험해야 합니다.

이러한 조정은 S3 버킷에서 데이터베이스를 클론하는 데 사용되는 호스트/VM에 적용되어야 합니다. 모든 버티카 노드에 적용할 필요는 없지만, 그렇게 하는 것이 안전합니다. WAN을 통해 운영하면서 정체가 심하거나 QoS 제한이 있거나 패킷 손실이 상당한 경우가 있습니다. 이러한 경우, 매개변수를 기본값으로 남겨둘 필요가 있거나 좀 더 신중하게 평가해야 할 수 있습니다.



5GB보다 큰 오브젝트에 CopyObject API를 사용하는 방법

AWS S3는 5GB 이하의 오브젝트에 대해 CopyObject API 호출을 지원하며, 5GB보다 더 큰 오브젝트에 대해서는 멀티파트 카피 (Multipart Copy)가 필요합니다. 플래시블레이드는 최대 64TB의 오브젝트 크기를 지원하며 동일한 CopyObject API 크기 제한이 없습니다. 다양한 S3 톨은 크기 제한을 가정하고 최적화되지 않은 API를 사용합니다.

다행히도, 멀티파트 크기 임계값은 ~/.aws/config 파일에서 구성할 수 있습니다. 버티카와 함께 사용되는 플래시블레이드의 프로필에 다음 값을 추가합니다.

```
[기본값]
s3 =
    multipart_threshold = 64TB
```

이렇게 하면 플래시블레이드에서 더 큰 오브젝트 제한을 사용할 수 있습니다. v1.2.1에서, s5cm는 더 큰 오브젝트와 함께 CopyObject API를 사용하기 위해 이러한 조정이 필요하지 않습니다.

재해 복구 및 백업 솔루션

운영 성숙도가 높은 조직은 문제가 발생한 경우에도 운영을 지속할 수 있습니다. 기존에는 데이터 웨어하우스가 재해 복구의 최우선 순위가 아니었습니다. 하지만 조직들은 분석과 메트릭을 기반으로 더 많은 운영 관련 의사 결정을 내릴 수 있도록 진화해 왔습니다. 데이터 웨어하우스와 분석적인 데이터베이스에 대한 액세스가 점점 더 중요해지고 있습니다.

버티카에는 데이터베이스와 클러스터의 백업, 복구 및 복사를 위한 강력한 네이티브 툴(예: VBR 및 복사 클러스터)이 포함되어 있습니다. 퓨어스토리지 플래시블레이드를 사용하는 버티카 이온 모드에는 몇 가지 추가 옵션과 최적화가 포함되어 있습니다. 이러한 기능은 플래시블레이드의 네이티브 복제 기능과 성능을 활용합니다. 플래시블레이드는 기본적으로 두 플래시블레이드 간에 공용 스토리지로 사용되는 Amazon S3 버킷을 복제할 수 있습니다. 또한 버킷을 AWS S3에 복제할 수도 있습니다. 버킷내 데이터베이스에 대한 모든 업데이트는 소스에서 대상 버킷으로 지속적으로 전달됩니다. 또한 플래시블레이드는 초당 수 기가바이트의 백업 스트림을 쓰고 데이터를 훨씬 더 빠르게 복구할 수 있습니다. VBR을 위한 이상적인 백업 타겟이라고 할 수 있습니다.

이 섹션의 나머지 부분에서는 플래시블레이드의 역량과 버티카 이온 모드를 결합하는 재해 복구 및 백업 솔루션에 대해 설명합니다. 먼저 플래시블레이드 오브젝트 복제의 핵심 기능과 플래시블레이드와 VBR을 함께 사용하는 것에 대해 살펴보겠습니다. 그런 다음 버티카 데이터베이스의 장애 발생 시에도 일관된(crash-consistent) 복제본을 활용하는 재해 복구 솔루션의 일부로 플래시블레이드를 사용하는 방법을 살펴보겠습니다. 마지막으로 VBR과 플래시블레이드를 결합하여 로컬 및 원격 사이트에서 일관된 특정 시점 백업을 관리하는 방법을 알아보겠습니다.

플래시블레이드 오브젝트 복제

퓨리티//FB 3을 통해 플래시블레이드는 이제 고속 오브젝트 역량을 향상하고 클라우드의 이점과 고속 오브젝트의 잠재력을 활용하는 [비동기식 오브젝트 복제](#)를 제공합니다. 사용자는 플래시블레이드의 소스 버킷과 다른 플래시블레이드나 클라우드의 AWS S3의 대상 버킷 간에 복제 링크를 구성합니다.



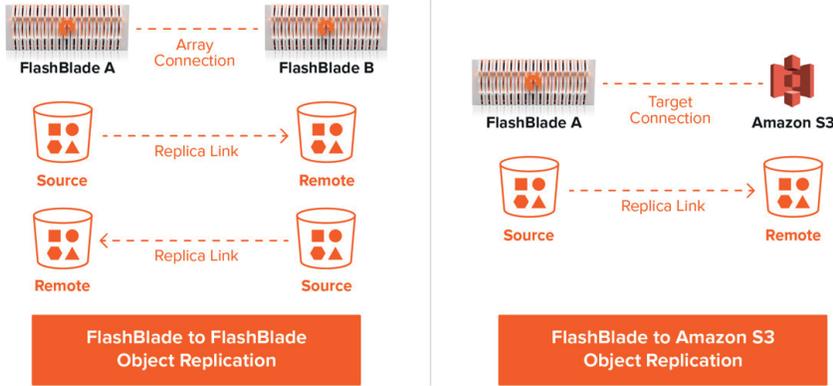


그림 8. 플래시블레이드 오브젝트 복제 옵션

일단 링크가 설정되면, 소스 버킷에 기록된 오브젝트가 대상 버킷에 빠르게 지속적으로 복제됩니다. 이는 데이터가 지속적으로 보호되고, 여러 위치에서 재해를 이겨내며, 클라우드에서 안전하게 사용될 수 있도록 해줍니다. 모든 플래시블레이드 기능과 마찬가지로, 이러한 기능은 제품의 일부로 제공되어 관리가 간편하며 라이선스나 특별한 게이트웨이가 필요하지 않습니다. 또한 플래시블레이드 복제 기능은 AI 중심 클라우드 기반 관리 플랫폼인 퓨어1(Pure1®)과 통합되어 전체 시스템의 모니터링과 관리를 간소화합니다.

VBR 타겟 및 소스로 사용되는 플래시블레이드

버티카에는 백업 및 복구 작업을 관리하기 위한 VBR이라는 강력한 유틸리티가 포함되어 있습니다. 최신 버전 버티카(10.0)에서 VBR은 다음을 지원합니다.

- 버티카 데이터베이스의 전체 백업 생성
- 백업에서 전체 데이터베이스 또는 개별 오브젝트 복원
- 데이터베이스에서 특정 오브젝트(예: 스키마 및 테이블) 백업(엔터프라이즈 모드만 해당)
- 데이터베이스를 다른 클러스터로 복사 예: 테스트 클러스터를 운영 환경으로 이동(엔터프라이즈 모드만 해당)
- 개별 오브젝트를 다른 클러스터로 복제(엔터프라이즈 모드만 해당)

사용 가능한 백업 목록 구성

VBR 유틸리티는 이온 및 엔터프라이즈 모드 데이터베이스를 Amazon S3 위치로 백업 및 복구할 수 있도록 지원합니다. 플래시블레이드는 이상적인 **백업 및 복구** 타겟이며 VBR과 원활하게 작동합니다. **구성은 간단합니다.** 먼저 VBR 유틸리티를 실행하는 셸에서 공용 스토리지 소스 및 백업 대상에 대한 엔드포인트와 크리덴셜을 정의하는 다양한 환경 변수를 설정합니다.

```
export VBR_COMMUNAL_STORAGE_ENDPOINT_URL=http://snfb-17-rr:80/
export VBR_COMMUNAL_STORAGE_ACCESS_KEY_ID="PS...JP"
export VBR_COMMUNAL_STORAGE_SECRET_ACCESS_KEY="84...b2/e8...DF"
export VBR_BACKUP_STORAGE_ENDPOINT_URL="http://snfb-11-rr:80/"
export VBR_BACKUP_STORAGE_ACCESS_KEY_ID="PS...EC"
export VBR_BACKUP_STORAGE_SECRET_ACCESS_KEY="0B...c0/af...CK"
```



그런 다음, Amazon S3 백업 위치 경로와 백업 작업 중에 잠금을 유지할 로컬 노드 경로를 포함하도록 VBR 구성 configuration.ini 파일을 편집합니다.

```
s3_backup_path = s3://sfo-bkup/prod/dw02-1T/
s3_backup_file_system_path = [ ]:/home/dbadmin/bac kup_locks_dir/
```

마지막으로, 첫 번째 백업 전에 VBR 유틸리티를 사용해 Amazon S3 위치를 초기화합니다. 그런 다음 백업 프로세스를 시작합니다.

```
vbr -t init -c eon_backup_restore.ini --s3-force-init
vbr -t backup -c eon_backup_restore.ini
```

퓨어스토리지 플래시블레이드를 위한 버티카 이온 모드를 위한 데이터 보호 워크플로우의 일부로 포함된 VBR에 대해서는 이 문서에서 이후에 자세히 설명하겠습니다.

재해 복구를 위한 플래시블레이드 복제

버티카 이온 모드는 충돌 시 일관성이 보장되는 데이터베이스 복제본을 Amazon S3 스토리지에 생성하여 재해 복구를 간소화합니다. 대부분의 오브젝트(데이터 및 카탈로그)는 쓰기가 한 번만 가능하며 업데이트되지 않습니다. 이전 오브젝트가 업데이트되는 경우는 거의 없습니다. 새로운 오브젝트는 통합(병합)될 수 있지만 이러한 작업은 데이터베이스의 일관성에 영향을 주지 않습니다. 그리고 recive_db 작업은 데이터베이스가 일관된 상태에 있는 가장 최근의 에포크(epoch)를 찾기 위해 자동으로 시간을 뒤로 이동합니다. 이러한 아키텍처 기능과 버티카의 기본 설정을 사용하면, 별다른 노력을 들이지 않고도 30분 미만의 복구 시점 목표(RPO)를 달성할 수 있습니다. 플래시블레이드의 기능과 함께 사용할 경우 이것이 어떤 모습인지 살펴보겠습니다.

로컬 장애 발생 시 일관성이 보장되는 재해 복구 복사본

경우에 따라 데이터베이스가 손상된 경우 신속한 복구에 사용할 수 있는 데이터베이스의 로컬 DR 복사본이 있는 것이 유용할 때가 있습니다. 플래시블레이드를 사용하면 빠르고 공간 효율적인 클론 생성 기술을 사용해 로컬 DR 복사본을 생성할 수 있습니다. 복사본은 공간 효율적이기 때문에 원본 데이터베이스 경로에서 삭제된 오브젝트만 DR 복사본의 공간을 차지합니다. DR 복사본이 주기적으로 재사용되는 경우, 삭제된 오브젝트는 결국 플래시블레이드에서 영구 제거됩니다. 데이터베이스는 revive_db 명령과 원본 소스에서 손실된 데이터를 다시 로드하여 빠르게 복구할 수 있습니다. 간단한 s5cmd 명령으로 클론을 생성할 수 있습니다.

```
s5cmd --endpoint-url=10.1.2.3 --log=error cp 's3://vertica/prod/dw01/*'
s3://vertica/snapshot/20200801142419/dw01/
```

또한 데이터베이스를 복제하기 전에 [데이터베이스의 일관성 보장 섹션](#)에 나와 있는 추가적인 vsq1 명령을 사용하는 것을 고려할 수 있습니다. DR에 클론이 더 이상 필요하지 않으면 s5cmd를 사용하여 오브젝트를 삭제합니다.

```
s5cmd --endpoint-url=10.1.2.3 --log=error rm 's3://vertica/snapshot/20200801142419/dw01/*'
```

이러한 작업을 간단한 cron 스크립트로 결합하여 최적의 스케줄에 따라 원하는 수의 클론을 유지할 수 있습니다. 데이터베이스를 복구해야 하는 경우 다음 단계를 따르십시오.



1. 현재(손상된) 데이터베이스를 중지하고 삭제합니다.
2. DR 클론 경로에서 운영 데이터베이스 경로로 오브젝트를 다시 복사합니다.
3. 운영 데이터베이스 경로에서 데이터베이스를 복구합니다.
4. 재개된 데이터베이스 에포크 이후 로드된 모든 데이터를 다시 로드합니다.

데이터베이스 크기와 변경률에 따라 이 워크플로우는 로컬 DR에 대한 다른 방법보다 훨씬 빠를 수 있습니다.

원격 장애 발생 시 일관성이 보장되는 재해 복구 복사본

데이터베이스의 로컬 복사본은 재해 복구 시나리오의 하위 집합에 적합합니다. 하지만 실제 사이트 수준의 재해라면 어떨까요? 재해가 반드시 화재, 홍수 또는 큰 지진일 필요는 없습니다. 몇 시간 동안 지속되는 지역의 정전이나 네트워크 중단도 운영에 심각한 영향을 미칠 수 있습니다. 이러한 시나리오에서 유일한 옵션은 원격 사이트에 인프라 및 데이터베이스의 복사본을 준비해두는 것입니다.

이 경우 플래시블레이드의 오브젝트 복제가 도움이 될 수 있습니다. 관리자는 2차 사이트에서 다른 버킷과 플래시블레이드 오브젝트의 복제 관계를 설정하여 버티카 데이터베이스를 재해로부터 보호할 수 있습니다. 그림 9는 ‘bos’라는 버킷을 다른 지역의 플래시블레이드에서 ‘bos-dr’이라는 버킷으로 복제하는 것을 보여 줍니다. 복제 관계가 설정되면 소스 버킷에 대한 모든 변경 내용이 대상 버킷에 복제됩니다.

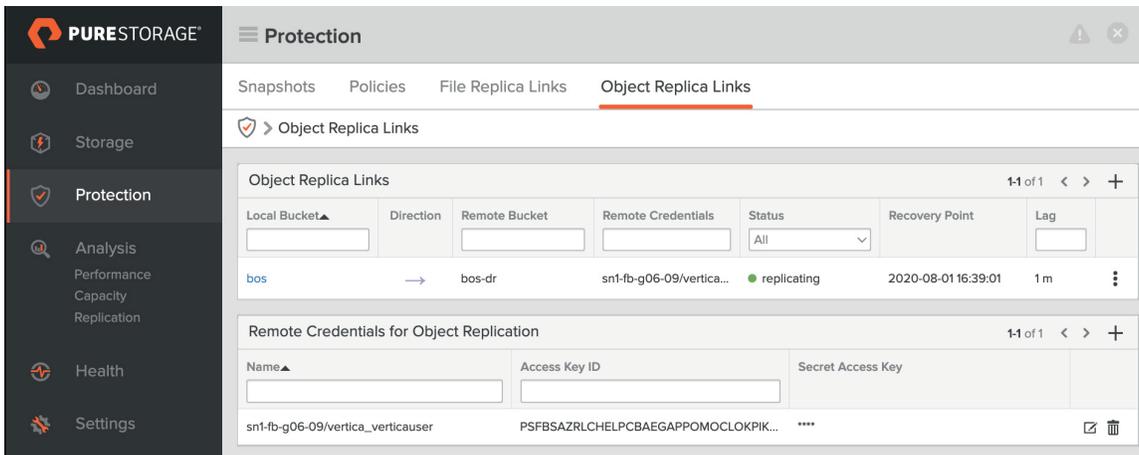


그림 9. 플래시블레이드 오브젝트 복제본 링크

연속적인 복제: 새 오브젝트가 몇 초 내에 감지되고 복제 스케줄이 정해집니다. 링크가 유효 상태인 경우 오브젝트는 즉시 복제됩니다. 복제 속도는 플래시블레이드 간의 네트워크가 지원할 수 있고 대상 플래시블레이드가 쓸 수 있는 속도가 됩니다. 구현에서는 일반적으로 초당 몇 기가바이트 규모의 데이터를 전송합니다. (그림 10)



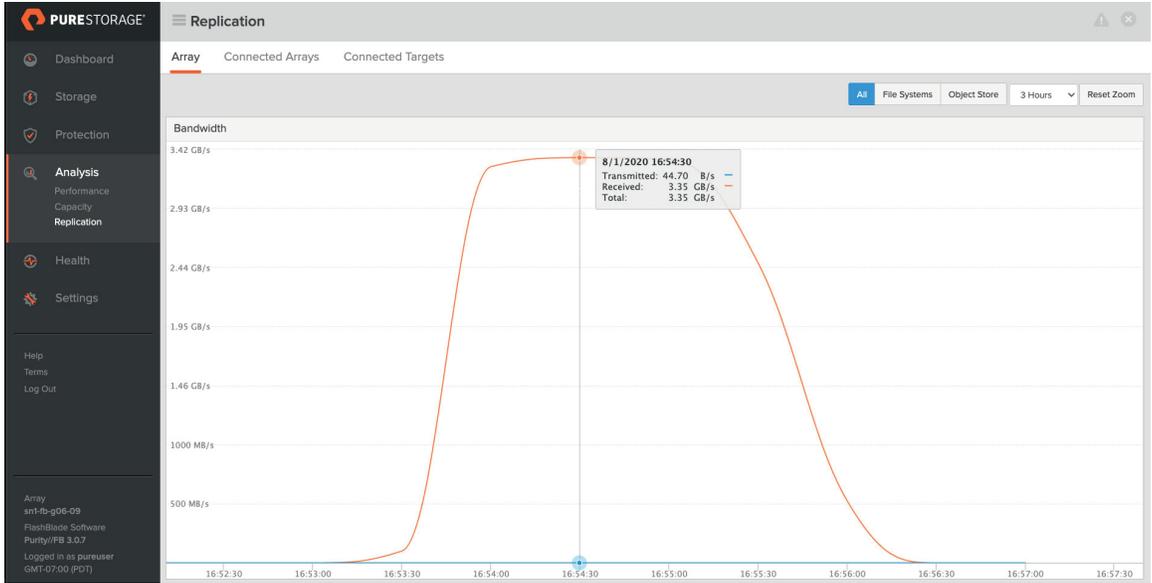


그림 10. 대상 플래시블레이드의 오브젝트 복제 쓰기 대역폭 예시

DR 버킷으로의 베이스라인(baseline) 복제

복제 링크는 새로 설정된 오브젝트와 수정된 오브젝트만 전송합니다. 데이터베이스를 생성하고 로드하기 전에 복제 링크를 설정하는 것이 가장 좋습니다. 그러나 기존 데이터베이스를 복제해야 하는 경우 퓨리티//FB3.0에는 수동 베이스라인(baseline) 복제가 필요합니다. 이 작업은 7단계 프로세스로 수행할 수 있습니다.

1. 소스 버킷에서 베이스라인을 정해야 하는 데이터의 오브젝트 수와 크기를 가져옵니다.

```
s5cmd --endpoint-url=10.1.2.3 du -H 's3://vertica/*'
```

2. 버킷의 콘텐츠를 소스 버킷 내의 특수 임시 접두사로 복제합니다. (임시 접두사 이름으로 "*" 문자를 사용하길 권합니다. ASCII 값이 낮아 모든 출력의 맨 위에 위치할 것이기 때문입니다. 그러나 셸이 이전 명령에 대한 참조로 해석하지 않도록 인용 부호를 써야 합니다.):

```
s5cmd --endpoint-url=10.1.2.3 --log=error cp 's3://vertica/*' 's3://vertica/!/'
```

3. 복제는 오브젝트 클론을 새로운 것으로 감지하여 이를 대상 버킷에 복제합니다. 복제가 완료될 때까지 기다립니다. s5cmd du 명령을 사용해 두 사이트에서 오브젝트 수와 크기가 동일한지 확인합니다.

```
s5cmd --endpoint-url=10.4.5.6 du -H 's3://vertica-dr/!/*'
```

4. 대상 버킷 내의 임시 접두사에 있는 오브젝트를 버킷의 루트로 다시 복제합니다. 이전 버전의 오브젝트가 새 버전을 덮어쓰지 않도록 s5cmd CP 를 위해 --no-clobber 옵션을 사용합니다.

```
s5cmd --endpoint-url=10.4.5.6 --log=error cp --no-clobber 's3://vertica-dr/!/*' s3://vertica-dr/
```

5. 소스 버킷의 임시 접두사를 삭제합니다.

```
s5cmd --endpoint-url=10.1.2.3 rm 's3://vertica/!/*'
```

6. 타겟 버킷에서 임시 접두사를 삭제합니다.

```
s5cmd --endpoint-url=10.4.5.6 rm 's3://vertica-dr/!/*'
```

7. 이제 두 버킷이 거의 동일인지 확인합니다. (아직 타겟으로 전달되지 않은 소스의 변경 사항이 있을 수 있습니다.)

```
s5cmd --endpoint-url=10.1.2.3 du -H 's3://vertica/*'
```



```
s5cmd --endpoint-url=10.4.5.6 du -H 's3://vertica-dr/*'
```

퓨어스토리지 지원팀이 데이터가 이미 포함된 버킷의 베이스라인 복제도 지원할 수 있습니다.

2차 사이트의 재해 복구 작업

데이터베이스 오브젝트가 2차 사이트에 성공적으로 복제되면 어떻게 사용할 수 있을까요? Fast CopyObject를 또다시 활용할 수 있습니다.

스냅샷 기반 파일 시스템 복제에 익숙한 사용자라면, 중요한 차이점이 있습니다. 대상 버킷은 쓰기가 가능합니다. 운영 데이터베이스가 업데이트를 보내는 버킷 접두사는 변경하지 않는 것이 좋습니다. 그러나 이 접두사를 동일한 버킷 내의 다른 접두사로 쉽게 복제할 수 있습니다. 그런 다음 해당 클론 접두사의 데이터베이스 복사본을 작업에 사용합니다. 다음과 같은 세 가지 시나리오가 유용합니다.

- 데이터베이스의 복제된 복사본이 일관성이 있는지, 또한 재해 복구에 사용할 수 있는지 여부를 검증합니다.
- 재해 발생 시 데이터베이스의 DR 복사본을 2차 사이트의 운영으로 전환합니다.
데이터베이스의 DR 복사본을 사용해 진행 중인 일부 작업을 2차 사이트로 오프로드합니다.

DR 데이터베이스 일관성 검증

여기서는 이전 섹션에서 설명한 것과 동일한 워크플로우를 따릅니다.

1. 트랜잭션 로그 동기화 서비스를 서둘러 소스 데이터베이스 측의 카탈로그를 동기화합니다
2. 플래시된 데이터를 복제하기 위해 소스 및 대상 플래시블레이드 간의 복제 지연이 발생할 때까지 기다립니다
3. 임시 접두사 경로에 DR 데이터베이스 클론을 생성합니다 (s5cmd cp)
4. 임시 경로에서 데이터베이스를 복원합니다 (admintools -t Revive_db)
5. 데이터베이스를 시작합니다 (admintools -t start_db)
6. 몇 가지 샘플 쿼리를 수행하여 제대로 작동하는지 확인합니다 (vsq1)
7. 데이터베이스를 중지합니다 (admintools -t stop_db)
8. 데이터베이스를 삭제합니다 (admintools -t drop_db)
9. 임시 접두사 경로에서 복제된 오브젝트를 제거합니다 (s5cmd rm)

DR 데이터베이스를 운영 환경으로 이동

이 워크플로우는 검증 워크플로우와 유사하게 시작되지만 무기한 작업으로 전환됩니다.

1. 복제에 사용되는 대상 버킷 액세스 키를 해제하여 소스 플래시블레이드에서 업데이트를 방지합니다
2. DR 데이터베이스를 활성화된 접두사 경로로 복제합니다 (s5cmd cp)
3. 활성화된 경로로부터 데이터베이스를 복원합니다 (admintools -t Revive_db)
4. 데이터베이스를 시작합니다 (admintools -t start_db)
5. 몇 가지 샘플 쿼리를 수행하여 제대로 작동하는지 확인합니다 (vsq1)
6. 로드 밸런서와 DNS 서버를 2차 사이트로 리디렉션합니다
7. 1차 사이트로 페일백(failback) 준비가 될 때까지 2차 사이트에서 주요 운영 작업을 수행합니다



DR 복제를 사용한 작업 오프로드

앞서 언급한 것처럼 2차 사이트의 모든 인프라를 사용하는 강력한 방법 중 하나는 1차 사이트 운영 작업의 일부를 오프로드하는 것입니다. 예를 들어, 정상적인 운영 중에 2차 사이트를 개발 및 QA에 사용하거나, 최신 업데이트가 필요하지 않은 덜 중요한 쿼리와 보고서를 실행하는 경우입니다.

워크플로우는 유사합니다. 한 가지 예외는 여러 엔지니어들을 위해 데이터베이스의 독립된 복사본을 여러 개 원하는 경우입니다. 이런 경우, 먼저 데이터베이스의 DR 복제본을 템플릿 접두사에 한 번 복제합니다. 그런 다음 필요에 따라 템플릿을 수정할 수 있습니다 (예: 익명 처리 또는 수정). 마지막으로, 템플릿을 반복적으로 복제하여 데이터베이스의 독립된 복사본을 여러 개 만들 수 있습니다.

1차 사이트로 파일백

재해가 해결되면 복제 관계를 되돌리고 2차 사이트에서 1차 사이트로 최신 버전의 데이터베이스를 복사하여 1차 사이트의 운영을 재개할 수 있습니다. 이 접근 방식은 간단하고 신뢰할 수 있습니다. 퓨리티//FB 3.0에서 이는 지원되는 주요 파일백 메커니즘입니다.

단점은 복제할 데이터가 매우 많을 수 있다는 것입니다. 재해로 인해 데이터가 완전히 손상된 것이 아니라 데이터의 대부분이 1차 사이트에 여전히 존재하는 경우, 1차 사이트에서 운영을 재개하기 위해 보다 더 효율적인 방법이 있을 수 있습니다. 파일오버가 발생한 이후에 생성되거나 덮어쓴 오브젝트를 확인한 다음, 1차 사이트의 올바른 경로에 이를 복제하는 것입니다. DR 경로가 복원되기 전에 클론된 경우, 2차 사이트에는 파일오버 시 오브젝트가 존재한 로컬 레코드가 있을 것입니다. 그러나 이러한 접근 방식을 구현하는 방법은 이 백서에서 다루지 않습니다.

버티카 VBR 백업 및 복원

VBR 타겟 및 소스로서의 플래시블레이드 섹션에서 설명한 것처럼, VBR 유틸리티는 버티카 데이터베이스를 백업하고 복원하는 강력한 메커니즘을 제공합니다. VBR은 Amazon S3 엔드포인트를 백업 대상 및 소스로 사용하는 것이 가능합니다. 또한 엔터프라이즈 및 이온 모드 데이터베이스 두 가지 모두와 작동합니다. 플래시블레이드와 함께 사용하면, VBR은 버티카 데이터베이스의 보호 및 복구 워크플로우를 가속화할 수 있습니다.

대부분의 가이드라인과 절차는 [버티카 문서](#)나 위 섹션에 설명되어 있습니다. 이 섹션의 나머지 부분에서는 퓨어스토리지의 플래시블레이드 구현을 위해 버티카 이온 모드의 일부로 VBR을 사용하는 방법에 대해 간단히 설명합니다.

로컬 VBR 백업

가장 간단한 백업 옵션은 로컬 플래시블레이드에서 데이터베이스의 VBR 백업을 사용하는 것입니다. 이렇게 하면 특정 버티카 오브젝트(예: 테이블 및 스키마)가 손상되어 데이터베이스 롤백이 의미가 없는 경우 쉽게 복원할 수 있습니다. VBR 백업에 대해 별도의 버킷을 생성하는 것을 추천합니다. 그런 다음 이 버킷을 재해 복구 전략의 일부로 다른 지역의 플래시블레이드에 복제할 수 있습니다. 보다 포괄적인 백업 및 아카이빙 전략의 일환으로 AWS S3에 복제할 수도 있습니다. 로컬 VBR 복사본의 이점은 다음과 같습니다.

- 로컬 백업은 WAN 대역폭에 의존하지 않으며 보다 신속하게 완료할 수 있습니다.
- VBR 스냅샷은 마지막 백업 이후 변경된 정보만 백업합니다.
- 완료된 백업은 항상 일관성이 있으며 오브젝트 수준으로 복구됩니다.
- VBR 대상 버킷이 플래시블레이드 복제의 소스인 경우, 복제가 더 효율적일 수 있습니다. 버킷은 일관된 형식으로 복제되며 나중에 삭제 또는 병합될 임시 오브젝트는 복제되지 않습니다.



4노드 클러스터 및 1TB 데이터베이스를 사용하여 VBR 백업과 복원 속도를 테스트했습니다. (표 4) 플래시블레이드는 클러스터 노드와 마찬가지로 비교적 유향 상태였습니다. 대부분의 시간은 VBR 유틸리티 내에서 프로세스를 동기화하는 데 사용했습니다. 기본 설정인 10개의 스트림은 백업 및 복원 모두에서 최상의 성능을 발휘했습니다.

스트림	지속시간(초)		처리량			
	백업	복원	백업 MB/초	백업 오브젝트/초	복원 MB/초	복원 오브젝트/초
1	1126.5	1104.2	331.2	34.9	337.9	35.6
4	647.4	609	576.3	60.7	612.6	64.6
10 (기본값)	614.4	601.1	607.2	64.0	620.7	65.4
64	629.7	615	592.5	62.4	606.6	63.9
128	640	623.4	582.9	61.4	598.5	63.1

표 4. VBR 백업 및 복원 속도

앞서 언급한 것처럼, 로컬 VBR 백업 대신 플래시블레이드 Fast CopyObject를 사용하여 공간 효율적인 데이터베이스 클론을 신속하게 생성할 수 있습니다. 이 방식은 계속 증가하는 VBR 백업보다 더 빠르고 공간 효율적입니다. 대량의 데이터를 복원해야 하는 경우 과거의 클론 경로 중 하나에서 데이터베이스를 복원하는 것이 더 빠를 수 있습니다.

그러나 몇 가지 단점은 있습니다.

- 특정 버티카 오브젝트만 복구하는 간단한 메커니즘이 없기 때문에, 전체 데이터베이스를 복구하는 것이 효율적입니다.
- 장애 시 일관성이 보장되는 데이터베이스의 클론은 일관된 에포크를 찾기 위해 더 많은 롤백을 진행해야 할 수도 있습니다.
- 복사본은 데이터베이스와 동일한 버킷에 있어야 하므로 복제가 복잡해집니다. 복제된 오브젝트는 공간 효율적이지 않으며 대상에서 대역폭과 용량을 모두 사용합니다.

데이터 보호 전략을 선택할 때는 두 가지 접근 방식의 장단점을 고려해야 합니다.

원격 VBR 백업

원격지에 VBR 백업을 저장하는 것은 전체 데이터 보호 및 재해 복구 전략의 일부가 될 수 있습니다. 앞에서 두 가지 옵션, 즉 로컬 플래시블레이드 VBR 버킷을 다른 플래시블레이드 또는 AWS S3로 복제하는 것을 언급했습니다. 세 번째 옵션은 원격 플래시블레이드 어레이를 VBR.ini 구성 파일에서 타겟으로 지정하는 것입니다.

- 구성 프로세스는 로컬 플래시블레이드를 사용할 때와 동일합니다.
- 사이트 간 네트워크는 HTTP/HTTPS 요청을 원격 위치로 라우팅할 수 있어야 합니다.
- QoS 대역폭 제한에는 백업 윈도우와 데이터 변경률이 포함되어야 합니다.
- 네트워크에 HTTP 프록시가 있는 경우, 프록시를 피하기 위해 플래시블레이드를 허용 목록에 추가해야 할 수 있습니다.

VBR 백업에서 데이터베이스를 복원하는 경우 버티카의 요구 사항을 염두에 두어야 합니다. 원격 사이트에서 복구하는 것이 목표라면(특히 재해 발생 시 빠르게), 복구할 대상 데이터베이스와 클러스터를 지원하기 위해 일부 자동화가 필요할 수 있습니다.



버티카 데이터 보호 및 오프로드 모범 사례

- 버킷에서 추가 접두사 경로를 사용하여 이온 모드 데이터베이스를 생성합니다(예: s3://vertica/dw01/ 대신 s3://vertica/prod/dw01/) 그러면 Fast CopyObject를 사용하여 데이터베이스를 추가 접두사(예: test, dev 등)로 간단히 복제할 수 있습니다.
- VBR 백업과 복제 타겟에 별도의 버킷을 사용합니다. (예: s3://bos-dr/prod/dw01/ 또는 s3://oak-bkup/prod/dw02/) 그러면 필요한 정보만 복제하거나 복제본 버킷에서 데이터베이스의 클론을 생성하는 것이 더 간단해집니다.
- 복제 타겟 계정에 복제할 별도의 사용자를 생성하고 그 사용자의 키를 사용하여 복제 관계를 설정합니다. 재해가 발생할 경우, 대상 사이트에서 키를 (일시적으로) 비활성화하여 1차 사이트가 업데이트되는 것을 방지할 수 있습니다. 이는 소스 플래시블레이드가 대상 사이트와 간헐적으로 통신할 수 있는 경우에 중요합니다.

자동화 예시

이 문서에서는 Ansible을 사용하여 이러한 작업의 많은 부분을 자동화하는 것에 대해 언급했습니다. GitHub의 [퓨어스토리지 오픈커넥트 스페이스\(PureStorage-OpenConnect space\)](#)에 [버티카 저장소\(Vertica Repository\)](#)를 만들어 코드 스니펫과 아이디어를 복제, 수정 또는 살펴볼 수 있는 Ansible playbook 사례를 추가해 두었습니다. 이 playbook은 플래시블레이드와의 상호 작용을 자동화하여, 버티카 이온 모드 인프라를 완벽하게 자동화하는 데 도움이 될 수 있습니다.

두 가지 주요 하위 섹션이 있습니다.

1. 버티카 PoC([vertica-poc](#)) - 데모 및 테스트용 버티카 PoC 환경을 신속하게 구축하기 위한 코드입니다. 주로 Ansible playbook과 일부 셋업 배시 스크립트(setup bash script)로 구성됩니다.
2. 버티카 클로닝([vertica-cloning](#)) - 테스트/개발 및 DR 검증 시나리오를 위해 버티카 데이터베이스 복제와 관련된 코드가 담긴 Ansible playbook의 예입니다.

코드 사용 방법은 각 하위 섹션의 자세한 README 파일을 참조하십시오.

요약

버티카 분석 플랫폼은 고성능 대규모 SQL 분석을 위한 강력한 툴입니다. 이온 모드는 플래시블레이드 스토리지와 같은 공유 고속 오브젝트 스토리지를 활용하는 버티카 데이터베이스를 위한 새로운 스토리지 구현 모델입니다. 버티카 이온 모드는 데이터베이스 성능을 유지(혹은 개선)하면서 운영 효율성을 크게 향상시킵니다. 퓨어스토리지 플래시블레이드 시스템에 버티카 이온 모드 데이터베이스를 구현해 실행하면 클라우드와 유사한 간소함과 유연성을 제공하는 동시에, 보다 안전한 프라이빗 클라우드 환경에서 예측 가능한 비용으로 사용할 수 있습니다.

버티카에는 데이터베이스와 클러스터의 백업, 복구 및 복사를 위한 강력한 네이티브 툴(예: VBR 및 복사 클러스터)들이 포함됩니다. 퓨어스토리지 플래시블레이드를 사용하는 버티카 이온 모드에는 추가 옵션과 최적화도 포함되어 있습니다. 퓨리티//FB3 버전부터는, 플래시블레이드 Amazon S3 버킷을 다른 플래시블레이드 또는 AWS S3에 복제할 수 있습니다. 버킷의 데이터베이스에 대한 모든 업데이트는 소스에서 타겟 버킷으로 지속적으로 전달됩니다. 또한 플래시블레이드는 초당 수 기가바이트의 백업 스트림을 쓰고 데이터를 훨씬 더 빠르게 복구할 수 있습니다. VBR을 위한 이상적인 백업 타겟이라 할 수 있습니다.

또한 페타바이트 규모의 버티카 이온 데이터베이스의 공간 효율적인 클론을 몇 분 내에 신속하게 생성할 수 있습니다. 이러한 데이터 웨어하우스 클론은 최신 데이터에서 새로운 코드를 개발하고 테스트하는 데 유용합니다. 클론을 사용하면 운영 환경에 영향을 미치지 않고 실제 데이터베이스의 콘텐츠에 대한 스키마나 데이터의 변환을 테스트할 수 있습니다. 또한 클론을 데이터베이스의 DR 복사본을 액티브 복사본으로



변환하여 추가 쿼리를 2차 위치로 오프로드하는 데 사용할 수 있습니다. 재해 복구 복사본을 무효로 만들 수 있는 혹시 모를 변수에 대해서도 걱정할 필요가 없습니다.

참고자료

- [버티카 문서](#)
- [솔루션 브리프: 퓨어스토리지와 버티카](#)
- [퓨어스토리지와 버티카 파트너십](#)
- [블로그 게시물: '최신 분석에 현대적인 인프라가 필요한 이유'](#)



부록: 플래시블레이드 스토리지를 통한 버티카 구현의 모범 사례

다음과 같은 모범 사례를 통해 이미 앞에서 설명한 퓨어스토리지 플래시블레이드 버티카 이온 모드의 장점을 이해하고 이를 성공적으로 구현할 수 있습니다.

1. Fast CopyObject 작업을 간소화하려면 버킷에서 접두사 명명을 사용합니다. 버킷의 루트에 기존 데이터베이스가 있는 경우, 중지된 데이터베이스를 접두사로 복제하고 새로운 경로에서 복원하는 것이 좋습니다.

2. 플래시블레이드에는 필요하지 않으므로 AWS Streaming을 비활성화합니다. (vsq1 프롬프트 사용)

```
select set_config_parameter('AWSStreamingConnection Percentage', 0);
```

3. 로컬 디포의 읽기 설정을 활성화(기본값) 상태로 유지합니다. 작업 집합이 로컬 캐시와 맞지 않는 큰 데이터 집합이 있는 경우, 디포의 읽기 기능을 비활성화하면 보다 간단한 작업으로 유사한 성능을 얻을 수도 있습니다. 어떤 시점에서는 디포 읽기 설정을 테스트하여 구현을 위한 이상적인 값을 vsq1 프롬프트를 통해 결정하는 것이 좋습니다.

```
select set_config_parameter('UseDepotForReads', 1);
```

4. 데이터베이스 ETL 작업을 향상시키므로 (vsq1 프롬프트를 통해) 로컬 디포 쓰기를 비활성화합니다.

```
select set_config_parameter('UseDepotForWrites', 0);
```

5. 버티카에서 서버별 TCP 연결을 (vsq1 프롬프트를 통해) 128개 연결로 조정합니다.

```
select set_config_parameter('AWSConnectionPoolSize', 128);
```

©2021 Pure Storage, the Pure P Logo, 퓨어의 등록 상표 목록(<https://www.purestorage.com/legal/productenduserinfo.html>)에 포함된 마크는 Pure Storage, Inc.의 등록상표입니다. 기타 모든 상표는 각 해당 소유권자의 재산입니다. 퓨어스토리지 제품 및 프로그램의 사용은 <https://www.purestorage.com/legal/productenduserinfo.html> 및 <https://www.purestorage.com/patents>에서 제공되는 엔드 유저 계약, IP 및 기타 약관의 적용을 받습니다.

이 문서에 설명된 퓨어스토리지 제품과 프로그램들은 제품의 사용, 복사, 배포 및 역컴파일/역엔지니어링을 제한하는 라이선스 계약 하에 배포됩니다. 이 문서의 어떠한 부분도 퓨어스토리지의 사전 서면 허가 없이 어떠한 형식이나 방법으로든 복제될 수 없습니다. 퓨어스토리지의 사전 통지 없이 언제든지 퓨어스토리지 제품 및/또는 본 문서에 설명된 프로그램을 개선 및/또는 변경할 수 있습니다.

이 문서는 '있는 그대로' 제공되며, 퓨어스토리지는 법적으로 허용된 범위 내에서 상품성, 특수 목적을 위한 적합성 또는 비침해성에 대한 보증은 물론 그 어떠한 명시적, 묵시적, 서면, 구술 또는 법적 보증을 부인합니다. 퓨어스토리지는 이 문서의 이용, 공급 또는 성과와 관련하여 발생하는 모든 우발적 또는 결과적 손해에 대해 어떠한 경우에도 책임을 지지 않습니다. 이 문서에 포함된 정보는 예고 없이 변경될 수 있습니다.

Pure Storage, Inc.
650 Castro Street, #400
Mountain View, CA 94041