

vSphere 리소스 관리

업데이트 3

VMware vSphere 8.0

VMware ESXi 8.0

vCenter 8.0

VMware by Broadcom 웹 사이트

<https://docs.vmware.com/kr>에서 최신 기술 문서를 찾을 수 있습니다.

VMware by Broadcom

3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2006~2024 Broadcom. All Rights Reserved. “Broadcom”은 Broadcom Inc. 및/또는 해당 자회사를 뜻합니다. 자세한 내용은 <https://www.broadcom.com> 페이지를 참조하십시오. 여기에서 언급된 모든 상표, 상호, 서비스 마크 및 로고는 해당 회사의 소유입니다.

목차

vSphere 리소스 관리	11
1 vSphere 리소스 관리 시작	12
ESXi 리소스 유형	12
물리적 리소스 사용	12
리소스 활용	13
vSphere를 통한 리소스 관리의 목표	13
2 vSphere 리소스 할당 설정 구성	14
리소스 할당 공유	14
리소스 할당 예약	15
리소스 할당 제한	16
리소스 할당 설정 제안	16
vSphere 설정 편집	16
리소스 할당 설정 변경 - 예	17
vSphere 승인 제어	18
3 vSphere의 CPU 가상화	19
소프트웨어 기반 CPU 가상화	19
하드웨어 지원 CPU 가상화	20
가상화 및 프로세서별 동작	20
CPU 가상화가 성능에 미치는 영향	20
4 vSphere CPU 리소스 관리	21
프로세서 정보 보기	21
CPU 구성 지정	21
다중 코어 프로세서	22
5 vSphere를 사용한 하이퍼스레딩	23
하이퍼스레딩 및 ESXi 호스트	23
하이퍼스레딩 사용	24
6 vSphere CPU 선호도 사용	25
특정 프로세서에 가상 시스템 할당	25
CPU 선호도의 잠재적인 문제	26
7 ESXi 호스트 전원 관리 정책	27

- CPU 전원 관리 정책 선택 27
- 호스트 전원을 위한 사용자 지정 정책 매개 변수 구성 28

8 vSphere를 통한 메모리 가상화 30

- 가상 시스템 메모리 30
- 메모리 오버 커밋 31
- 메모리 공유 32
- 메모리 가상화 32
- 하드웨어 지원 메모리 가상화 33
- 큰 페이지 크기 지원 34

9 vSphere를 사용하여 메모리 리소스 관리 35

- 메모리 오버헤드 이해 36
- 가상 시스템의 오버헤드 메모리 36
- ESXi 호스트의 메모리 할당 방법 37
- 유휴 가상 시스템의 메모리 세울 37
- VMX 스왑 파일 38
- 메모리 회수 38
- 메모리 벌룬 드라이버 38
- 가상 시스템 사이에 메모리 공유 39
- 메모리 압축 40
- 메모리 압축 캐시 활성화 또는 비활성화 40
- 메모리 압축 캐시의 최대 크기 설정 40
- 메모리 사용 유형 측정 및 차별화 41
- 메모리 안정성 43
- 오류 분리 알림 수정 43
- 시스템 스왑 43
- 시스템 스왑 구성 44

10 vSphere 스왑 파일 사용 45

- 스왑 파일 위치 45
- DRS 클러스터에 대해 호스트-로컬 스왑 사용 46
- 독립 실행형 호스트에 대해 호스트-로컬 스왑 사용 46
- 스왑 공간 및 메모리 오버 커밋 47
- 호스트에 대한 가상 시스템 스왑 파일 속성 구성 48
- 클러스터의 가상 시스템 스왑 파일 위치 구성 49
- 스왑 파일 삭제 49

11 vSphere에서 영구 메모리 사용 50

- PMem VM용 vSphere HA 구성 52

- vSphere HA 승인 제어 PMem 예약 53
- vSphere 메모리 모니터링 및 업데이트 적용 54

12 vSphere에서 가상 그래픽 구성 56

- GPU 통계 보기 56
- 가상 시스템에 NVIDIA GRID vGPU 추가 57
- 호스트 그래픽 구성 57
- 그래픽 디바이스 구성 58
- vGPU가 있는 VM 마이그레이션 59
- vGPU 크기 구성 59

13 vSphere를 사용하여 스토리지 I/O 리소스 관리 61

- 가상 시스템 스토리지 정책 62
- I/O 필터 62
- Storage I/O Control 요구 사항 63
- Storage I/O Control 리소스 공유 및 제한 63
- Storage I/O Control 공유 및 제한 보기 63
- Storage I/O Control 공유 모니터링 64
- Storage I/O Control 리소스 공유 및 제한 설정 64
- Storage I/O Control 사용 65
- Storage I/O Control 임계값 설정 66
- 스토리지 프로파일과 Storage DRS 통합 67

14 vSphere를 사용하여 리소스 풀 관리 69

- 리소스 풀을 사용하는 원인은 무엇입니까? 70
- 리소스 풀 생성 72
- 리소스 풀 편집 73
- 리소스 풀에 가상 시스템 추가 74
- 리소스 풀에서 가상 시스템 제거 75
- 리소스 풀 제거 75
- 리소스 풀 승인 제어 75
- 확장 가능한 예약 예제 1 76
- 확장 가능한 예약 예제 2 76

15 vSphere 클러스터 서비스 78

- 내장된 vCLS 79
- vSphere DRS 및 vCLS VM 82
- 외부 vCLS에 대한 데이터스토어 선택 82
- 외부 vCLS 데이터스토어 배치 83
- vSphere 클러스터 서비스 모니터링 83

- vSphere 클러스터 서비스의 상태 유지 84
- 클러스터를 철회 모드로 전환 85
- 외부 vCLS의 암호 검색 86
- vCLS VM 반선택도 정책 86
- vCLS VM 반선택도 정책 생성 또는 삭제 86

16 vSphere DRS 클러스터 생성 88

- 승인 제어 및 초기 배치 89
- 단일 가상 시스템 전원 켜기 89
- 그룹 전원 켜기 89
- 가상 시스템 마이그레이션 90
- DRS 마이그레이션 임계값 91
- 마이그레이션 권장 사항 92
- DRS 클러스터 요구 사항 93
- 공유 스토리지 요구 사항 93
- 공유 VMFS 볼륨 요구 사항 93
- 프로세서 호환성 요구 사항 93
- DRS 클러스터에 대한 vMotion 요구 사항 94
- 가상 플래시로 DRS 구성 94
- 클러스터 생성 95
- 클러스터 설정 편집 96
- 가상 시스템에 대한 사용자 지정 자동화 수준 설정 98
- DRS 비활성화 99
- 리소스 풀 트리 복원 100
- vSAN 확장된 클러스터의 DRS 인식 100
- vGPU DRS 배치 101
- VM용 DRS 오버헤드 메모리 관리 101

17 ROBO Enterprise 라이선스를 사용한 vSphere DRS 유지 보수 모드 기능 103

- ROBO Enterprise 라이선스로 제공되는 DRS 유지 보수 모드의 제한 사항 103
- ROBO Enterprise 라이선스로 DRS 유지 보수 모드 사용 103
- ROBO Enterprise 라이선스로 DRS 유지 보수 모드 문제 해결 104

18 DRS 클러스터를 사용하여 vSphere로 리소스 관리 106

- 클러스터에 호스트 추가 107
 - 클러스터에 관리 호스트 추가 107
 - 클러스터에 관리되지 않는 호스트 추가 108
- 클러스터에 가상 시스템 추가 108
 - 가상 시스템을 클러스터로 이동 109
- 클러스터에서 가상 시스템 제거 109

가상 시스템을 클러스터 외부로 이동	109
클러스터에서 호스트 제거	110
유지 보수 모드로 호스트 전환	110
클러스터에서 호스트 제거	111
대기 모드 사용	112
VM BestEffortRestart 정책 구성	112
DRS 클러스터 유효성	113
유효한 DRS 클러스터	114
오버 커밋된 DRS 클러스터	116
잘못된 DRS 클러스터	117
전원 리소스 관리	118
vSphere DPM의 IPMI 또는 iLO 설정 구성	118
vSphere DPM을 위한 Wake-on-LAN 테스트	119
DRS 클러스터에 vSphere DPM 활성화	120
vSphere DPM 모니터링	122
VM 정적 전원	122
vSphere DRS와 선호도 규칙 사용	123
vSphere DRS의 VM-VM 선호도 규칙	124
vSphere DRS의 VM-호스트 선호도 규칙	124
VM-호스트 그룹 생성	124
가상 시스템 그룹 생성	125
VM-VM 선호도 규칙 생성	125
VM-VM 선호도 규칙 충돌	126
VM 호스트 선호도 규칙 생성	126
VM-호스트 선호도 규칙 사용	127
vSphere DRS 없이 선호도 규칙 사용	128
19 vSphere 데이터스토어 클러스터 생성	130
초기 배치 및 지속적인 균형 조정	131
스토리지 마이그레이션 권장 사항	131
데이터스토어 클러스터 생성	132
Storage DRS 활성화 및 비활성화	132
데이터스토어 클러스터의 자동화 수준 설정	133
Storage DRS의 강도 수준 설정	133
Storage DRS 런타임 규칙 설정	134
데이터스토어 클러스터 요구 사항	135
데이터스토어 클러스터에서 데이터스토어 추가 및 제거	136
20 데이터스토어 클러스터를 사용하여 vSphere 스토리지 리소스 관리	137
Storage DRS 유지 보수 모드 사용	137

- 유지 보수 모드로 데이터스토어 전환 137
- 유지 보수 모드에 대해 Storage DRS 선호도 규칙 무시 138
- Storage DRS 권장 사항 적용 139
 - Storage DRS 권장 사항 새로 고침 139
- 가상 시스템에 대해 Storage DRS 자동화 수준 변경 140
- Storage DRS에 대한 근무 외 시간 스케줄링 설정 140
- Storage DRS 반선택도 규칙 142
 - VM 반선택도 규칙 생성 143
 - VMDK 반선택도 규칙 생성 143
 - VMDK 선호도 규칙 재정의 144
- Storage DRS 통계 지우기 145
- 데이터스토어 클러스터와의 Storage vMotion 호환성 146

21 ESXi와 함께 NUMA 시스템 사용 147

- NUMA란? 147
- 운영 체제 문제점 148
- ESXi NUMA 스케줄링 작동 방식 148
- VMware NUMA 최적화 알고리즘 및 설정 149
- 홈 노드 및 초기 배치 149
- 동적 로드 밸런싱 및 페이지 마이그레이션 150
- NUMA용으로 최적화된 투명 페이지 공유 151
- NUMA 아키텍처의 리소스 관리 151
- 가상 NUMA 사용 151
- ESXi 8.0의 가상 토폴로지 152
- 가상 NUMA 제어 153
- NUMA 제어 지정 154
- 특정 프로세서에 가상 시스템 연결 155
- 메모리 선호도를 사용하여 특정 NUMA 노드와 메모리 할당 연결 155
- 가상 시스템을 지정된 NUMA 노드에 연결 156

22 vSphere의 고급 특성 158

- 고급 호스트 특성 설정 158
 - 고급 메모리 특성 159
 - 고급 NUMA 특성 160
- 고급 가상 시스템 특성 설정 161
 - 고급 가상 시스템 특성 161
 - 고급 가상 NUMA 특성 162
- 지연 시간 감도 163
 - 지연 시간 감도 조정 163
- VM에 대한 가상 하이퍼스레딩 지원 164

- vHT 전체 CPU 예약 164
- VM에 대해 vHT 활성화 165
- 신뢰할 수 있는 메모리 165
 - 신뢰할 수 있는 메모리 보기 166
- 게스트 vRAM에 1GB 페이지 사용 166

23 장애 정의 167

- 가상 시스템이 고정되어 있음 168
- 가상 시스템이 다른 호스트와 호환되지 않음 168
- 다른 호스트로 이동할 때 VM/VM DRS 규칙 위반 168
- 호스트가 가상 시스템과 호환되지 않음 168
- 호스트의 가상 시스템에서 VM/VM DRS 규칙을 위반함 168
- 호스트에 가상 시스템을 위한 용량이 충분하지 않음 169
- 잘못된 상태의 호스트 169
- 호스트에 가상 시스템을 위한 물리적 CPU 수가 충분하지 않음 169
- 호스트에 각 가상 시스템 CPU를 위한 용량이 충분하지 않음 169
- 가상 시스템이 vMotion에 있음 169
- 클러스터에 활성화된 호스트 없음 169
- 불충분한 리소스 169
- HA용으로 구성된 페일오버 수준을 만족하기 위한 리소스가 충분하지 않음 169
- 호환되는 하드 선호도 호스트가 없음 170
- 호환되는 소프트 선호도 호스트가 없음 170
- 소프트 규칙 위반 수정 허용되지 않음 170
- 소프트 규칙 위반 수정 영향 170

24 DRS 문제 해결 정보 171

- 클러스터 문제 171
 - 클러스터의 로드 불균형 171
 - 클러스터가 노란색으로 표시됨 172
 - 일치하지 않는 리소스 풀로 인해 클러스터가 빨간색으로 표시됨 172
 - 페일오버 용량 위반으로 인해 클러스터가 빨간색으로 표시됨 173
 - 전체 클러스터 로드가 낮을 때는 호스트의 전원이 꺼지지 않음 173
 - 전체 클러스터 로드가 높을 때 호스트의 전원이 꺼짐 174
 - DRS가 거의 또는 전혀 vMotion 마이그레이션을 수행하지 않음 174
- 호스트 문제 175
 - DRS는 전체 클러스터 로드가 낮을 때에는 용량을 증가시키기 위해 호스트의 전원이 켜져 있도록 권장함 175
 - 전체 클러스터 로드가 높음 175
 - 전체 클러스터 로드가 낮음 176
 - DRS가 유지 보수 모드나 대기 모드로 전환하도록 요청한 호스트를 제거하지 않음 177

- DRS가 가상 시스템을 호스트로 이동하지 않음 177
- DRS가 가상 시스템을 호스트에서 이동하지 않음 178
- VM과 해당 호스트 간의 비호환성으로 인해 DRS에서 시작된 VM 마이그레이션 178
- 가상 시스템 문제 180
 - CPU 또는 메모리 리소스 부족 180
 - VM/VM DRS 규칙 또는 VM/호스트 DRS 규칙 위반 181
 - 가상 시스템의 전원 켜기 작업 실패 181
 - DRS가 가상 시스템을 이동하지 않음 182

vSphere 리소스 관리

vSphere 리소스 관리에서는 VMware® ESXi 및 vCenter® Server 환경의 리소스 관리에 대해 설명합니다. 이 설명서에서는 다음 항목에 대해 중점적으로 다룹니다.

- 리소스 할당 및 리소스 관리 개념
- 가상 시스템 특성 및 승인 제어
- 리소스 풀 및 리소스 풀의 관리 방법
- 클러스터, vSphere® DRS(Distributed Resource Scheduler), vSphere DPM(Distributed Power Management) 및 이를 사용한 작업 방법
- 데이터스토어 클러스터, Storage DRS, Storage I/O Control 및 이를 사용한 작업 방법
- 고급 리소스 관리 옵션
- 성능 고려 사항

VMware는 포용성을 중요하게 생각합니다. 고객, 파트너 및 내부 커뮤니티 안에서 이러한 원칙을 강화하기 위해 포용성 있는 언어를 사용하여 콘텐츠를 만듭니다.

대상 사용자

이 정보는 시스템의 리소스 관리 방법 및 기본 동작을 사용자 지정하는 방법을 이해하고자 하는 시스템 관리자를 대상으로 합니다. 또한 리소스 풀, 클러스터, DRS, 데이터스토어 클러스터, Storage DRS, Storage I/O Control 또는 vSphere DPM을 이해하고 사용하려는 모든 사용자가 참고해야 할 필수 자료입니다.

이 설명서에서는 사용자가 VMware ESXi 및 vCenter Server에 대한 실무 지식을 갖추고 있다고 가정합니다.

참고 이 문서에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

vSphere 리소스 관리 시작

1

vSphere 리소스 관리를 이해하려면 리소스의 구성 요소, 목표 및 클러스터 설정에서 리소스를 최적으로 구현하는 방법을 알아야 합니다.

가상 시스템에 대한 리소스 할당 설정을 지정하고 보는 방법을 포함하여 이러한 설정(공유, 예약 및 제한)에 대해 설명합니다. 또한 기존 리소스에 대해 리소스 할당 설정의 유효성을 검사하는 프로세스인 승인 제어에 대해서도 설명합니다.

리소스 관리는 리소스 공급자가 리소스 소비자에게 리소스를 할당하는 것입니다.

리소스의 오버 커밋(용량보다 수요가 더 많음) 그리고 수요와 용량은 시간에 따라 변한다는 사실 때문에 리소스 관리가 필요합니다. 리소스 관리를 통해 리소스를 동적으로 재할당함으로써 가용 용량을 더 효율적으로 이용할 수 있습니다.

참고 이 장에서 "메모리"는 물리적 RAM을 나타냅니다.

다음으로 아래 항목을 읽으십시오.

- [ESXi 리소스 유형](#)
- [물리적 리소스 사용](#)
- [리소스 활용](#)
- [vSphere를 통한 리소스 관리의 목표](#)

ESXi 리소스 유형

리소스에는 CPU, 메모리, 전원, 스토리지 및 네트워크 리소스가 포함됩니다.

참고 ESXi는 네트워크 트래픽 조절 및 비례 공유 메커니즘을 각각 사용하여 호스트별로 네트워크 대역폭 및 디스크 리소스를 관리합니다.

물리적 리소스 사용

vSphere를 사용하면 호스트와 클러스터(데이터스토어 클러스터 포함)를 물리적 리소스로 사용할 수 있습니다.

호스트의 경우 사용 가능한 리소스는 호스트의 하드웨어 사양에서 가상화 소프트웨어에 사용되는 리소스를 뺀 리소스입니다.

클러스터는 호스트 그룹입니다. vSphere Client를 사용하여 클러스터를 생성하고 여러 호스트를 클러스터에 추가할 수 있습니다. vCenter Server에서는 이러한 호스트의 리소스를 함께 관리합니다. 클러스터가 모든 호스트의 모든 CPU 및 메모리를 소유하고 있습니다. 결합 로드 밸런싱 또는 페일오버에 클러스터를 사용하도록 설정할 수 있습니다. 자세한 내용은 [장 16 vSphere DRS 클러스터 생성](#)의 내용을 참조하십시오.

데이터스토어 클러스터는 데이터스토어의 그룹입니다. DRS 클러스터와 마찬가지로 vSphere Client를 사용하여 데이터스토어 클러스터를 생성하고 클러스터에 여러 데이터스토어를 추가할 수 있습니다. vCenter Server는 데이터스토어 리소스를 함께 관리합니다. Storage DRS를 사용하도록 설정하여 I/O 로드 및 공간 사용률의 균형을 조정할 수 있습니다. [장 19 vSphere 데이터스토어 클러스터 생성](#)을 참조하십시오.

리소스 활용

가상 시스템은 리소스 소비자입니다.

생성 중에 할당된 기본 리소스 설정은 대부분의 시스템에서 잘 작동합니다. 나중에 가상 시스템 설정을 편집하여 리소스 공급자의 전체 CPU, 메모리 및 스토리지 I/O 또는 CPU 및 메모리의 보장된 예약에 대한 공유 기반 백분율을 할당할 수 있습니다. 해당 가상 시스템의 전원을 켜면 서버는 사용할 수 있는 예약되지 않은 리소스가 충분한지 여부를 확인하고 리소스가 충분한 경우에만 전원을 켤 수 있도록 허용합니다. 이 프로세스를 승인 제어라고 합니다.

리소스 풀은 리소스를 유연성 있게 관리하기 위한 논리적 추상화입니다. 리소스 풀을 계층 구조로 그룹화하여 사용할 가능한 CPU 및 메모리 리소스를 계층적으로 분할하는 데 사용할 수 있습니다. 따라서 리소스 풀은 리소스 제공자와 소비자 모두로 간주될 수 있습니다. 리소스 풀은 하위 리소스 풀 및 가상 시스템에 리소스를 제공하지만 또한 자신의 상위 리소스를 소비하므로 리소스 소비자이기도 합니다. [장 14 vSphere를 사용하여 리소스 풀 관리](#)의 내용을 참조하십시오.

ESXi 호스트는 다음과 같은 요인을 기준으로 각 가상 시스템에 기본 하드웨어 리소스의 일부를 할당합니다.

- 사용자가 정의하는 리소스 제한
- ESXi 호스트 또는 클러스터에 사용할 수 있는 총 리소스
- 전원이 켜진 가상 시스템 수 및 해당 가상 시스템에 의한 리소스 사용량
- 가상화를 관리하는 데 필요한 오버헤드

vSphere를 통한 리소스 관리의 목표

vSphere를 통해 리소스를 관리할 때는 목표가 무엇인지 알고 있어야 합니다.

vSphere를 사용한 리소스 관리는 리소스 오버 커밋 문제를 해결하는 것 외에도 다음을 달성하는 데 도움이 될 수 있습니다.

- 성능 분리: 가상 시스템이 리소스를 독점하는 것을 방지하고 예측 가능한 서비스율을 보장합니다.
- 효율적 사용: 언더 커밋된 리소스를 활용하고 적절한 성능 저하로 오버 커밋합니다.
- 쉬운 관리: 가상 시스템의 상대적 중요성을 제어하고, 유연한 동적 파티셔닝을 제공하고, 절대적 서비스 수준 계약을 충족합니다.

vSphere 리소스 할당 설정 구성

2

사용 가능한 리소스 용량이 리소스 소비자(및 가상화 오버헤드)가 필요로 하는 양보다 부족할 경우에는 관리자가 vSphere의 가상 시스템 또는 리소스 풀에 할당된 리소스 양을 사용자 지정해야 할 수 있습니다.

리소스 할당 설정(공유, 예약 및 제한)을 사용하여 가상 시스템에 제공된 CPU, 메모리 및 스토리지 리소스의 양을 확인할 수 있습니다. 특히 관리자는 몇 가지 리소스 할당 옵션을 사용할 수 있습니다.

- 호스트 또는 클러스터의 물리적 리소스를 예약합니다.
- 가상 시스템에 할당할 수 있는 리소스의 상한을 설정합니다.
- 특정 가상 시스템에 다른 가상 시스템보다 항상 더 높은 비율의 물리적 리소스를 할당합니다.

참고 이 장에서 "메모리"는 물리적 RAM을 나타냅니다.

다음으로 아래 항목을 읽으십시오.

- 리소스 할당 공유
- 리소스 할당 예약
- 리소스 할당 제한
- 리소스 할당 설정 제안
- vSphere 설정 편집
- 리소스 할당 설정 변경 - 예
- vSphere 승인 제어

리소스 할당 공유

공유는 가상 시스템이나 리소스 풀의 상대적 중요도를 지정합니다. 가상 시스템이 다른 가상 시스템 리소스 공유의 두 배를 가지고 있는 경우 두 개의 가상 시스템이 리소스 확보를 위해 경쟁한다면 이 리소스의 두 배를 소비할 수 있는 자격이 주어집니다.

일반적으로 공유는 **높음**, **보통** 또는 **낮음**으로 지정되며 이 값은 각각 4:2:1 비율로 공유 값을 지정합니다. **사용자 지정**을 선택하여 각 가상 시스템에 특정 공유 개수(비례 가중치를 나타냄)를 할당할 수도 있습니다.

공유를 지정하는 것은 형제 가상 시스템 또는 리소스 풀, 즉 리소스 풀 계층 구조에서 상위 개체가 동일한 가상 시스템 또는 리소스 풀과 관련해서만 의미가 있습니다. 형제는 예약 및 제한에 의해 정해지는 상대적 공유 값에 따라 리소스를 공유합니다. 가상 시스템에 공유를 할당할 때 항상 해당 가상 시스템의 우선 순위를 전원이 켜진 다른 가상 시스템에 상대적으로 지정하십시오.

다음 표에서는 하나의 가상 시스템에 대한 기본 CPU 및 메모리 공유 값을 보여 줍니다. 리소스 풀의 경우 기본 CPU 및 메모리 공유 값은 같지만 리소스 풀을 네 개의 가상 CPU 및 16GB 메모리가 포함된 가상 시스템인 것처럼 보아 공유 값에 곱해야 합니다.

표 2-1. 공유 값

설정	CPU 공유 값	메모리 공유 값
높음	가상 CPU당 2000 공유	구성된 가상 시스템 메모리에 대해 메가바이트당 20 공유
일반	가상 CPU당 1000 공유	구성된 가상 시스템 메모리에 대해 메가바이트당 10 공유
낮음	가상 CPU당 500 공유	구성된 가상 시스템 메모리에 대해 메가바이트당 5 공유

예를 들어 CPU 및 메모리 공유가 **보통**으로 설정된 두 개의 가상 CPU 및 1GB RAM이 포함된 SMP 가상 시스템에는 $2 \times 1000 = 2000$ CPU 공유 및 $10 \times 1024 = 10240$ 메모리 공유가 있습니다.

두 개 이상의 가상 CPU가 포함된 가상 시스템을 SMP(대칭적 다중 처리) 가상 시스템이라고 합니다.

새 가상 시스템의 전원을 켜면 각 공유로 표시되는 상대적 우선 순위가 변경됩니다. 이로 인해 동일한 리소스 풀에 있는 모든 가상 시스템이 영향을 받습니다. 모든 가상 시스템에는 동일한 수의 가상 CPU가 있습니다. 다음과 같은 예를 고려해 보십시오.

- 두 개의 CPU 바인딩된 가상 시스템이 8GHz의 총 CPU 용량을 가진 호스트에서 실행되고 있습니다. 해당 CPU 공유는 **보통**으로 설정되어 있고 각각 4GHz를 가져옵니다.
- 세 번째 CPU 바인딩된 가상 시스템의 전원이 켜집니다. 해당 CPU 공유 값은 **높음**으로 설정되어 있으며 이는 **보통**으로 설정된 시스템보다 두 배의 공유를 가져야 한다는 의미입니다. 새 가상 시스템은 4GHz를 받으며 다른 두 개의 시스템은 각각 2GHz만 가져옵니다. 사용자가 세 번째 가상 시스템에 대해 사용자 지정 공유 값으로 2000을 지정하는 경우에도 동일한 결과가 발생합니다.

리소스 할당 예약

예약은 가상 시스템에 보장된 최소 할당을 지정합니다.

vCenter Server 또는 ESXi를 사용하면 예약되지 않은 리소스가 가상 시스템의 예약을 처리할 수 있을 만큼 충분히 있는 경우에만 가상 시스템의 전원을 켤 수 있습니다. 서버는 물리적 서버의 로드가 심한 경우에도 이 양을 보장합니다. 예약은 구체적인 단위(메가헤르츠 또는 메가바이트)로 표현됩니다.

예를 들어 2GHz를 사용할 수 있고 VM1에 1GHz, VM2에 1GHz 예약을 지정한다고 가정합니다. 이제 각 가상 시스템은 필요한 경우 1GHz를 보장받을 수 있습니다. 그러나 VM1이 500MHz만 사용하는 경우 VM2는 1.5GHz를 사용할 수 있습니다.

예약은 기본적으로 0으로 설정됩니다. 가상 시스템에 필요한 최소 CPU 또는 메모리 양을 항상 사용할 수 있도록 보장해야 하는 경우 예약을 지정할 수 있습니다.

리소스 할당 제한

제한은 가상 시스템에 할당할 수 있는 CPU, 메모리 또는 스토리지 I/O 리소스의 상한을 지정합니다.

서버는 가상 시스템에 예약된 것보다 더 많은 리소스를 할당할 수 있지만 시스템에 사용하지 않는 리소스가 있는 경우에도 제한 수보다 많이 할당하지는 않습니다. 제한은 구체적인 단위(메가헤르쯔, 메가바이트 또는 초당 I/O 작업 수)로 표시됩니다.

CPU, 메모리 및 스토리지 I/O 리소스 제한의 기본값은 무제한입니다. 메모리 한계가 무제한이면, 메모리가 생성되었을 때 가상 시스템을 위해 구성된 메모리 양은 이것의 유효 한계가 됩니다.

대부분의 경우 제한을 지정할 필요가 없습니다. 제한을 지정할 경우 장단점이 있습니다.

- **장점** — 적은 수의 가상 시스템으로 시작하고 사용자 기대치를 관리하려는 경우 제한을 할당하면 유용합니다. 가상 시스템을 더 추가할수록 성능이 저하됩니다. 제한을 지정하여 사용 가능한 리소스가 더 적은 상태를 시뮬레이션할 수 있습니다.
- **단점** — 제한을 지정하는 경우 유휴 리소스를 낭비할 수 있습니다. 시스템이 충분히 사용되지 않는 상태여서 유휴 리소스를 사용할 수 있는 경우에도 시스템은 가상 시스템에서 제한보다 많은 리소스를 사용하도록 허용하지 않습니다. 제한을 지정할 만한 이유가 충분한 경우에만 제한을 지정하십시오.

리소스 할당 설정 제안

ESXi 환경에 적합한 리소스 할당 설정(예약, 제한 및 공유)을 선택할 수 있습니다.

다음 지침은 가상 시스템의 성능을 높이는 데 도움이 됩니다.

- **예약**을 사용하여 CPU 또는 메모리의 사용 가능한 크기가 아닌 허용되는 최소 크기를 지정합니다. 예약으로 표시되는 구체적인 리소스 크기는 가상 시스템을 추가하거나 제거하는 등 환경을 변경하는 경우에 변경되지 않습니다. 호스트는 공유 수, 예측된 요구량 및 가상 시스템의 제한을 기준으로 추가 리소스를 사용 가능한 리소스로 할당합니다.
- 가상 시스템의 예약을 지정할 때 모든 리소스를 커밋하지 말고 최소 10%는 예약되지 않은 상태로 두도록 계획하십시오. 예약한 양이 시스템의 전체 용량에 가까워질수록 승인 제어를 위반하지 않고 예약 및 리소스 풀 계층 구조를 변경하기가 점점 어려워집니다. DRS를 사용하도록 설정된 클러스터에서 클러스터 또는 클러스터의 개별 호스트 용량 전체를 커밋하도록 예약하면 DRS가 호스트 간에 가상 시스템을 마이그레이션하지 못할 수 있습니다.
- 사용 가능한 전체 리소스를 자주 변경해야 하는 경우 **공유**를 사용하여 가상 시스템 간에 공평하게 리소스를 할당하십시오. 예를 들어 **공유**를 사용하고 호스트를 업그레이드하는 경우, 각 공유가 대량의 메모리, CPU 또는 스토리지 I/O 리소스를 나타내는 경우에도 각 가상 시스템은 동일한 우선 순위에 있으며 같은 수의 공유를 유지합니다.

vSphere 설정 편집

메모리와 CPU 리소스 할당을 변경하려면 설정 편집 대화상자를 이용합니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
- 2 마우스 오른쪽 버튼을 클릭하고 **설정 편집**을 선택합니다.
- 3 CPU 리소스를 편집합니다.

옵션	설명
공유	CPU는 상위의 전체와 관련된 리소스 풀을 공유합니다. 형제 리소스 풀은 예약이나 한도로 바인딩된 상대적 공유 값에 따라서 리소스를 공유합니다. 낮음 , 보통 또는 높음 을 선택합니다. 이 설정은 각각 1:2:4 비율의 공유 값을 지정합니다. 각 가상 시스템에 특정 공유 개수(비례 가중치)를 지정하려면 사용자 지정 을 선택합니다.
예약	이 리소스 풀에 보장된 CPU 할당.
제한	리소스 풀의 CPU 할당 상한 값. 상한 값을 지정하지 않으려면 무제한 을 선택합니다.

- 4 메모리 리소스를 편집합니다.

옵션	설명
공유	메모리는 상위의 전체와 관련된 리소스 풀을 공유합니다. 형제 리소스 풀은 예약이나 한도로 바인딩된 상대적 공유 값에 따라서 리소스를 공유합니다. 낮음 , 보통 또는 높음 을 선택합니다. 이 설정은 각각 1:2:4 비율의 공유 값을 지정합니다. 각 가상 시스템에 특정 공유 개수(비례 가중치)를 지정하려면 사용자 지정 을 선택합니다.
예약	이 리소스 풀에 보장된 메모리 할당.
제한	리소스 풀의 메모리 할당 상한 값. 상한 값을 지정하지 않으려면 무제한 을 선택합니다.

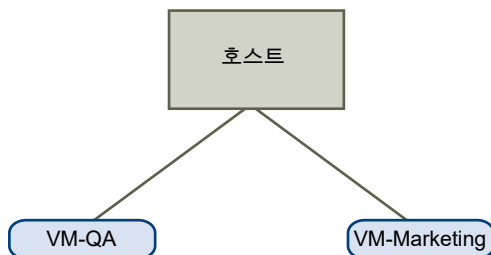
- 5 **확인**을 클릭합니다.

리소스 할당 설정 변경 - 예

다음 예에서는 리소스 할당 설정을 변경하여 가상 시스템 성능을 높이는 방법을 보여 줍니다.

ESXi 호스트에서 QA(VM-QA) 부서와 마케팅(VM-Marketing) 부서에 대해 하나씩, 새 가상 시스템을 두 개 생성한 것으로 가정합니다.

그림 2-1. 두 개의 가상 시스템이 있는 단일 호스트



다음 예에서는 VM-QA에서 메모리를 많이 사용하여 두 가상 시스템의 리소스 할당 설정을 필요에 맞게 변경해야 한다고 가정합니다.

- 시스템 메모리가 오버 커밋되면 VM-QA가 마케팅 가상 시스템보다 CPU와 메모리 리소스를 두 배 더 많이 사용할 수 있도록 지정합니다. CPU 공유 및 메모리 공유를 VM-QA에 대해서는 **높음**으로 설정하고 VM-Marketing에 대해서는 **보통**으로 설정합니다.
- 마케팅 가상 시스템에 어느 정도의 CPU 리소스가 보장되어야 합니다. 이를 위해서는 예약 설정을 사용할 수 있습니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
- 2 공유를 변경할 대상 가상 시스템인 **VM-QA**를 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
- 3 **가상 하드웨어**에서 CPU를 확장하고 **공유** 드롭다운 메뉴에서 **높음**을 선택합니다.
- 4 **가상 하드웨어**에서 메모리를 확장하고 **공유** 드롭다운 메뉴에서 **높음**을 선택합니다.
- 5 **확인**을 클릭합니다.
- 6 마케팅 가상 시스템(**VM-Marketing**)을 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
- 7 **가상 하드웨어**에서 CPU를 확장하고 **예약** 값을 원하는 값으로 변경합니다.
- 8 **확인**을 클릭합니다.

vSphere 승인 제어

가상 시스템의 전원을 켜면 아직 예약되지 않은 CPU 및 메모리 리소스의 양이 확인됩니다. 예약되지 않은 사용 가능한 리소스를 기반으로 시스템은 가상 시스템이 구성되어 있는 예약(있는 경우)을 보장할 수 있는지 여부를 판단합니다. 이 프로세스를 승인 제어라고 합니다.

예약되지 않은 CPU와 메모리가 충분하거나 예약이 없으면 가상 시스템의 전원이 켜집니다. 그렇지 않으면 리소스 부족 주의가 나타납니다.

참고 사용자가 지정한 메모리 예약 외에도 각 가상 시스템에는 오버헤드 메모리 양도 존재합니다. 이러한 추가 메모리 커밋은 승인 제어 계산에 포함됩니다.

vSphere DPM 기능을 사용하도록 설정하면 전원 소비를 줄이기 위해 호스트가 대기 모드(즉, 전원이 꺼짐)로 전환될 수 있습니다. 이러한 호스트에서 제공하는 예약되지 않은 리소스는 승인 제어에 사용할 수 있는 것으로 간주됩니다. 이러한 리소스가 없어서 가상 시스템의 전원을 켤 수 없으면 충분한 대기 호스트의 전원을 켜라는 권장 사항이 제공됩니다. 자세한 내용은 **전원 리소스 관리**의 내용을 참조하십시오.

vSphere의 CPU 가상화

3

CPU 가상화는 성능에 중점을 두며 가능할 때마다 프로세서에서 바로 실행됩니다. 기본적인 물리적 리소스는 가능한 모든 경우에 사용되며 가상화 계층은 가상 시스템이 물리적 시스템에서 직접 실행되는 것처럼 작동하도록 하는데 필요한 명령만 실행합니다.

CPU 가상화는 에뮬레이션과 동일하지 않습니다. ESXi는 에뮬레이션을 사용하여 가상 CPU를 실행하지 않습니다. 에뮬레이션을 사용하면 모든 작업이 에뮬레이터에 의해 소프트웨어로 실행됩니다. 소프트웨어 에뮬레이터를 사용하면 프로그램이 원래 작성된 컴퓨터 시스템이 아닌 다른 컴퓨터 시스템에서 실행될 수 있습니다. 에뮬레이터는 동일한 데이터 또는 입력을 수락하고 동일한 결과를 달성함으로써 원래 컴퓨터의 동작을 에뮬레이션하거나 재현하여 이 작업을 수행합니다. 에뮬레이션은 이식성을 제공하며 여러 플랫폼에서 하나의 플랫폼용으로 설계된 소프트웨어를 실행합니다.

CPU 리소스가 오버 커밋되면 ESXi 호스트는 모든 가상 시스템에서 물리적 프로세서를 시간 분할하며 따라서 각 가상 시스템은 지정된 수의 가상 프로세서가 있는 것처럼 실행됩니다. ESXi 호스트는 여러 가상 시스템을 실행할 때 각 가상 시스템에 물리적 리소스의 공유를 할당합니다. 기본 리소스 할당 설정을 사용하면 동일한 호스트와 연결된 모든 가상 시스템에 가상 CPU당 동일한 CPU 공유가 할당됩니다. 따라서 단일 프로세서 가상 시스템에는 이중 프로세서 가상 시스템 리소스의 절반만 할당됩니다.

다음으로 아래 항목을 읽으십시오.

- 소프트웨어 기반 CPU 가상화
- 하드웨어 지원 CPU 가상화
- 가상화 및 프로세서별 동작
- CPU 가상화가 성능에 미치는 영향

소프트웨어 기반 CPU 가상화

소프트웨어 기반 CPU 가상화를 사용하면 게스트 애플리케이션 코드는 프로세서에서 직접 실행되는 반면, 게스트 특권 코드는 변환되며 변환된 코드가 프로세서에서 실행됩니다.

변환된 코드는 네이티브 버전에 비해 다소 크며 대개 더 느리게 실행됩니다. 따라서 특권 코드 구성 요소가 적은 게스트 애플리케이션은 네이티브 버전과 유사한 속도로 실행됩니다. 하지만 시스템 호출, 트랩 또는 페이지 테이블 업데이트 같이 특권 코드 구성 요소가 많은 애플리케이션은 가상화된 환경에서 매우 느리게 실행될 수 있습니다.

하드웨어 지원 CPU 가상화

특정 프로세서는 CPU 가상화에 대한 하드웨어 지원을 제공합니다.

이 지원을 사용하면 게스트는 게스트 모드라고 하는 별도의 실행 모드를 사용할 수 있습니다. 애플리케이션 코드 또는 특권 코드인 게스트 코드는 게스트 모드에서 실행됩니다. 특정 이벤트가 발생하면 프로세서가 게스트 모드를 종료하고 루트 모드로 들어갑니다. 그러면 하이퍼바이저가 루트 모드에서 실행되어 종료 이유를 확인하고, 필요한 조치를 수행한 다음 게스트를 게스트 모드에서 재시작합니다.

가상화에 하드웨어 지원을 사용하면 코드를 변환할 필요가 없습니다. 그 결과 시스템 호출 또는 트랩이 많은 워크로드가 기본 속도에 매우 근접한 속도로 실행됩니다. 페이지 테이블 업데이트 관련 워크로드를 포함한 일부 워크로드의 경우 게스트 모드를 종료하고 루트 모드로 전환되는 경우가 많이 발생합니다. 이러한 종로의 수와 종료에 소요되는 전체 시간에 따라 하드웨어 지원 CPU 가상화로 실행 속도를 크게 높일 수 있습니다.

가상화 및 프로세서별 동작

VMware 소프트웨어는 CPU를 가상화하지만 가상 시스템은 해당 시스템이 실행 중인 프로세서의 특정 모델을 감지합니다.

프로세서 모델마다 제공하는 CPU 기능이 다를 수 있으며 가상 시스템에서 실행 중인 애플리케이션은 이러한 기능을 이용할 수 있습니다. 따라서 vMotion[®]을 사용하여 기능이 서로 다른 프로세서에서 실행 중인 시스템 간에 가상 시스템을 마이그레이션하는 것은 불가능합니다. 경우에 따라서는 EVC(Enhanced vMotion Compatibility) 기능을 지원하는 프로세서에서 이 기능을 사용하여 이 제한을 피할 수 있습니다. 자세한 내용은 "vCenter Server 및 호스트 관리" 설명서를 참조하십시오.

CPU 가상화가 성능에 미치는 영향

CPU 가상화는 사용되는 가상화 유형과 워크로드에 따라 오버헤드 양을 다르게 추가합니다.

애플리케이션이 사용자 상호 작용, 디바이스 입력 또는 데이터 검색 등의 외부 이벤트를 기다리는 대신 명령을 실행하는 데 대부분의 시간을 보낼 경우 CPU 바인딩되었다고 합니다. 이러한 애플리케이션의 경우 CPU 가상화 오버헤드에 실행되어야 할 추가 명령이 포함됩니다. 이 오버헤드는 애플리케이션 자체에서 사용할 수 있는 CPU 처리 시간에 영향을 미칩니다. CPU 가상화 오버헤드는 일반적으로 전체적인 성능 저하로 해석됩니다.

CPU 바인딩되지 않은 애플리케이션의 경우 CPU 가상화는 CPU 사용 증가로 해석됩니다. 여분의 CPU 용량으로 오버헤드를 흡수할 수 있는 경우 전체적인 처리량 차원에서 여전히 우수한 성능을 제공합니다.

ESXi는 각 가상 시스템에 최대 128개의 가상 프로세서(CPU)를 지원합니다.

참고 최상의 성능 및 리소스 사용을 위해서는 단일 스레드 애플리케이션을 여러 CPU가 있는 SMP 가상 시스템 대신 단일 프로세서 가상 시스템에 배포하십시오.

단일 스레드 애플리케이션에서는 단일 CPU만 사용할 수 있습니다. 듀얼 프로세서 가상 시스템에 이러한 애플리케이션을 배포해도 애플리케이션의 속도가 향상되지는 않습니다. 대신, 두 번째 가상 CPU가 다른 가상 시스템이 사용할 가능성이 있는 물리적 리소스를 사용하게 됩니다.

vSphere CPU 리소스 관리

4

각각 고유한 등록 및 제어 구조를 가진 가상 프로세서를 하나 이상 사용하는 가상 시스템을 구성할 수 있습니다.

가상 시스템을 스케줄링하면 해당 가상 프로세서가 물리적 프로세서에서 실행되도록 스케줄링됩니다. VMkernel 리소스 관리자는 물리적 CPU의 가상 CPU를 스케줄링하여 물리적 CPU 리소스에 대한 가상 시스템의 액세스를 관리합니다.

참고 이 장에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- 프로세서 정보 보기
- CPU 구성 지정
- 다중 코어 프로세서

프로세서 정보 보기

vSphere Client에서 현재 CPU 구성에 대한 정보에 액세스할 수 있습니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 **하드웨어**에서 **CPU**를 확장하여 물리적 프로세서의 개수 및 유형과 논리적 프로세서의 개수에 대한 정보를 봅니다.

참고 하이퍼스레드 시스템에서는 각 하드웨어 스레드가 논리 프로세서입니다. 예를 들어, 하이퍼스레딩이 활성화된 듀얼코어 프로세서에는 2개의 코어와 4개의 논리 프로세서가 있습니다.

CPU 구성 지정

CPU 구성을 지정하여 리소스 관리를 향상시킬 수 있습니다. 하지만 CPU 구성을 사용자 지정하지 않으면 ESXi 호스트는 대부분의 상황에서 잘 작동하는 기본값을 사용합니다.

다음과 같은 방법으로 CPU 구성을 지정할 수 있습니다.

- vSphere Client를 통해 사용할 수 있는 특성과 특수 기능을 사용합니다. vSphere Client를 사용하여 ESXi 호스트나 vCenter Server 시스템에 연결할 수 있습니다.
- 특정 상황에서 고급 설정을 사용합니다.
- 스크립팅된 CPU 할당에 대해 vSphere SDK를 사용합니다.
- 하이퍼스레딩을 사용합니다.

다중 코어 프로세서

다중 코어 프로세서는 가상 시스템의 멀티태스킹을 수행하는 호스트에 많은 장점을 제공합니다.

참고 이 항목에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

Intel과 AMD에서는 둘 이상의 프로세서 코어를 단일 집적 회로(일반적으로 패키지 또는 소켓이라고 함)로 결합하는 프로세서를 개발했습니다. VMware에서는 논리 프로세서가 각각 하나 이상씩 있는 하나 이상의 프로세서 코어를 포함할 수 있는 단일 패키지를 설명할 때 소켓이라는 용어를 사용합니다.

예를 들어 듀얼 코어 프로세서는 두 개의 가상 CPU를 동시에 실행할 수 있으므로 단일 코어 프로세서보다 거의 두 배의 성능을 제공합니다. 동일한 프로세서 내의 코어는 일반적으로 모든 코어에 사용되는 공유된 마지막 수준 캐시를 사용하여 구성되므로 느린 메인 메모리에 액세스할 필요성이 잠재적으로 줄어듭니다. 물리적 프로세서를 메인 메모리에 연결해 주는 공유 메모리 버스는 논리 프로세서에서 실행 중인 가상 시스템이 동일한 메모리 버스 리소스를 얻기 위해 경쟁하는 메모리 소모가 많은 워크로드를 실행할 경우 논리 프로세서의 성능을 제한할 수 있습니다.

각 프로세서 코어의 각 논리 프로세서는 ESXi CPU 스케줄러에서 독립적으로 사용되어 가상 시스템을 실행함으로써 SMP 시스템과 유사한 기능을 제공합니다. 예를 들어 양방향 가상 시스템의 가상 프로세서는 동일한 코어에 속한 논리 프로세서 또는 다른 물리적 코어에 있는 논리 프로세서에서 실행될 수 있습니다.

ESXi CPU 스케줄러는 프로세서 토폴로지 및 프로세서 코어와 해당 코어의 논리 프로세서 간 관계를 감지할 수 있습니다. 스케줄러는 이 정보를 사용하여 가상 시스템을 스케줄링하고 성능을 최적화합니다.

ESXi CPU 스케줄러는 소켓, 코어 및 논리 프로세서 간의 관계를 비롯한 프로세서 토폴로지를 해석할 수 있습니다. 스케줄러는 토폴로지 정보를 사용하여 서로 다른 소켓에서 가상 CPU의 배치를 최적화합니다. 이러한 최적화를 통해 전체적인 캐시 사용량을 최대화하고 가상 CPU 마이그레이션을 최소화하여 캐시 선호도를 개선할 수 있습니다.

하이퍼스레딩 기술을 사용하면 단일 물리적 프로세서 코어가 두 개의 논리적 프로세서처럼 동작할 수 있습니다. 해당 프로세서는 동시에 두 개의 독립 애플리케이션을 실행할 수 있습니다. 논리적 프로세서와 물리적 프로세서 간의 혼동을 피하기 위해 Intel에서는 물리적 프로세서를 소켓이라고 부르며 이 장의 설명에서도 이 용어를 사용합니다.

Intel Corporation은 펜티엄 IV 및 Xeon 프로세서 라인의 성능을 향상시키기 위해 하이퍼스레딩 기술을 개발했습니다. 하이퍼스레딩 기술을 사용하면 단일 프로세서 코어가 두 개의 독립 스레드를 동시에 실행할 수 있습니다.

하이퍼스레딩은 시스템 성능을 두 배로 늘리지는 않지만 유휴 리소스를 더 잘 활용하여 특정한 중요 워크로드 유형에 대한 처리량을 높여 성능을 향상시킬 수 있습니다. 사용량이 많은 코어의 논리적 프로세서 하나에서 실행되는 애플리케이션은 하이퍼스레딩되지 않은 프로세서에서 단독으로 실행되는 동안 얻는 처리량의 절반보다 약간 더 많은 처리량을 기대할 수 있습니다. 하이퍼스레딩 성능 향상은 애플리케이션에 따라 크게 좌우되며, 많은 프로세서 리소스(예: 캐시)가 논리적 프로세서 간에 공유되기 때문에 하이퍼스레딩을 사용할 경우 일부 애플리케이션의 성능이 저하될 수 있습니다.

참고 Intel 하이퍼스레딩 기술이 탑재된 프로세서에서 각 코어에는 대부분의 코어 리소스(예: 메모리 캐시 및 기능 단위)를 공유하는 두 개의 논리적 프로세서가 있을 수 있습니다. 이러한 논리 프로세서는 일반적으로 스레드라고 합니다.

일부 프로세서는 하이퍼스레딩을 지원하지 않으므로 코어당 스레드가 하나만 있습니다. 이러한 프로세서의 경우 코어 수가 논리 프로세서 수와도 일치합니다.

다음으로 아래 항목을 읽으십시오.

- [하이퍼스레딩 및 ESXi 호스트](#)
- [하이퍼스레딩 사용](#)

하이퍼스레딩 및 ESXi 호스트

하이퍼스레딩을 사용하도록 설정된 호스트는 하이퍼스레딩을 사용하지 않는 호스트와 비슷하게 동작합니다. 하지만 하이퍼스레딩을 사용하도록 설정하는 경우에는 특정 사항을 고려해야 할 수 있습니다.

ESXi 호스트는 로드가 시스템의 프로세서 코어 간에 매끄럽게 분산되도록 프로세서 시간을 지능적으로 관리합니다. 동일한 코어에 있는 논리 프로세서는 연속된 CPU 번호를 가지므로 CPU 0과 1은 첫 번째 코어에 함께 있고 CPU 2와 3은 두 번째 코어에 함께 있습니다. 가상 시스템은 동일한 코어에 있는 두 논리 프로세서가 아닌 서로 다른 코어에서 스케줄링됩니다.

작업이 없는 논리 프로세서는 중지된 상태가 되어 실행 리소스가 해제되고 동일한 코어의 다른 논리 프로세서에서 실행 중인 가상 시스템이 코어의 전체 실행 리소스를 사용할 수 있게 됩니다. VMware 스케줄러는 이 중지 시간을 적절하게 감안하며 코어 절반에서 실행되는 가상 시스템보다 코어의 전체 리소스로 실행 중인 가상 시스템에 높은 가중치를 부여합니다. 이러한 프로세서 관리 방식을 통해 서버가 표준 ESXi 리소스 클레임 규칙을 위반하지 않도록 합니다.

하이퍼스레딩을 사용하여 호스트에서 CPU 선호도를 사용하도록 설정하기 전에 리소스 관리 요구 사항을 고려하십시오. 예를 들어 우선 순위가 높은 가상 시스템을 CPU 0에 바인딩하고 우선 순위가 높은 다른 가상 시스템을 CPU 1에 바인딩하면 두 가상 시스템은 동일한 물리적 코어를 공유해야 합니다. 이 경우 이 두 가상 시스템의 리소스 수요를 충족하지 못할 수 있습니다. 모든 사용자 지정 선호도 설정이 하이퍼스레딩 시스템에 적합한지 확인하십시오.

하이퍼스레딩 사용

하이퍼스레딩을 사용하도록 설정하려면 먼저 시스템의 BIOS 설정에서 하이퍼스레딩을 사용하도록 설정한 다음 vSphere Client에서 하이퍼스레딩을 설정해야 합니다. 하이퍼스레딩은 기본적으로 사용하도록 설정되어 있습니다.

현재 CPU에서 하이퍼스레딩을 지원하는지 여부를 확인하려면 해당 시스템 설명서를 참조하십시오.

절차

- 1 현재 시스템에서 하이퍼스레딩 기술을 지원하는지 확인합니다.
- 2 시스템 BIOS에서 하이퍼스레딩을 설정합니다.

일부 제조업체에서는 이 옵션을 **논리 프로세서**라고 하는 반면, 다른 제조업체에서는 **하이퍼스레딩 사용**이라고 합니다.

- 3 ESXi 호스트에 하이퍼스레딩을 사용하도록 설정되어 있는지 확인합니다.

- a vSphere Client에서 호스트를 찾습니다.
- b **구성**을 클릭합니다.
- c **시스템**에서 **고급 시스템 설정**을 클릭하고 **VMkernel.Boot.hyperthreading**을 선택합니다.

설정을 적용하려면 호스트를 다시 시작해야 합니다. 이 값이 **true**이면 하이퍼스레딩을 사용하도록 설정되어 있는 것입니다.

- 4 **하드웨어**에서 **프로세서**를 클릭하여 논리 프로세서 수를 확인합니다.

결과

하이퍼스레딩이 사용되도록 설정되었습니다.

각 가상 시스템에 대해 CPU 선호도 설정을 지정하여 다중 프로세서 시스템에서 사용 가능한 프로세서의 하위 집합으로 가상 시스템 할당을 제한할 수 있습니다. 이 기능을 사용하면 지정된 선호도 집합의 프로세서에 각 가상 시스템을 할당할 수 있습니다.

CPU 선호도는 가상 시스템과 프로세서 간 배치 제약 조건을 지정하며 가상 시스템과 가상 시스템 간 호스트 배치 제약 조건을 지정하는 VM-VM 또는 VM-호스트 선호도 규칙에 의해 생성된 관계와는 다릅니다.

이 컨텍스트에서 CPU라는 용어는 하이퍼스레딩된 시스템의 논리 프로세서를 의미하며 하이퍼스레딩이 아닌 시스템의 코어를 나타냅니다.

가상 시스템에 대한 CPU 선호도 설정은 가상 시스템과 연결된 모든 가상 CPU 및 가상 시스템과 연결된 다른 모든 스레드(월드라고도 함)에 적용됩니다. 이러한 가상 시스템 스레드는 마우스, 키보드, 화면, CD-ROM 및 기타 레거시 디바이스를 에뮬레이션하는 데 필요한 처리를 수행합니다.

디스플레이 사용이 많은 워크로드와 같은 경우에 가상 CPU와 다른 가상 시스템 스레드 간에 상당한 통신이 발생할 수 있습니다. 가상 시스템의 선호도 설정으로 인해 이러한 추가 스레드가 가상 시스템의 가상 CPU와 동시에 스케줄링되지 않으면 성능이 저하될 수 있습니다. 예를 들어 단일 CPU에 대한 선호도가 있는 단일 프로세서 가상 시스템 또는 CPU 2개에만 선호도가 있는 양방향 SMP 가상 시스템이 있습니다.

최상의 성능을 위해 수동 선호도 설정을 사용하는 경우에는 선호도 설정에 물리적 CPU를 추가로 하나 이상 포함하여 가상 시스템의 스레드 중 하나 이상이 가상 CPU와 동시에 스케줄링되도록 하는 것이 좋습니다. CPU 2개 이상에 대한 선호도가 있는 단일 프로세서 가상 시스템 또는 CPU 3개 이상에 대한 선호도가 있는 양방향 SMP 가상 시스템이 이러한 예에 포함됩니다.

다음으로 아래 항목을 읽으십시오.

- [특정 프로세서에 가상 시스템 할당](#)
- [CPU 선호도의 잠재적인 문제](#)

특정 프로세서에 가상 시스템 할당

CPU 선호도를 이용하여 가상 시스템을 특정 프로세서에 할당할 수 있습니다. 이는 다중 프로세서 시스템에서 사용할 수 있는 특정한 프로세서로 가상 시스템 할당을 제한할 수 있습니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
 - a 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - b **VM**을 선택합니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭한 후 **설정 편집**을 클릭합니다.
- 3 가상 하드웨어 아래에서 **CPU**를 확장합니다.
- 4 스케줄링 선호도 아래에서 가상 시스템의 물리적 프로세서 선호도를 선택합니다.
범위를 나타내려면 '-'를 사용하고 여러 값을 구분해 입력하려면 ','를 사용하십시오.
예를 들어 "0, 2, 4-7"은 프로세서 0, 2, 4, 5, 6 및 7을 나타냅니다.
- 5 가상 시스템을 실행할 프로세서를 선택하고 **확인**을 클릭합니다.

CPU 선호도의 잠재적인 문제

CPU 선호도를 사용하기 전에 특정 문제를 고려해야 할 수 있습니다.

CPU 선호도와 관련된 잠재적인 문제는 다음과 같습니다.

- 다중 프로세서 시스템의 경우 ESXi 시스템은 자동 로드 밸런싱을 수행합니다. 프로세서 간에 로드 밸런싱을 수행하는 스케줄러의 기능을 향상하려면 가상 시스템 선호도를 수동으로 지정하지 마십시오.
- 선호도는 ESXi 호스트가 가상 시스템에 대해 지정된 예약 및 공유를 충족하는 기능을 방해할 수 있습니다.
- CPU 승인 제어는 선호도를 고려하지 않기 때문에 수동 선호도 설정이 있는 가상 시스템은 전체 예약을 항상 수신하지 못할 수 있습니다.
수동 선호도 설정이 없는 가상 시스템은 수동 선호도 설정이 있는 가상 시스템으로부터 부정적인 영향을 받지 않습니다.
- 가상 시스템을 한 호스트에서 다른 호스트로 이동하는 경우 선호도가 더 이상 적용되지 않을 수 있습니다. 새 호스트에 있는 프로세서 수가 다를 수 있기 때문입니다.
- NUMA 스케줄러는 선호도를 사용하여 특정 프로세서에 이미 할당된 가상 시스템을 관리하지 못할 수 있습니다. 자세한 내용은 [장 21 ESXi와 함께 NUMA 시스템 사용](#)의 내용을 참조하십시오.
- 선호도는 다중 코어 또는 하이퍼스레딩 프로세서에서 가상 시스템을 스케줄링하여 해당 프로세서에서 공유되는 리소스를 최대한 활용하는 호스트의 기능에 영향을 줄 수 있습니다.

ESXi 호스트 전원 관리 정책

7

호스트 하드웨어에서 제공하는 ESXi의 여러 전원 관리 기능을 적용하여 성능과 전원 간의 균형을 조정할 수 있습니다. ESXi에서 이러한 기능을 사용하는 방법은 전원 관리 정책을 선택하여 제어할 수 있습니다.

고성능 정책을 선택하면 보다 강력한 성능을 얻을 수 있지만 효율성 및 와트당 성능은 낮아집니다. 저전력 정책을 선택하면 절대 성능은 줄어들이지만 효율성이 높아집니다.

VMware Host Client를 사용하면 관리하는 호스트에 대한 정책을 선택할 수 있습니다. 정책을 선택하지 않으면 ESXi에서는 기본적으로 균형 조정을 사용합니다.

표 7-1. CPU 전원 관리 정책

전원 관리 정책	설명
고성능	전원 관리 기능을 사용하지 않습니다.
균형 조정(기본값)	성능 손실을 최소화하면서 에너지 소비 줄임
저전력	성능 저하의 위험을 감수하고 에너지 소비 줄임
사용자 지정	사용자 정의 전원 관리 정책입니다. 고급 구성을 사용할 수 있게 됩니다.

CPU가 낮은 주파수로 실행되면 낮은 전압으로 실행되어 전력을 절감할 수 있습니다. 이러한 유형의 전원 관리를 일반적으로 DVFS(Dynamic Voltage and Frequency Scaling)라 합니다. ESXi에서는 가상 시스템 성능이 저하되지 않도록 CPU 주파수를 조정하려고 합니다.

CPU가 유휴 상태이면 ESXi는 깊은 중지 상태(C-상태라고도 함)를 적용할 수 있습니다. C-상태가 깊어질수록 CPU에서 더 적은 전력을 사용하지만 CPU가 실행을 재개하는 데 걸리는 시간도 늘어납니다. CPU가 유휴 상태가 되면 ESXi에서는 유휴 상태 기간을 예측하는 알고리즘을 적용해 전환할 적절한 C-상태를 선택합니다. 깊은 C-상태를 사용하지 않는 전원 관리 정책의 경우 ESXi는 유휴 상태의 CPU에 가장 얇은 중지 상태인 C1만 사용합니다.

다음으로 아래 항목을 읽으십시오.

- CPU 전원 관리 정책 선택
- 호스트 전원 관리를 위한 사용자 지정 정책 매개 변수 구성

CPU 전원 관리 정책 선택

vSphere Client를 사용하여 호스트에 대한 CPU 전원 관리 정책을 설정할 수 있습니다.

사전 요구 사항

호스트 시스템의 BIOS 설정에서 운영 체제의 전원 관리 제어를 허용하는지 확인합니다(예: **OS 제어됨**). 호스트 하드웨어에서 운영 체제가 전원을 관리하도록 허용하지 않으면 지원되지 않음 정책만 사용할 수 있습니다. 일부 시스템의 경우 고성능 정책만 사용할 수 있습니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 구성을 클릭합니다.
- 3 하드웨어에서 개요를 선택하고 **전원 정책 편집**을 클릭합니다.
- 4 호스트에 대한 전원 관리 정책을 선택하고 **확인**을 클릭합니다.

정책 선택은 호스트 구성에 저장되며 부팅 시 다시 사용될 수 있습니다. 이 정책은 언제든지 변경할 수 있으며 서버 재부팅이 필요 없습니다.

호스트 전원 관리를 위한 사용자 지정 정책 매개 변수 구성

호스트 전원 관리에 사용자 지정 정책을 사용할 경우 ESXi의 전원 관리 정책은 여러 고급 구성 매개 변수의 값을 기반으로 합니다.

사전 요구 사항

CPU 전원 관리 정책 선택에 설명된 대로 전원 관리 정책의 **사용자 지정**을 선택합니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 구성을 클릭합니다.
- 3 시스템에서 **고급 시스템 설정**을 선택합니다.
- 4 오른쪽 창에서 사용자 지정 정책에 영향을 주는 전원 관리 매개 변수를 편집할 수 있습니다.

사용자 지정 정책에 영향을 주는 전원 관리 매개 변수의 설명은 **사용자 지정 정책에서**로 시작합니다. 다른 모든 전원 매개 변수는 모든 전원 관리 정책에 영향을 줍니다.

- 5 매개 변수를 선택하고 **편집** 버튼을 클릭합니다.

참고 전원 관리 매개 변수의 기본값은 균형 조정 정책과 일치합니다.

매개 변수	설명
Power.UsePStates	프로세서가 사용 중일 때 전원을 절약하기 위해 ACPI P-상태를 사용합니다.
Power.MaxCpuLoad	CPU가 실제 시간의 지정된 백분율 이하로 사용 중일 때만 CPU의 전원을 절약하기 위해 P-상태를 사용합니다.
Power.MinFreqPct	최대 CPU 속도의 지정된 백분율보다 느린 P-상태는 사용하지 않습니다.

매개 변수	설명
Power.UseStallCtr	프로세서가 캐시 누락과 같은 이벤트를 기다리는 동안 빈번하게 지연되는 경우 보다 깊은 P-상태를 사용합니다.
Power.TimerHz	ESXi에서 각 CPU가 어떤 P-상태로 있어야 하는지를 재평가하는 초당 횟수를 제어합니다.
Power.UseCStates	프로세서가 유휴 상태인 경우 깊은 ACPI C-상태(C2 이하)를 사용합니다.
Power.CStateMaxLatency	지연 시간이 이 값보다 큰 C-상태는 사용하지 않습니다.
Power.CStateResidencyCoef	CPU가 유휴 상태가 될 때 해당 지연 시간에 이 값을 곱한 시간이 호스트에서 예측한 CPU의 유휴 시간보다 작은 가장 깊은 C-상태를 선택합니다. 값이 크면 ESXi에서는 보수적으로 깊은 C-상태를 사용하고 값이 작으면 적극적으로 깊은 C-상태를 사용합니다.
Power.CStatePredictionCoef	CPU의 유휴 상태가 유지되는 시간을 예측하기 위한 ESXi 알고리즘의 매개 변수입니다. 이 값은 변경하지 않는 것이 좋습니다.
Power.PerfBias	성능 에너지 바이어스 힌트(Intel 전용)입니다. Intel 프로세서의 MSR을 Intel에서 권장하는 값으로 설정합니다. Intel에서는 고성능의 경우 0, 균형 조정의 경우 6, 저전력의 경우 15를 권장합니다. 다른 값은 정의되어 있지 않습니다.

6 확인을 클릭합니다.

vSphere를 통한 메모리 가상화

8

메모리 리소스를 관리하려면 먼저 ESXi에서 어떻게 메모리 리소스를 가상화하고 사용하는지 이해해야 합니다.

VMkernel은 호스트의 모든 물리적 RAM을 관리합니다. VMkernel에서는 이 관리되는 물리적 RAM의 일부를 VMkernel 전용으로 지정합니다. 나머지 메모리는 가상 시스템에서 사용할 수 있습니다.

가상 및 물리적 메모리 공간은 페이지라는 블록으로 나뉩니다. 물리적 메모리가 가득 차면 물리적 메모리에 있지 않은 가상 페이지의 데이터가 디스크에 저장됩니다. 프로세서 아키텍처에 따라 페이지는 일반적으로 4KB나 2MB입니다. 고급 메모리 특성의 내용을 참조하십시오.

다음으로 아래 항목을 읽으십시오.

- 가상 시스템 메모리
- 메모리 오버 커밋
- 메모리 공유
- 메모리 가상화
- 하드웨어 지원 메모리 가상화
- 큰 페이지 크기 지원

가상 시스템 메모리

각 가상 시스템은 구성된 크기와 가상화를 위한 추가 오버헤드 메모리를 더한 크기만큼 메모리를 사용합니다.

구성된 크기는 게스트 운영 체제에 표시되는 메모리의 양입니다. 이 양은 가상 시스템에 할당된 물리적 RAM의 양과 다릅니다. 후자는 호스트의 메모리 부담 수준 및 리소스 설정(공유, 예약, 제한)에 따라 다릅니다.

예를 들어, 구성된 크기가 1GB인 가상 시스템을 가정해 볼 수 있습니다. 게스트 운영 체제가 부팅되면 이 운영 체제는 1GB의 물리적 메모리를 가진 전용 시스템에서 실행되고 있다고 감지합니다. 어떤 경우에는 가상 시스템이 1GB 전체에 할당될 수도 있고, 때에 따라서는 더 적은 할당을 받을 수도 있습니다. 실제 할당량에 상관없이 게스트 운영 체제는 1GB의 물리적 메모리를 가진 전용 시스템에서 실행되는 것처럼 동작합니다.

공유

예약보다 많은 경우 가상 시스템에 상대적인 우선 순위를 지정합니다.

예약

메모리가 오버 커밋되더라도 호스트가 가상 시스템을 위해 예약하는 물리적 RAM 양의 보장되는 하한 값입니다. 가상 시스템이 과도한 페이지징 없이 효율적으로 실행될 수 있을 만큼 충분한 메모리를 갖도록 예약을 설정하십시오.

가상 시스템이 해당 예약 내의 모든 메모리를 사용한 후에는 해당 양만큼의 메모리를 보유할 수 있으며 이 메모리는 가상 시스템이 유휴 상태인 경우에도 회수되지 않습니다. 일부 게스트 운영 체제(예: Linux)는 부팅 후 곧바로 구성된 모든 메모리에 액세스하지 않을 수 있습니다. 가상 시스템이 해당 예약 내의 모든 메모리를 사용하기 전까지 VMkernel이 해당 예약에서 사용되지 않은 부분을 다른 가상 시스템에 할당할 수 있습니다. 하지만 게스트의 워크로드가 증가하고 가상 시스템이 전체 예약을 사용한 후에는 이 메모리를 유지할 수 없습니다.

제한

호스트가 가상 시스템에 할당할 수 있는 물리적 RAM 양의 상한 값입니다. 가상 시스템의 메모리 할당은 구성된 크기에 의해서도 암시적으로 제한됩니다.

메모리 오버 커밋

실행 중인 각 가상 시스템에 대해 시스템은 가상 시스템의 예약(있는 경우) 및 가상화 오버헤드용으로 물리적 RAM을 예약합니다.

모든 가상 시스템의 구성된 총 메모리 크기가 호스트에서 사용 가능한 물리적 메모리 양을 초과할 수 있습니다. 하지만 이 상황이 메모리가 오버 커밋되었음을 의미하는 것만은 아닙니다. 모든 가상 시스템의 결합된 작업 메모리 공간이 호스트 메모리 크기의 결합된 작업 메모리 공간을 초과할 때 메모리가 오버 커밋됩니다.

ESXi 호스트가 사용하는 메모리 관리 기술 때문에 가상 시스템은 호스트에서 사용 가능한 물리적 RAM보다 많은 가상 RAM을 사용할 수 있습니다. 예를 들어 메모리가 2GB인 호스트가 있는 경우 메모리가 각각 1GB인 4개의 가상 시스템을 실행할 수 있습니다. 이 경우 메모리가 오버 커밋됩니다. 예를 들어 4개의 가상 시스템 모두가 유휴 상태인 경우 결합된 메모리 소비량은 2GB가 되지 않을 수 있습니다. 하지만 4GB 가상 시스템 모두가 활발하게 메모리를 사용 중이라면 메모리 공간이 2GB를 초과하여 ESXi 호스트가 오버 커밋될 수 있습니다.

일반적으로 가상 시스템에 따라 로드가 적기도 하고 많기도 해서 시간에 따라 상대적인 활동 수준이 달라질 수 있기 때문에 오버 커밋이 가능합니다.

메모리 사용률을 높이기 위해 ESXi 호스트는 유휴 가상 시스템의 메모리를 메모리가 더 필요한 가상 시스템으로 넘겨줍니다. 예약 또는 공유 매개 변수를 사용하면 중요한 가상 시스템에 메모리를 우선적으로 할당할 수 있습니다. 이러한 메모리는 사용되지 않을 경우 다른 가상 시스템에 계속 제공됩니다. ESXi는 호스트가 크게 메모리 오버 커밋되지 않은 경우라도 적절한 성능을 제공하기 위해 벌루닝, 메모리 공유, 메모리 압축 및 스와핑 등의 여러 메커니즘을 구현합니다.

메모리 오버 커밋된 환경에서 가상 시스템이 예약된 메모리를 모두 소비하면 ESXi 호스트에 메모리가 부족할 수 있습니다. 전원이 켜진 가상 시스템에는 영향이 없지만 메모리 부족 때문에 새 가상 시스템의 전원을 켜지 못할 수 있습니다.

참고 모든 가상 시스템 메모리 오버헤드 역시 예약된 것으로 간주됩니다.

또한 ESXi 호스트에서는 **메모리 압축**에 설명된 것처럼 메모리 압축이 기본적으로 사용되어 메모리가 오버 커밋될 경우 가상 시스템의 성능을 향상시킵니다.

메모리 공유

메모리 공유는 호스트에서 메모리 밀도를 높이는 데 도움이 되는 독점적인 ESXi 기술입니다.

메모리 공유는 몇몇 가상 시스템이 동일한 게스트 운영 체제의 인스턴스를 실행할 수 있다는 관찰에 의존합니다. 이러한 가상 시스템에는 동일한 애플리케이션 또는 구성 요소가 로드되어 있거나 공통 데이터가 포함되어 있을 수 있습니다. 이러한 경우 호스트는 독점적인 TPS(투명 페이지 공유) 기술을 사용하여 메모리 페이지의 중복되는 사본을 제거합니다. 메모리 공유를 사용할 경우, 가상 시스템에서 실행 중인 워크로드는 일반적으로 물리적 시스템에서 실행 중인 워크로드보다 소비하는 메모리 양이 적습니다. 결과적으로 높은 수준의 오버 커밋을 효율적으로 지원할 수 있습니다. 메모리 공유를 통해 절약되는 메모리의 양은 워크로드가 거의 동일한 시스템으로 구성되어 있는지 여부에 따라 달라집니다. 워크로드가 거의 동일한 시스템으로 구성되어 있는 경우에는 더 많은 메모리를 확보할 수 있습니다. 워크로드가 다양한 경우에는 메모리 절약 비율이 낮아질 수 있습니다.

참고 보안 문제로 인해 가상 시스템 간 투명 페이지 공유는 기본적으로 비활성화되고 페이지 공유는 가상 시스템 내 메모리 공유로 제한됩니다. 페이지 공유는 가상 시스템 간에는 발생하지 않고 가상 시스템 내에서만 발생합니다. 자세한 내용은 [가상 시스템 사이에 메모리 공유의 내용을 참조하십시오](#).

메모리 가상화

가상화를 통해 추가적인 수준의 메모리 매핑이 적용되기 때문에 ESXi는 모든 가상 시스템에 대해 효과적으로 메모리를 관리할 수 있습니다.

가상 시스템의 물리적 메모리 일부는 매핑되지 않거나 스왑되지 않은 페이지로 매핑되거나 공유 페이지로 매핑될 수 있습니다.

호스트는 게스트 운영 체제를 인식하지 않으며 게스트 운영 체제의 자체 메모리 관리 하위 시스템을 방해하지 않고 가상 메모리 관리를 수행합니다.

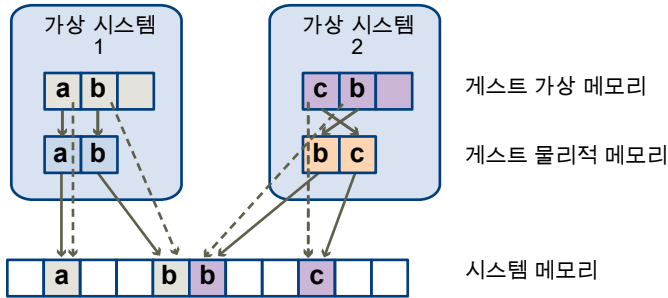
각 가상 시스템의 VMM은 게스트 운영 체제의 물리적 메모리 페이지와 기반 시스템의 물리적 메모리 페이지 간 매핑을 유지합니다. (VMware에서는 기반 호스트의 물리적 페이지를 “시스템” 페이지라고 하고 게스트 운영 체제의 물리적 페이지를 “물리적” 페이지라고 표현합니다.)

각 가상 시스템에게는 0부터 시작하는 연속적 주소 지정이 가능한 물리적 메모리 공간이 보입니다. 하지만 각 가상 시스템에서 사용하는 서버의 기반 시스템 메모리는 연속적일 필요가 없습니다.

게스트 가상 대 게스트 물리적 주소는 게스트 운영 체제에서 관리됩니다. 하이퍼바이저는 게스트 물리적 주소를 시스템 주소로 변환하는 일만 담당합니다. 하드웨어 지원 메모리 가상화는 하드웨어 기능을 활용하여 게스트의 페이지 테이블과 하이퍼바이저가 유지하는 중첩된 페이지 테이블이 포함된 결합된 매핑을 생성합니다.

이 그림에서는 메모리 가상화의 ESXi 구현을 보여 줍니다.

그림 8-1. ESXi 메모리 매핑



- 상자는 페이지를 나타내며 화살표는 여러 메모리 매핑을 보여 줍니다.
- 게스트 가상 메모리에서 게스트 물리적 메모리로의 화살표는 게스트 운영 체제에서 페이지 테이블에 유지되는 매핑을 보여 줍니다. x86 아키텍처 프로세서에 대한 가상 메모리에서 선형 메모리로의 매핑은 표시되어 있지 않습니다.
- 게스트 물리적 메모리에서 시스템 메모리로의 화살표는 VMM에서 유지하는 매핑을 보여 줍니다.
- 점선 화살표는 마찬가지로 VMM에서 유지하는 새도우 페이지 테이블에서 게스트 가상 메모리로부터 시스템 메모리로의 매핑을 보여 줍니다. 가상 시스템을 실행하는 기본 프로세서는 새도우 페이지 테이블 매핑을 사용합니다.

하드웨어 지원 메모리 가상화

AMD SVM-V 및 Intel Xeon 5500 시리즈와 같은 일부 CPU는 두 계층의 페이지 테이블을 사용하여 메모리 가상화에 대한 하드웨어 지원을 제공합니다.

참고 이 항목에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

첫 번째 페이지 테이블 계층에는 게스트 V2P(가상에서 물리) 변환이 저장되며 두 번째 페이지 테이블 계층에는 게스트 P2M(물리에서 시스템) 변환이 저장됩니다. TLB(Translation Look-aside Buffer)는 프로세서의 MMU(메모리 관리 유닛) 하드웨어가 유지하는 변환의 캐시입니다. TLB 누락은 이 캐시의 누락이며 하드웨어는 필요한 변환을 찾기 위해 메모리로 이동해야 할 경우가 많습니다. 특정 게스트 가상 주소에 대한 TLB 누락의 경우 하드웨어는 게스트 가상 주소를 시스템 주소로 변환하기 위해 두 페이지 테이블을 모두 확인합니다. 페이지 테이블의 첫 번째 계층은 게스트 운영 체제에 의해 유지됩니다. VMM은 페이지 테이블의 두 번째 계층만 유지합니다.

성능 고려 사항

하드웨어 지원을 사용하면 소프트웨어 메모리 가상화의 오버헤드가 제거됩니다. 특히 하드웨어 지원은 새도우 페이지 테이블과 게스트 페이지 테이블의 동기화 상태 유지에 필요한 오버헤드를 제거합니다. 하지만 하드웨어 지원을 사용할 때는 TLB 누락 지연 시간이 조금 더 높습니다. 기본적으로 하이퍼바이저는 TLB 누락의 비용을 줄이기 위해 하드웨어 지원 모드에서 큰 페이지를 사용합니다. 이로 인해 하드웨어 지원을 사용할 때 워크로드에 이점이 있는지 여부는 소프트웨어 메모리 가상화를 사용할 때 메모리 가상화가 초래하는 오버헤드에 따라 달라집니다. 프로세스 생성, 메모리 매핑 또는 컨텍스트 전환 등과 같이 페이지 테이블 작업의 양이 적은 워크로드의 경우에는 소프트웨어 가상화로 인해 발생하는 오버헤드가 크지 않습니다. 이와 반대로 페이지 테이블 작업의 양이 많은 워크로드의 경우 하드웨어 지원을 사용하면 이점을 얻을 수 있습니다.

기본적으로 하이퍼바이저는 TLB 누락의 비용을 줄이기 위해 하드웨어 지원 모드에서 큰 페이지를 사용합니다. 게스트 가상 대 게스트 물리 및 게스트 물리 대 시스템 주소 변환 모두에 큰 페이지를 사용하면 가장 높은 성능이 구현됩니다.

LPage.LPageAlwaysTryForNPT 옵션은 게스트 물리 대 시스템 주소 변환에 큰 페이지를 사용하기 위한 정책을 변경할 수 있습니다. 자세한 내용은 [고급 메모리 특성](#) 항목을 참조하십시오.

큰 페이지 크기 지원

ESXi는 큰 페이지 크기를 제한적으로 지원합니다.

x86 아키텍처에서는 시스템 소프트웨어가 4KB, 2MB 및 1GB 페이지를 사용할 수 있습니다. 4KB 페이지는 작은 페이지라고 하며, 2MB 및 1GB 페이지는 큰 페이지라고 합니다. 큰 페이지는 TLB(Translation Lookaside Buffer) 부담을 줄이고 페이지 테이블 처리 비용을 줄여 워크로드 성능을 향상시킵니다.

가상화된 환경에서 큰 페이지는 하이퍼바이저와 게스트 운영 체제가 독립적으로 사용할 수 있습니다. 게스트와 하이퍼바이저에서 모두 큰 페이지를 사용할 경우 성능 향상 효과가 가장 크지만, 대부분의 경우 하이퍼바이저 수준에서만 큰 페이지를 사용하는 경우에도 성능 향상이 나타날 수 있습니다.

ESXi 하이퍼바이저는 기본적으로 게스트 vRAM에 2MB 페이지를 사용합니다. vSphere ESXi는 게스트 vRAM에 1GB 페이지를 사용하는 것을 제한적으로 지원합니다. 자세한 내용은 "게스트 vRAM에 1GB 페이지 사용" 을 참조하십시오.

vSphere를 사용하여 메모리 리소스 관리

9

vSphere Client를 사용하면 메모리 할당 설정에 대한 정보를 보고 변경할 수 있습니다. 메모리 리소스를 효과적으로 관리하려면 메모리 오버헤드, 유틸리티 메모리 세율, 그리고 ESXi 호스트의 메모리 회수 방법을 잘 알고 있어야 합니다.

메모리 리소스를 관리할 때 메모리 할당을 지정할 수 있습니다. 메모리 할당을 사용자 지정하지 않으면 ESXi 호스트는 대부분의 환경에 적합한 기본값을 사용합니다.

메모리 할당은 다양한 방법으로 지정할 수 있습니다.

- vSphere Client를 통해 사용 가능한 특성과 특수 기능을 사용합니다. vSphere Client를 사용하여 ESXi 호스트나 vCenter Server 시스템에 연결할 수 있습니다.
- 고급 설정을 사용합니다.
- 스크립트로 작성된 메모리 할당에는 vSphere SDK를 사용합니다.

참고 이 장에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- 메모리 오버헤드 이해
- 가상 시스템의 오버헤드 메모리
- ESXi 호스트의 메모리 할당 방법
- 유틸리티 가상 시스템의 메모리 세율
- VMX 스왑 파일
- 메모리 회수
- 메모리 벌룬 드라이버
- 가상 시스템 사이에 메모리 공유
- 메모리 압축
- 메모리 압축 캐시 활성화 또는 비활성화
- 메모리 압축 캐시의 최대 크기 설정
- 메모리 사용 유형 측정 및 차별화

- 메모리 안정성
- 오류 분리 알림 수정
- 시스템 스왑
- 시스템 스왑 구성

메모리 오버헤드 이해

메모리 리소스의 가상화는 어느 정도의 오버헤드와 연관이 있습니다.

ESXi 가상 시스템에서는 두 가지 종류의 메모리 오버헤드가 발생할 수 있습니다.

- 가상 시스템 내에서 메모리에 액세스하기 위한 추가 시간
- 각 가상 시스템에 할당된 메모리 외에도, ESXi 호스트에서 자체 코드 및 데이터 구조에 추가 공간 필요

ESXi 메모리 가상화로 인해 메모리 액세스에 약간의 시간 오버헤드가 추가됩니다. 프로세서의 페이징 하드웨어는 페이지 테이블(소프트웨어 기반 방식의 경우 새도우 페이지 테이블 또는 하드웨어 지원 방식의 경우 2 수준 페이지 테이블)을 직접 사용하므로 가상 시스템에서 대부분의 메모리 액세스는 주소 변환 오버헤드 없이 실행될 수 있습니다.

메모리 공간 오버헤드의 두 구성 요소는 다음과 같습니다.

- VMkernel에 대해 고정된 시스템 전체 오버헤드
- 각 가상 시스템에 대한 추가적인 오버헤드

오버헤드 메모리에는 새도우 페이지 테이블과 같이 가상 시스템 프레임 버퍼 및 여러 가상화 데이터 구조용으로 예약된 공간이 포함됩니다. 오버헤드 메모리는 가상 CPU의 수와 게스트 운영 체제용으로 구성된 메모리에 따라 달라집니다.

가상 시스템의 오버헤드 메모리

가상 시스템은 특정 양의 가용 오버헤드 메모리 전원이 켜져 있어야 합니다. 이러한 오버헤드 양을 알고 있어야 합니다.

가상 시스템에 필요한 오버헤드 메모리의 양은 vCPU의 개수와 메모리 크기, 디바이스의 유형과 개수, 모니터의 실행 모드, 가상 시스템의 하드웨어 버전 등을 포함하여 다양한 요인에 따라 달라집니다. 현재 사용 중인 vSphere 버전도 필요한 메모리 양에 영향을 줄 수 있습니다. VMX는 가상 시스템에 필요한 오버헤드 메모리 양을 자동으로 계산합니다.

특정 구성에 필요한 오버헤드 메모리 양을 확인하려면 먼저 대상 가상 시스템의 전원을 켜야 합니다. 그런 다음 vmware.log 파일을 살펴봅니다. 가상 시스템의 전원이 켜지면 필요한 오버헤드 메모리의 양이 로그에 출력됩니다. 로그 내에서 VMEM을 검색하여 가상 메모리에 대해 예약된 오버헤드 메모리의 초기 양과 정확한 양을 확인하십시오.

ESXi 호스트의 메모리 할당 방법

호스트에서는 메모리가 오버 커밋된 경우가 아니면 `Limit` 매개 변수로 지정된 메모리를 각 가상 시스템에 할당합니다. ESXi에서는 지정된 물리적 메모리 크기보다 많은 메모리를 가상 시스템에 할당하지 않습니다.

예를 들어 1GB 가상 시스템에는 기본 제한(무제한) 또는 사용자 지정 제한(예: 2GB)이 있을 수 있습니다. 두 경우 모두 ESXi 호스트는 해당 호스트에 지정된 물리적 메모리 크기인 1GB를 초과하여 할당하지 않습니다.

메모리가 오버 커밋된 경우에는 **예약**으로 지정된 양과 **제한**으로 지정된 양 사이의 메모리 양을 각 가상 시스템에 할당합니다. 예약된 양을 초과하여 가상 시스템에 부여된 메모리의 양은 대부분 현재 메모리 로드에서 따릅니다.

호스트는 할당된 공유 수 및 최근 작업 설정 크기의 예상값에 기반하여 각 가상 시스템에 대한 할당을 결정합니다.

- 공유 - ESXi 호스트는 수정된 비례적 공유 메모리 할당 정책을 사용합니다. 가상 시스템은 메모리 공유를 통해 사용 가능한 물리적 메모리의 일부를 사용할 수 있습니다.
- 작업 설정 크기 - ESXi 호스트는 가상 시스템의 연속 실행 시간 동안 메모리 작업을 모니터링하여 가상 시스템의 작업 설정을 예상합니다. 작업 설정 크기의 증가에는 빠르게 반응하고 작업 설정 크기 감소에는 상대적으로 느리게 반응하는 기술을 사용하여 이 예상값을 몇 번의 기간에 걸쳐 평균화합니다.

이 방법은 유휴 메모리가 회수된 가상 시스템의 메모리 사용이 증가하기 시작하면 신속하게 전체 공유 기반 할당으로 전환할 수 있도록 합니다.

메모리 작업을 모니터링하여 60초의 기본 기간 동안 작업 설정 크기를 예측합니다. 이 기본값을 수정하려면 `Mem.SamplePeriod` 고급 설정을 조정합니다. [고급 호스트 특성 설정 항목](#)을 참조하십시오.

유휴 가상 시스템의 메모리 세율

가상 시스템이 현재 할당된 메모리 중 일부만 사용하고 있으면 ESXi는 사용 중인 메모리보다 유휴 메모리에 대해 더 많이 과세합니다. 이는 가상 시스템이 유휴 메모리를 비축하는 것을 방지하기 위해서입니다.

유휴 메모리 세율은 누진적 방식으로 적용됩니다. 가상 시스템의 활성 메모리에 대한 유휴 메모리의 비율이 높아지면 유휴 세율이 늘어납니다. 계층적 리소스 풀을 지원하지 않는 이전 버전의 ESXi에서는 가상 시스템의 모든 유휴 메모리가 동일하게 과세되었습니다.

`Mem.IdleTax` 옵션을 사용하여 유휴 메모리 세율을 수정할 수 있습니다. `Mem.SamplePeriod` 고급 특성과 함께 이 옵션을 사용하면 시스템이 가상 시스템에 대한 목표 메모리 할당을 결정하는 방법을 제어할 수 있습니다. [고급 호스트 특성 설정 항목](#)을 참조하십시오.

참고 대부분의 경우 `Mem.IdleTax`를 변경하는 것은 불필요하며 적절하지도 않습니다.

VMX 스왑 파일

VMX(가상 시스템 실행 파일) 스왑 파일을 사용하면 호스트에서 VMX 프로세스용으로 예약되는 오버헤드 메모리의 양을 대폭 줄일 수 있습니다.

참고 VMX 스왑 파일은 호스트 스왑 캐시로 스왑 기능이나 일반적인 호스트 수준 스왑 파일과 관련이 없습니다.

ESXi에서는 각 가상 시스템에 대해 다양한 용도의 메모리를 예약합니다. VMM(가상 시스템 모니터) 및 가상 디바이스와 같은 일부 구성 요소에 필요한 메모리는 가상 시스템의 전원을 켤 때 완전히 예약됩니다. 그러나 VMX 프로세스용으로 예약되는 오버헤드 메모리의 일부는 스왑할 수 있습니다. VMX 스왑 기능은 예를 들어 가상 시스템당 약 50MB 이상의 메모리를 가상 시스템당 약 10MB로 줄이는 등 VMX 메모리 예약 크기를 상당히 줄여 줍니다. 이 경우 호스트 메모리가 오버 커밋될 때 나머지 메모리를 스왑 아웃할 수 있으므로 각 가상 시스템의 오버헤드 메모리 예약 크기가 줄어듭니다.

가상 시스템의 전원을 켤 때 사용 가능한 디스크 공간이 충분히 있으면 호스트에서는 VMX 스왑 파일을 자동으로 만듭니다.

메모리 회수

ESXi 호스트는 가상 시스템에서 메모리를 회수할 수 있습니다.

호스트는 예약으로 지정된 메모리 양을 직접 가상 시스템에 할당합니다. 예약을 초과하는 메모리는 호스트의 물리적 리소스를 사용하여 할당하며, 물리적 리소스를 사용할 수 없으면 벌루닝 또는 스와핑 같은 특수 기술을 사용하여 처리합니다. 가상 시스템에 할당된 메모리 양을 동적으로 확장하거나 줄이기 위해 호스트는 두 가지 기술을 사용할 수 있습니다.

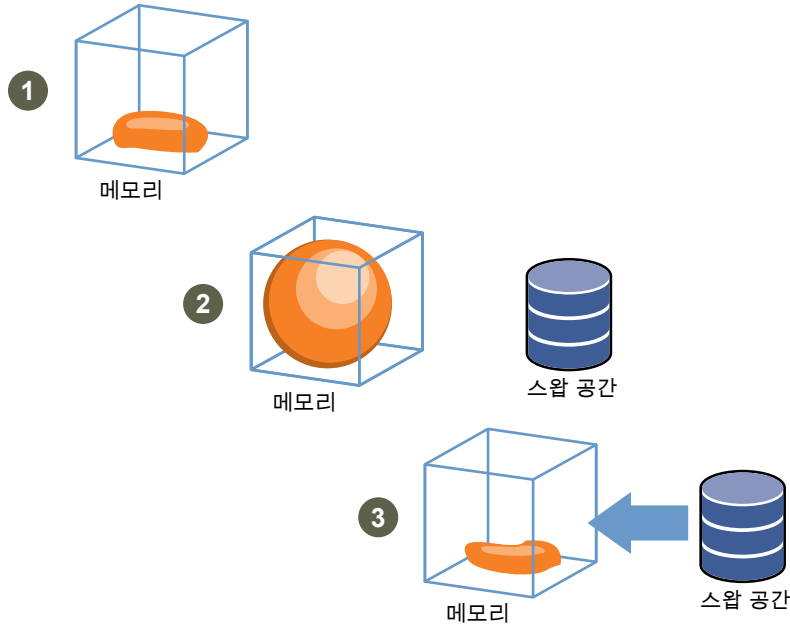
- ESXi 시스템은 가상 시스템에서 실행되고 있는 게스트 운영 체제로 로드되는 메모리 벌룬 드라이버(vmmemctl)를 사용합니다. **메모리 벌룬 드라이버**를 참조하십시오.
- ESXi 시스템은 게스트 운영 체제의 개입 없이 페이지를 가상 시스템에서 서버 스왑 파일로 스왑합니다. 각 가상 시스템에는 자체의 스왑 파일이 있습니다.

메모리 벌룬 드라이버

메모리 벌룬 드라이버(vmmemctl)는 서버와 공동으로 작업하여 게스트 운영 체제에서 가장 중요하지 않은 것으로 판단되는 페이지를 회수합니다.

이 드라이버는 유사한 메모리 제약 조건에서 네이티브 시스템의 동작과 거의 일치하는 예측 가능한 성능을 제공하는 독점적인 벌루닝 기술을 사용합니다. 이 기술은 게스트 운영 체제의 메모리 부담을 늘리거나 줄여서 게스트가 자체의 네이티브 메모리 관리 알고리즘을 사용하도록 합니다. 메모리가 빠듯하면 게스트 운영 체제는 회수할 페이지를 결정하고 필요한 경우 자체의 가상 디스크로 페이지를 스와핑합니다.

그림 9-1. 게스트 운영 체제에서의 메모리 벌루닝



참고 게스트 운영 체제는 충분한 스왑 공간으로 구성해야 합니다. 일부 게스트 운영 체제에는 추가 제한 사항이 있습니다.

필요한 경우 특정 가상 시스템에 대해 `sched.mem.maxmemctl` 매개 변수를 설정하여 `vmmemctl`에서 회수하는 메모리 양을 제한할 수 있습니다. 이 옵션은 가상 시스템에서 회수할 수 있는 메모리의 최대 양을 MB 단위로 지정합니다. 고급 가상 시스템 특성 설정 항목을 참조하십시오.

가상 시스템 사이에 메모리 공유

가상 시스템 간에서 또는 단일 가상 시스템 내에서 메모리를 공유하기에 적합한 ESXi 워크로드가 많습니다.

ESXi 메모리 공유는 공유 가능성을 시간에 따라 검사하는 백그라운드 작업으로 실행됩니다. 절약되는 메모리의 양은 시간에 따라 달라집니다. 비교적 일정한 워크로드의 경우 일반적으로 모든 공유 가능성을 활용할 때까지 절약되는 양이 서서히 증가합니다.

지정된 워크로드에 대한 메모리 공유의 효과를 확인하려면 워크로드를 실행한 후 `resxtop` 또는 `esxtop`를 사용하여 실제 절감되는 양을 관찰합니다. 메모리 페이지에 있는 대화식 모드의 `PSHARE` 필드에서 정보를 찾을 수 있습니다.

시스템이 메모리를 공유할 수 있는 가능성이 있는지 확인하기 위해 메모리를 검색하는 속도를 제어하려면 `Mem.ShareScanTime` 및 `Mem.ShareScanGHz` 고급 설정을 사용하십시오.

`sched.mem.pshare.enable` 옵션을 설정하여 개별 가상 시스템에 대해 공유 기능을 구성할 수도 있습니다.

보안 문제로 인해 가상 시스템 간 투명 페이지 공유는 기본적으로 비활성화되고 페이지 공유는 가상 시스템 내 메모리 공유로 제한됩니다. 다시 말해 페이지 공유는 가상 시스템 간에는 적용되지 않고 가상 시스템 내에서만 적용됩니다. 투명 페이지 공유로 인해 발생하는 보안 영향으로 인한 시스템 관리자의 우려 사항을 해결하기 위해 솔팅 개념이 도입되었습니다. 솔팅을 사용하면 투명 페이지 공유에 참여하는 가상 시스템을 이전에 비해 더 세부적으로 관리할 수 있습니다. 새로운 솔팅 설정을 사용하면 솔트 값과 페이지의 콘텐츠가 동일한 경우에만 가상 시스템이 페이지를 공유할 수 있습니다. 새로운 호스트 구성 옵션인 **Mem.ShareForceSalting**을 구성하여 솔팅을 활성화하거나 비활성화할 수 있습니다.

고급 옵션을 설정하는 방법은 [장 22 vSphere의 고급 특성](#)의 내용을 참조하십시오.

메모리 압축

ESXi는 사용자가 메모리 오버 커밋을 사용할 경우 가상 시스템 성능을 향상시키는 메모리 압축 캐시를 제공합니다. 메모리 압축은 기본적으로 활성화되어 있습니다. 호스트의 메모리가 오버 커밋되면 ESXi는 가상 페이지를 압축하여 이를 메모리에 저장합니다.

압축된 메모리에 액세스하는 것이 디스크에 스와핑된 메모리에 액세스하는 것보다 빠르므로 ESXi의 메모리 압축을 사용하면 큰 성능 저하 없이 메모리를 오버 커밋할 수 있습니다. 가상 페이지를 스와핑해야 하는 경우 ESXi는 먼저 페이지 압축을 시도합니다. 2KB 이하로 압축될 수 있는 페이지는 가상 시스템의 압축 캐시에 저장되므로 호스트의 용량이 늘어납니다.

vSphere Client의 [고급 설정] 대화상자에서 압축 캐시의 최대 크기를 설정하고 메모리 압축을 비활성화할 수 있습니다.

메모리 압축 캐시 활성화 또는 비활성화

메모리 압축은 기본적으로 활성화되어 있습니다. vSphere Client에서 고급 시스템 설정을 사용하여 호스트에 대한 메모리 압축을 활성화하거나 비활성화할 수 있습니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 구성을 클릭합니다.
- 3 시스템에서 **고급 시스템 설정**을 선택합니다.
- 4 Mem.MemZipEnable을 찾아 **편집** 버튼을 클릭합니다.
- 5 메모리 압축 캐시를 활성화하려면 1을 입력하고 비활성화하려면 0을 입력합니다.
- 6 **확인**을 클릭합니다.

메모리 압축 캐시의 최대 크기 설정

호스트의 가상 시스템에 대한 메모리 압축 캐시의 최대 크기를 설정할 수 있습니다.

가상 시스템 메모리 크기의 백분율로 압축 캐시 크기를 설정합니다. 예를 들어 가상 시스템의 메모리 크기가 1000MB인 경우 20을 입력하면 ESXi가 최대 200MB의 호스트 메모리를 가상 시스템의 압축 페이지를 저장하는 데 사용합니다.

압축 캐시 크기를 설정하지 않으면 ESXi가 기본값인 10%를 사용합니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 구성을 클릭합니다.
- 3 시스템에서 고급 시스템 설정을 선택합니다.
- 4 Mem.MemZipMaxPct를 찾고 편집 버튼을 클릭합니다.
이 특성 값은 가상 시스템에 대한 압축 캐시의 최대 크기를 결정합니다.
- 5 압축 캐시의 최대 크기를 입력하십시오.
값은 가상 시스템 메모리 크기의 백분율이며 5-100% 사이여야 합니다.
- 6 확인을 클릭합니다.

메모리 사용 유형 측정 및 차별화

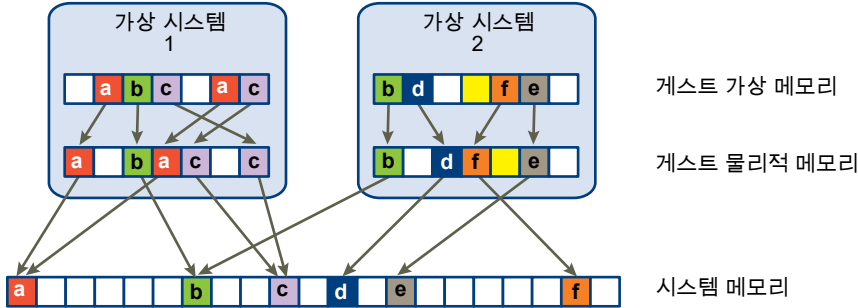
vSphere Client의 성능 탭에는 메모리 사용량을 분석하는 데 사용할 수 있는 몇 가지 메트릭이 표시됩니다.

메모리 메트릭 중 일부는 게스트 물리적 메모리를 측정하고 일부는 시스템 메모리를 측정합니다. 예를 들어 성능 메트릭을 사용하여 확인할 수 있는 두 가지 메모리 사용 유형에는 게스트 물리적 메모리와 시스템 메모리가 있습니다. 게스트 물리적 메모리는 부여된 메모리 메트릭(가상 시스템용) 또는 공유 메모리(호스트용)를 사용하여 측정할 수 있습니다. 반면에, 시스템 메모리는 메모리 사용량(가상 시스템용) 또는 공통 공유 메모리(호스트용)를 사용하여 측정합니다. 이러한 메모리 사용 유형의 개념적 차이를 이해하는 것은 각 메트릭의 측정 내용과 그 해석 방식을 파악하는 데 중요합니다.

VMkernel은 게스트 물리적 메모리를 시스템 메모리에 매핑하지만 이러한 매핑이 항상 일대일로 이루어지는 것은 아닙니다. 게스트 물리적 메모리의 여러 영역이 시스템 메모리의 동일한 영역에 매핑되거나(메모리 공유의 경우), 게스트 물리적 메모리의 특정 영역이 시스템 메모리에 매핑되지 않을 수 있습니다(VMkernel이 게스트 물리적 메모리를 스왑 아웃하거나 버퍼링하는 경우). 이러한 상황에서는 개별 가상 시스템 또는 호스트의 게스트 물리적 메모리 사용 및 시스템 메모리 사용에 대한 계산이 다를 수 있습니다.

다음 그림의 예를 고려해 봅니다. 이 그림은 한 호스트에서 실행되는 두 가상 시스템을 보여 줍니다. 각 블록은 4KB의 메모리를 나타내고 각 색상/문자는 한 블록에 있는 서로 다른 데이터 집합을 나타냅니다.

그림 9-2. 메모리 사용 예



가상 시스템의 성능 메트릭은 다음과 같이 결정될 수 있습니다.

- 가상 시스템 1에 부여된 메모리(시스템 메모리에 매핑되는 게스트 물리적 메모리의 양)를 결정하려면 가상 시스템 1에서 시스템 메모리에 대한 화살표가 있는 게스트 물리적 메모리의 블록 수를 세어서 4KB를 곱합니다. 화살표가 있는 블록이 5개이므로 부여된 메모리는 20KB입니다.
- 메모리 사용량은 가상 시스템에 할당된 시스템 메모리의 양으로, 공유 메모리로 인해 절약되는 메모리를 감안합니다. 먼저 가상 시스템 1의 게스트 물리적 메모리로부터 화살표가 있는 시스템 메모리의 블록 수를 셉니다. 이러한 블록은 3개지만 그중 하나는 가상 시스템 2와 공유됩니다. 따라서 완전한 블록 2개와 1/2 블록을 더하고 여기에 4KB를 곱하면 메모리 사용량은 총 10KB가 됩니다.

이 두 메트릭의 중요한 차이점은 부여된 메모리는 게스트 물리적 메모리 수준에서 화살표가 있는 블록 수를 세는 반면 메모리 사용량은 시스템 메모리 수준에서 화살표가 있는 블록 수를 셉는 것입니다. 두 수준 간에 블록 수가 차이가 나는 것은 메모리 공유 때문이며, 이로 인해 부여된 메모리와 메모리 사용량이 서로 차이가 납니다. 메모리는 공유나 다른 회수 기술을 통해 절약됩니다.

호스트에 대한 공유 메모리와 공통 공유 메모리를 결정할 때도 유사한 결과가 나타납니다.

- 호스트의 공유 메모리는 각 가상 시스템의 공유 메모리를 합한 것입니다. 공유 메모리를 계산하려면 각 가상 시스템의 게스트 물리적 메모리를 기준으로 시스템 메모리 블록(해당 시스템 메모리 블록은 둘 이상의 화살표를 받아야 함)을 가리키는 화살표가 있는 블록 수를 셉니다. 이 예에서는 이러한 블록이 6개이므로 호스트의 공유 메모리는 24KB입니다.
- 공통 공유 메모리는 가상 시스템에 의해 공유되는 시스템 메모리의 양입니다. 공통 메모리를 결정하려면 시스템 메모리를 기준으로 둘 이상의 화살표가 가리키는 블록 수를 셉니다. 이러한 블록은 3개이므로 공통 공유 메모리는 12KB입니다.

공유 메모리는 게스트 물리적 메모리와 관련되어 있으므로 화살표의 시작점을 살펴보아야 합니다. 하지만 공통 공유 메모리는 시스템 메모리를 나타내므로 화살표의 끝점을 살펴보아야 합니다.

게스트 물리적 메모리와 시스템 메모리를 측정하는 메모리 메트릭은 서로 모순되는 것처럼 보입니다. 사실 이들 메트릭은 가상 시스템의 메모리 사용량을 서로 다른 측면에서 측정합니다. 이들 메트릭의 차이점을 이해함으로써 성능 문제를 진단하는 데 이를 보다 잘 사용할 수 있습니다.

메모리 안정성

오류 분리라고도 하는 메모리 안정성 기능을 사용하면 ESXi에서 장애가 발생했을 경우는 물론 장애가 발생할 것으로 판단되는 경우에도 메모리 사용을 일부 중지할 수 있습니다.

특정 주소에서 수정된 오류가 어느 정도 충분히 보고되면 ESXi는 이 주소의 사용을 중지하여 수정된 오류가 수정되지 않은 오류가 되지 않도록 합니다.

메모리 안정성 기능은 RAM의 수정된 오류와 수정되지 않은 오류에도 불구하고 더 나은 VMkernel 안정성을 제공합니다. 또한 오류가 포함되었을 수 있는 메모리 페이지를 시스템에서 사용하지 않도록 합니다.

오류 분리 알림 수정

메모리 안정성 기능을 사용하면 VMkernel은 오류 분리 알림이 나타나는 페이지의 사용을 중지합니다.

VMkernel이 수정할 수 없는 메모리 오류로부터 복구되거나, VMkernel이 수정 가능한 오류의 수가 많아서 시스템 메모리의 상당 부분을 회수하거나, 회수할 수 없는 페이지 수가 많은 경우, vSphere Client에서 사용자에게 이벤트가 수신됩니다.

절차

- 1 호스트를 비웁니다.
- 2 가상 시스템을 마이그레이션합니다.
- 3 메모리 관련 하드웨어 테스트를 실행합니다.

시스템 스왑

시스템 스왑은 전체 시스템의 사용되지 않는 메모리 리소스를 활용할 수 있는 메모리 회수 프로세스입니다.

시스템 스왑을 사용하면 시스템이 가상 시스템이 아닌 메모리 소비자로부터 메모리를 회수할 수 있습니다. 시스템 스왑이 사용하도록 설정되면 다른 프로세스로부터 메모리 회수의 영향을 가급적으로 줄이면서 이 메모리를 사용할 수 있는 가상 시스템에 할당할 수 있습니다. 시스템 스왑에 필요한 공간 크기는 1GB입니다.

메모리 회수는 메모리에서 데이터를 가져와 백그라운드 스토리지에 쓰는 방식으로 이루어집니다. 백그라운드 스토리지에서 데이터에 액세스하면 메모리에서 데이터에 액세스하는 경우보다 속도가 느리므로 스와핑되는 데이터를 어디에 저장할 것인지 신중하게 선택해야 합니다.

ESXi는 시스템 스왑을 저장할 위치를 자동으로 결정하며 이 위치가 **기본 설정 스왑 파일 위치**입니다. 이러한 결정은 특정 옵션 모음을 선택하여 지원받을 수 있습니다. 시스템은 가장 사용 가능성이 높은 옵션을 선택합니다. 실행할 수 있는 옵션이 없는 경우 시스템 스왑이 활성화되지 않습니다.

사용 가능한 옵션은 다음과 같습니다.

- 데이터스토어 - 지정된 데이터스토어 사용을 허용합니다. 시스템 스왑 파일에는 vSAN 데이터스토어 또는 VMware vSphere® Virtual Volumes™ 데이터스토어를 지정할 수 없습니다.
- 호스트 스왑 캐시 - 호스트 스왑 캐시의 일부 사용을 허용합니다.

- 기본 설정 스왑 파일 위치 - 호스트에 구성된 기본 설정 스왑 파일 위치 사용을 허용합니다.

시스템 스왑 구성

시스템 스왑 위치를 결정하는 옵션을 사용자 지정할 수 있습니다.

사전 요구 사항

시스템 스왑 설정 편집 대화상자에서 **사용**을 선택합니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 **구성**을 클릭합니다.
- 3 **시스템**에서 **시스템 스왑**을 선택합니다.
- 4 **편집**을 클릭합니다.
- 5 사용하도록 설정할 각 옵션의 확인란을 선택합니다.
- 6 **데이터스토어** 옵션을 선택한 경우 드롭다운 메뉴에서 데이터스토어를 선택합니다.
- 7 **확인**을 클릭합니다.

게스트 스왑 파일의 위치를 지정하고, 메모리가 오버 커밋될 경우 스왑 공간을 예약하고, 스왑 파일을 삭제할 수 있습니다.

`vmemctl` 드라이버를 사용할 수 없거나 응답이 없는 경우 ESXi 호스트는 스와핑을 사용하여 강제로 가상 시스템의 메모리를 회수합니다.

- 설치되지 않았습니다.
- 명시적으로 비활성화되었습니다.
- 실행되고 있지 않습니다(예: 게스트 운영 체제가 부팅 중인 경우).
- 현재 시스템 요구를 충족할 수 있는 메모리를 일시적으로 회수할 수 없습니다.
- 올바르게 작동하고 있지만 최대 벌룬 크기에 도달했습니다.

가상 시스템이 페이지를 필요로 할 때 표준 요구 페이지징 기술이 페이지를 다시 스왑 인합니다.

다음으로 아래 항목을 읽으십시오.

- 스왑 파일 위치
- DRS 클러스터에 대해 호스트-로컬 스왑 사용
- 독립 실행형 호스트에 대해 호스트-로컬 스왑 사용
- 스왑 공간 및 메모리 오버 커밋
- 호스트에 대한 가상 시스템 스왑 파일 속성 구성
- 클러스터의 가상 시스템 스왑 파일 위치 구성
- 스왑 파일 삭제

스왑 파일 위치

기본적으로 스왑 파일은 가상 시스템의 구성 파일과 동일한 위치에 생성됩니다. 이 위치는 VMFS 데이터스토어, vSAN 데이터스토어 또는 VMware vSphere® Virtual Volumes™ 데이터스토어 중 하나일 수 있습니다. vSAN 데이터스토어나 VVol 데이터스토어에서는 스왑 파일이 별도의 vSAN 또는 VVol 개체로 생성됩니다.

가상 시스템 전원을 켜면 ESXi 호스트가 스왑 파일을 생성합니다. 이 파일을 생성할 수 없는 경우 가상 시스템 전원이 켜지지 않습니다. 기본값을 사용하는 대신 다음 옵션을 사용할 수도 있습니다.

- 가상 시스템별 구성 옵션을 사용하여 데이터스토어를 다른 공유 스토리지 위치로 변경할 수 있습니다.
- 호스트에 로컬로 저장되는 데이터스토어를 지정할 수 있는 호스트-로컬 스왑을 사용할 수 있습니다. 이렇게 하면 호스트 단위 수준에서 스왑을 수행하여 SAN 공간을 절약할 수 있지만 소스 호스트의 로컬 스왑 파일로 스왑되는 페이지는 네트워크를 통해 대상 호스트로 전송되어야 하므로 vSphere vMotion의 성능이 다소 저하될 수 있습니다. 현재는 호스트-로컬 스왑에 vSAN 및 VVol 데이터스토어를 지정할 수 없습니다.

DRS 클러스터에 대해 호스트-로컬 스왑 사용

호스트-로컬 스왑 기능을 사용하면 호스트에 로컬로 저장된 데이터스토어를 스왑 파일 위치로 지정할 수 있습니다. DRS 클러스터에 대해 호스트-로컬 스왑을 사용하도록 설정합니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 **구성**을 클릭합니다.
- 3 **구성**에서 **일반**을 선택하여 스왑 파일 위치를 보고 위치를 변경하려면 **편집**을 클릭합니다.
- 4 **호스트에서 지정한 데이터스토어** 옵션을 선택하고 **확인**을 클릭합니다.
- 5 vSphere Client에서 클러스터의 호스트 중 하나를 찾습니다.
- 6 **구성**을 클릭합니다.
- 7 가상 시스템에서 **스왑 파일 위치**를 선택합니다.
- 8 **편집**을 클릭하고 사용할 로컬 데이터스토어를 선택한 다음 **확인**을 클릭합니다.
- 9 클러스터의 각 호스트에 대해 **단계 5~단계 8** 단계를 반복합니다.

결과

이제 DRS 클러스터에 대해 호스트-로컬 스왑 기능이 설정되었습니다.

독립 실행형 호스트에 대해 호스트-로컬 스왑 사용

호스트-로컬 스왑 기능을 사용하면 호스트에 로컬로 저장된 데이터스토어를 스왑 파일 위치로 지정할 수 있습니다. 독립 실행형 호스트에 대해 호스트-로컬 스왑을 사용하도록 설정할 수 있습니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 **구성**을 클릭합니다.
- 3 **가상 시스템**에서 **스왑 파일 위치**를 선택합니다.
- 4 **편집**을 클릭하고 **선택된 데이터스토어**를 선택합니다.

5 목록에서 로컬 데이터스토어를 선택하고 **확인**을 클릭합니다.

결과

이제 독립 실행형 호스트에 호스트-로컬 스왑 기능이 사용하도록 설정됩니다.

스왑 공간 및 메모리 오버 커밋

각 가상 시스템 스왑 파일에 대해 예약되지 않은 가상 시스템 메모리를 위한 스왑 공간(예약 크기와 구성된 메모리 크기의 차이)을 예약해야 합니다.

이 스왑 예약은 일부 경우에 ESXi 호스트가 가상 시스템 메모리를 유지할 수 있도록 하기 위해 필요합니다. 실제로는 호스트 수준 스왑 공간의 일부분만 사용될 수 있습니다.

ESXi를 사용하여 메모리를 오버 커밋하는 경우 벌루닝으로 인한 게스트 간 스왑을 지원하려면 게스트 운영 체제에도 충분한 스왑 공간이 있어야 합니다. 이 게스트 수준 스왑 공간은 가상 시스템의 구성된 메모리 크기와 예약 크기의 차이보다 크거나 같아야 합니다.

경고 메모리가 오버 커밋되어 있고 게스트 운영 체제에 스왑 공간이 충분히 구성되어 있지 않으면 가상 시스템의 게스트 운영 체제에 장애가 발생할 수 있습니다.

가상 시스템 장애를 방지하려면 가상 시스템의 스왑 공간 크기를 늘리십시오.

- Windows 게스트 운영 체제 - Windows 운영 체제에서는 스왑 공간을 페이징 파일이라고 합니다. 일부 Windows 운영 체제에서는 사용 가능한 디스크 공간이 충분한 경우 페이징 파일의 크기를 자동으로 늘립니다.

Microsoft Windows 설명서를 참조하거나 Windows 도움말 파일에서 "페이징 파일"을 검색하여 가상 메모리 페이징 파일의 크기를 변경하기 위한 지침을 따르십시오.

- Linux 게스트 운영 체제 - Linux 운영 체제에서는 스왑 공간을 스왑 파일이라고 합니다. 스왑 파일을 늘리는 방법은 다음의 Linux 매뉴얼 페이지를 참조하십시오.
 - `mkswap` - Linux 스왑 영역을 설정합니다.
 - `swapon` - 디바이스와 파일을 페이징 및 스왑할 수 있도록 합니다.

예를 들어 RAM이 8GB이고 가상 디스크가 2GB인 가상 시스템과 같이 메모리는 많고 가상 디스크는 적은 게스트 운영 체제는 스왑 공간이 충분하지 않을 가능성이 큽니다.

참고 스왑 파일을 씬 프로비저닝된 LUN에 저장하지 마십시오. 씬 프로비저닝된 LUN에 저장된 스왑 파일을 사용하는 가상 시스템을 실행하면 스왑 파일 증가 오류가 발생할 수 있으며 이로 인해 가상 시스템이 종료될 수 있습니다.

100GB 이상의 큰 스왑 파일을 만들면 가상 시스템의 전원을 켜는 데 소요되는 시간이 크게 늘어날 수 있습니다. 이를 방지하려면 큰 가상 시스템에는 예약 크기를 높게 설정하십시오.

호스트-로컬 스왑 파일을 사용하여 비용이 보다 적게 드는 스토리지에 스왑 파일을 저장할 수도 있습니다.

호스트에 대한 가상 시스템 스왑 파일 속성 구성

vSphere Client에서 호스트의 스왑 파일 위치를 구성하여 가상 시스템 스왑 파일의 기본 위치를 결정합니다.

기본적으로 가상 시스템의 스왑 파일은 다른 가상 시스템 파일이 들어 있는 폴더의 데이터스토어에 있습니다. 그러나 가상 시스템 스왑 파일을 다른 데이터스토어에 배치하도록 호스트를 구성할 수 있습니다.

이 옵션을 사용하면 가상 시스템 스왑 파일을 저가형 스토리지 또는 고성능 스토리지에 배치할 수 있습니다. 호스트 수준의 이 설정을 개별 가상 시스템에서 재정의할 수도 있습니다.

다른 스왑 파일 위치를 설정하면 vMotion을 사용한 마이그레이션이 보다 천천히 수행될 수도 있습니다. vMotion 성능을 최대화하려면 가상 시스템 스왑 파일과 동일한 디렉토리 대신 로컬 데이터스토어에 가상 시스템을 저장합니다. 가상 시스템이 로컬 데이터스토어에 저장되어 있으면 다른 가상 시스템 파일과 함께 스왑 파일을 저장해도 vMotion의 성능이 향상되지 않습니다.

사전 요구 사항

필요한 권한: **호스트 시스템.구성.스토리지 파티션 구성**

절차

1 vSphere Client에서 호스트를 찾습니다.

2 **구성**을 클릭합니다.

3 **가상 시스템**에서 **스왑 파일 위치**를 클릭합니다.

선택한 스왑 파일 위치가 표시됩니다. 구성된 스왑 파일 위치를 선택한 호스트에서 지원하지 않으면 해당 기능이 지원되지 않는다는 메시지가 탭에 표시됩니다.

호스트가 클러스터의 일부이고, 가상 시스템과 동일한 디렉토리에 스왑 파일을 저장하도록 클러스터가 설정되어 있으면 **구성** 아래의 호스트에서 스왑 파일 위치를 편집할 수 없습니다. 이러한 호스트에 대해 스왑 파일 위치를 변경하려면 클러스터 설정을 편집합니다.

4 **편집**을 클릭합니다.

5 스왑 파일을 저장할 위치를 선택합니다.

옵션	설명
가상 시스템 디렉토리	가상 시스템 구성 파일과 동일한 디렉토리에 스왑 파일을 저장합니다.
특정 데이터스토어 사용	지정된 위치에 스왑 파일을 저장합니다. 호스트가 지정하는 데이터스토어에 스왑 파일을 저장할 수 없으면 스왑 파일이 가상 시스템과 동일한 폴더에 저장됩니다.

6 (선택 사항) **특정 데이터스토어 사용**을 선택한 경우 목록에서 데이터스토어를 선택합니다.

7 **확인**을 클릭합니다.

결과

선택한 위치에 가상 시스템 스왑 파일이 저장됩니다.

클러스터의 가상 시스템 스왑 파일 위치 구성

기본적으로 가상 시스템의 스왑 파일은 다른 가상 시스템 파일이 들어 있는 폴더의 데이터스토어에 있습니다. 하지만, 선택한 다른 데이터스토어에 가상 시스템 스왑 파일을 배치하도록 클러스터의 호스트를 구성할 수 있습니다.

필요에 따라 저가형 스토리지 또는 고성능 스토리지에 가상 시스템 스왑 파일을 배치하도록 대체 스왑 파일 위치를 구성할 수 있습니다.

사전 요구 사항

클러스터의 가상 시스템 스왑 파일 위치를 구성하기 전에 [호스트에 대한 가상 시스템 스왑 파일 속성 구성의 설명](#)에 따라 클러스터의 호스트에 대한 가상 시스템 스왑 파일 위치를 구성해야 합니다.

절차

- 1 클러스터를 찾습니다.
- 2 **구성**을 클릭합니다.
- 3 **구성 > 일반**을 선택합니다.
- 4 스왑 파일 위치 옆의 **편집**을 클릭합니다.
- 5 스왑 파일을 저장할 위치를 선택합니다.

옵션	설명
가상 시스템 디렉토리	가상 시스템 구성 파일과 동일한 디렉토리에 스왑 파일을 저장합니다.
호스트에서 지정한 데이터스토어	호스트 구성에 지정된 위치에 스왑 파일을 저장합니다. 호스트가 지정하는 데이터스토어에 스왑 파일을 저장할 수 없으면 스왑 파일이 가상 시스템과 동일한 폴더에 저장됩니다.

- 6 **확인**을 클릭합니다.

스왑 파일 삭제

스왑 파일을 사용하는 가상 시스템이 실행 중인 호스트에서 장애가 발생하면 해당 스왑 파일은 계속 유지되어 기가 바이트에 이르는 디스크 공간을 차지할 수 있습니다. 스왑 파일을 삭제하면 이 문제를 해결할 수 있습니다.

절차

- 1 장애가 발생한 호스트에 있는 가상 시스템을 재시작합니다.
- 2 가상 시스템을 중지합니다.

결과

가상 시스템의 스왑 파일이 삭제됩니다.

NVM(비휘발성 메모리)이라고도 하는 PMem(영구 메모리)은 전원 공급이 중단된 후에도 데이터를 유지할 수 있습니다. PMem은 다운타임에 민감하고 고성능이 필요한 애플리케이션에 사용할 수 있습니다.

독립형 호스트 또는 클러스터에서 PMem을 사용하도록 VM을 구성할 수 있습니다. PMem은 로컬 데이터스토어로 취급됩니다. 영구 메모리를 사용하면 스토리지 지연 시간이 크게 감소합니다. ESXi에서 PMem을 사용하도록 구성된 VM을 생성할 수 있으며, 이 경우 해당 VM 내 애플리케이션이 이러한 속도 향상의 이점을 활용할 수 있습니다. VM의 전원을 처음 켜 후에는 VM을 켜든지 끄든지 관계없이 PMem이 예약되어 있습니다. 이 PMem은 VM이 마이그레이션되거나 제거될 때까지 예약 상태로 유지됩니다.

영구 메모리는 가상 시스템에서 두 가지 모드로 사용될 수 있습니다. 기존 게스트 운영 체제는 여전히 가상 영구 메모리 디스크 기능을 활용할 수 있습니다.

■ vPMem(가상 영구 메모리)

vPMem을 사용하는 경우 메모리가 가상 NVDIMM으로 게스트 운영 체제에 표시됩니다. 이를 통해 게스트 운영 체제가 바이트 주소 지정이 가능한 임의의 모드로 PMem을 사용할 수 있습니다.

참고 VM 하드웨어 버전 14 및 NVM 기술을 지원하는 게스트 운영 체제를 사용해야 합니다.

참고 PMem VM용 vSphere HA를 구성할 때는 VM 하드웨어 버전 19를 사용해야 합니다. 자세한 내용은 [PMem VM용 vSphere HA 구성](#)의 내용을 참조하십시오.

■ vPMemDisk(가상 영구 메모리 디스크)

vPMemDisk를 사용하는 경우 게스트 운영 체제에서 메모리를 가상 SCSI 디바이스로 액세스할 수 있지만 가상 디스크는 PMem 데이터스토어에 저장됩니다.

PMem을 사용하는 VM을 생성하는 경우 하드 디스크를 생성할 때 메모리가 예약됩니다. 또한 승인 제어도 하드 디스크를 생성할 때 수행됩니다. 자세한 내용은 [vSphere HA 승인 제어 PMem 예약](#)의 내용을 참조하십시오.

클러스터의 각 VM에는 약간의 PMem 용량이 할당됩니다. PMem의 총 용량은 클러스터에서 사용할 수 있는 총 용량보다 클 수 없습니다. 전원이 켜진 VM과 전원이 꺼진 VM에서 모두 PMem이 사용됩니다. PMem을 사용하도록 VM이 구성되어 있고 DRS를 사용하지 않는 경우 충분한 PMem 용량이 있는 호스트를 수동으로 선택하여 VM을 배치해야 합니다.

NVDIMM 및 기존 스토리지

NVDIMM은 메모리로 액세스됩니다. 기존 스토리지를 사용하는 경우 애플리케이션과 스토리지 디바이스 사이의 소프트웨어로 인해 처리 시간이 지연될 수 있습니다. PMem을 사용하는 경우 애플리케이션이 직접 스토리지를 사용합니다. 즉, PMem 성능이 기존 스토리지보다 더 우수합니다. 스토리지는 호스트에 로컬입니다. 그러나 시스템 소프트웨어는 변경 내용을 추적할 수 없으므로 백업과 같은 솔루션은 현재 PMem에서 작동하지 않습니다.

vPMem이 PMem이 아닌 데이터스토어에 대해 동시 쓰기가 아닌 모드에서 사용되는 경우 vSphere HA와 같은 솔루션의 범위가 제한됩니다. 페일오버를 사용하도록 설정된 vPMem VM에 대해 vSphere HA가 활성화되면 VM을 다른 호스트로 페일오버할 수 있습니다. 이 경우 VM은 새 호스트의 PMem 리소스를 사용합니다. 이전 호스트의 리소스를 확보하기 위해 가비지 수집기는 해당 리소스를 주기적으로 식별하고 확보하여 다른 VM에서 사용할 수 있도록 합니다.

네임스페이스

PMem의 네임스페이스는 ESXi가 시작되기 전에 구성됩니다. 네임스페이스는 시스템의 디스크와 비슷합니다. ESXi는 네임스페이스를 읽고 GPT 헤더를 작성하여 여러 네임스페이스를 하나의 논리적 볼륨으로 결합합니다. 네임스페이스는 이전에 구성되지 않은 경우 기본적으로 자동으로 포맷됩니다. 이미 포맷된 경우 ESXi가 PMem을 마운트하려고 시도합니다.

GPT 테이블

PMem 스토리지의 데이터가 손상될 경우 ESXi에 장애가 발생할 수 있습니다. 이 문제를 방지하기 위해 ESXi는 PMem 마운트 시 메타데이터에서 오류를 확인합니다.

PMem 영역

PMem 영역은 단일 vNVDimm 또는 vPMemDisk를 나타내는 연속 바이트 스트림입니다. 각 PMem 볼륨은 단일 호스트에 속합니다. 관리자가 많은 수의 호스트가 포함된 클러스터의 각 호스트를 관리해야 하는 경우 이는 관리하기 어려울 수 있습니다. 그러나 관리자가 각 데이터스토어를 개별적으로 관리할 필요가 없습니다. 대신 클러스터에 있는 전체 PMem 용량을 하나의 데이터스토어로 간주할 수 있습니다.

VC 및 DRS는 PMem 데이터스토어의 초기 배치를 자동화합니다. VM이 생성되거나 디바이스가 VM에 추가될 때 로컬 PMem 스토리지 프로파일을 선택합니다. 구성의 나머지 부분은 자동으로 수행됩니다. 한 가지 제한 사항으로, ESXi는 PMem 데이터스토어에 VM 홈을 배치하도록 허용하지 않습니다. VM 홈은 VM 로그 및 통계 파일을 저장하기 위해 귀중한 공간을 사용하기 때문입니다. 이러한 영역은 VM 데이터를 나타내는 데 사용되며 바이트 주소 지정 가능한 nvDimm 또는 VpMem 디스크로 표시될 수 있습니다.

마이그레이션

PMem은 로컬 데이터스토어이므로 VM을 이동하려는 경우 Storage vMotion을 사용해야 합니다. vPMem을 사용하는 VM은 PMem 리소스가 있는 ESX 호스트에만 마이그레이션할 수 있습니다. vPMemDisk를 사용하는 VM은 PMem 리소스가 없는 ESX 호스트에 마이그레이션할 수 있습니다.

오류 처리 및 NVDimm 관리

호스트 장애로 인해 동시 쓰기가 아닌 모드의 vPMem VM에서 가용성이 손실될 수 있습니다. 심각한 오류가 발생할 경우 모든 데이터가 손실될 수 있으며 수동 단계를 수행하여 PMem을 다시 포맷해야 합니다.

vSphere Client에서 vSphere 영구 메모리 사용

영구 메모리에 대한 간략한 개념을 보려면 다음 비디오를 시청하십시오.



(vSphere Client에서 vSphere 영구 메모리 사용)

vSphere Client에서 PMEM을 사용한 작업의 향상된 기능

PMem으로 작업 시 HTML5 기반 vSphere Client의 향상된 기능에 대한 간략한 개요는 다음을 참조하십시오.



(vSphere Client에서 PMEM을 사용한 작업의 향상된 기능)

vSphere Client에서 PMEM을 사용하여 VM 마이그레이션 및 복제

PMem을 사용하는 가상 시스템 마이그레이션 및 복제에 대한 간략한 개요는 다음을 참조하십시오.



(vSphere Client에서 PMEM을 사용하여 VM 마이그레이션 및 복제)

다음으로 아래 항목을 읽으십시오.

- [PMem VM용 vSphere HA 구성](#)
- [vSphere HA 승인 제어 PMem 예약](#)
- [vSphere 메모리 모니터링 및 업데이트 적용](#)

PMem VM용 vSphere HA 구성

동시 쓰기 모드에서 PMem VM용 vSphere HA를 구성하면 호스트에 장애가 발생하는 경우 작동되는 다른 호스트에서 VM이 복원될 수 있습니다.

사전 요구 사항

- 하드웨어 버전 19를 선택해야 합니다.
- vPMemDisk가 있는 PMem VM은 지원되지 않습니다.

절차

1 새 가상 시스템 마법사에서 새 VM을 생성할 때 **하드웨어 사용자 지정**을 선택합니다.

- a 새 디바이스 추가를 클릭하고 드롭다운 메뉴에서 **NVDIMM 추가**를 선택합니다.
- b 모든 NVDIMM 디바이스에 대해 다른 호스트에서 **페일오버 허용** 확인란을 선택합니다.
- c 다음을 클릭하고 새 가상 시스템 마법사를 완료합니다.

호스트 장애 시 NVDIMM PMem 데이터를 복구할 수 없습니다. 기본적으로 HA는 다른 호스트에서 이 가상 시스템을 다시 시작하려고 시도하지 않습니다. 호스트 장애 시 HA가 가상 시스템을 페일오버하도록 허용하면 비어있는 새 NVDIMM이 있는 다른 호스트에서 가상 시스템이 다시 시작됩니다.

2 기존 VM에서 HA를 사용하도록 설정하려면 해당 VM을 찾습니다.

- a VM 하드웨어에서 **편집**을 클릭합니다.
- b NVDIMM을 선택합니다.
- c 모든 NVDIMM 디바이스에 대해 다른 호스트에서 **페일오버 허용** 확인란을 선택합니다.
- d **확인**을 클릭합니다.

호스트 장애 시 HA는 비어있는 새 NVDIMM이 있는 다른 호스트에서 가상 시스템을 다시 시작합니다.

vSphere HA 승인 제어 PMem 예약

승인 제어는 클러스터 내의 페일오버 용량을 확보하기 위해 vSphere HA에서 사용하는 정책입니다.

허용할 수 있는 잠재적인 호스트 장애 수를 늘리면 예약된 용량 및 가용성 제한이 증가합니다. 호스트 페일오버 용량을 위해 일정 백분율의 영구 메모리를 예약할 수 있습니다. 이것은 차단된 실제 스토리지 용량이며, 호스트 전원 끄기 시 고려해야 합니다.

클러스터 설정 편집에서 **승인 제어**를 선택하여 호스트가 허용할 장애 수를 지정할 수 있습니다.

CPU/메모리 예약을 정의한 기준 및 관련 정보:

- **클러스터 리소스 백분율**, 클러스터의 가상 시스템이 현재 영구 메모리를 사용하지 않는 경우에도 클러스터의 일부 영구 메모리 용량이 페일오버 전용으로 사용됩니다. 이 백분율은 재정의될 수 없으며, **허용할 호스트 장애** 설정에 따라 자동으로 계산됩니다. PMem 승인 제어를 사용하도록 설정하면 PMem을 디스크로 사용하는 VM이 있더라도 클러스터 전체에서 PMem 용량이 예약됩니다.
- **슬롯 정책(전원이 켜진 VM)**, 영구 메모리 승인 제어는 영구 메모리 리소스에 대해서만 클러스터 리소스 백분율로 슬롯 정책을 재정의합니다. 백분율 값은 클러스터에서 허용하는 호스트 장애 설정에서 자동으로 계산되며 재정의할 수 없습니다.
- **전용 페일오버 호스트**, 전용 페일오버 호스트의 영구 메모리는 페일오버 전용이며 이러한 호스트에는 영구 메모리를 사용하여 가상 시스템을 프로비저닝할 수 없습니다.

참고 승인 제어 정책을 선택한 후에는 **영구 메모리 페일오버 용량 예약** 확인란도 클릭하여 PMem 승인 제어를 사용하도록 설정해야 합니다.

vSphere 메모리 모니터링 및 업데이트 적용

vMMR은 메모리 모드로 인해 애플리케이션 워크로드가 재발하는지 확인할 수 있도록 데이터를 수집하고 성능 통계에 대한 가시성을 제공합니다.

Intel Optane Persistent Memory는 애플리케이션 직접 모드 또는 메모리 모드의 BIOS 설정에서 구성할 수 있습니다. 애플리케이션 직접 모드에서 영구 메모리는 DRAM과 함께 바이트 주소 지정이 가능한 영구 메모리로 액세스할 수 있습니다. 메모리 모드에서 DRAM은 하드웨어 캐시가 되며 더 큰 PMem은 휘발성이 되어 시스템 메모리로 나타납니다.

메모리 모드는 보이지 않으며 VM에 투명합니다. 메모리 모드에서 시스템을 구성하면 시스템이 DRAM을 사용하는 기존 시스템으로 나타납니다. 클러스터에는 구성이 다른 호스트가 혼합되어 있을 수 있습니다. vSphere는 메모리 모드에 있는 시스템에 대한 추가 정보를 표시합니다. ESXi는 호스트 수준 및 VM 수준 통계에 대한 정보를 수집하는 성능 카운터를 프로그래밍합니다. 이러한 성능 통계는 경보를 생성하는 데 사용됩니다. 통계도 성능 차트에서 추적됩니다.

호스트 요약 탭, **메모리 계층화: 몇 가지 추가 세부 정보가 있는 하드웨어**에서 시스템이 메모리 모드에 있는지 확인할 수 있습니다.

The screenshot shows the vSphere Summary tab for a host. The 'Memory Tiering' is set to 'Hardware'. A tooltip is displayed over the 'Hardware' text, stating: 'Intel Optane™ Persistent Memory configured in Memory Mode.' The tooltip has a close button (X) in the top right corner.

Summary	Monitor	Configure	Permissions	VMs	Datastores	Networks	Updates
Logical Processors:	96						
NICs:	3						
Virtual Machines:	1						
Memory Tiering:	Hardware						
State:	Connected						
Uptime:	6 hours						

구성 > 하드웨어 > 개요 > 메모리에서 DRAM 및 PMEM의 크기를 볼 수도 있습니다.


The screenshot shows the vSphere Configure > Hardware > Memory configuration page. The 'Memory Tiering' is set to 'Hardware'. The table below shows the memory configuration details.

Memory	
Total	503.68 GB
System	385.17 MB
Virtual machines	503.3 GB
Memory Tiering	Hardware ⓘ
Tier 0	256 GB DRAM (Cache)
Tier 1	503.67 GB PMem (Memory)

ESXi는 두 가지 종류의 메모리 통계를 수집하고 노출합니다.

- **호스트 수준 통계:** 메모리 하위 구성 요소는 성능 카운터를 프로그래밍하여 DRAM 및 PMem 성능을 측정합니다. 호스트 수준 통계는 서로 다른 메모리 유형(DRAM, PMem)에 대한 총계, 읽기/쓰기 대역폭, 읽기/쓰기 지연 시간 및 누락 비율입니다.
- **VM 수준 통계:** vSphere는 성능 카운터를 모니터링하여 VM의 DRAM 및 PMEM 읽기 대역폭에 대한 데이터를 가져옵니다.

호스트와 VM 모두 성능 차트 아래에 새 메모리 창이 있습니다. 여기에는 메모리 활용률 및 메모리 회수와 같은 메모리 세부 정보와 새로운 통계가 표시됩니다. ESXi 호스트 수준에서 메모리 대역폭 및 메모리 누락 비율 차트를 모니터링할 수 있습니다. VM 수준에서 PMem 읽기 대역폭 및 DRAM 읽기 대역폭을 볼 수 있습니다.

ESXi 호스트의 **VM** 탭에서 호스트에 상주하는 모든 가상 시스템에 대한 성능 정보가 포함된 목록을 볼 수 있습니다. 메모리 모드가 가상 시스템에 미치는 영향에 대한 정보를 표시하려면 열 보기() 아이콘을 클릭하고 활성 메모리, DRAM 읽기 대역폭 및 PMem 읽기 대역폭 메트릭을 선택합니다.

미리 구성된 기본 경보가 2개 있는데 하나는 호스트 수준(호스트 메모리 모드 높은 활성 DRAM 사용량)이고 다른 하나는 VM 수준(가상 시스템 높은 PMem 대역폭 사용량)입니다. 경보 조건이 충족되면 해당 경보를 트리거하는 이벤트가 게시됩니다. 성능 메트릭을 기반으로 사용자 지정 경보를 생성할 수도 있습니다. vMMR 경보는 메모리 모드로 구성된 호스트에서만 작동합니다.

클러스터에서 DRS가 사용되도록 설정되고 완전히 자동화된 경우 호스트의 활성 메모리 활용률이 DRAM 캐시 크기의 특정 비율을 초과하는 경우 DRS는 로드 밸런싱을 위해 일부 VM을 호스트 외부로 이동할 수 있습니다.

자세한 내용은 "vSphere 모니터링 및 성능" 을 참조하십시오.

참고 vMMR은 Intel Broadwell, Skylake, Cascade Lake 및 Ice Lake 플랫폼에서 지원됩니다. 호스트 수준 DRAM 통계는 이러한 플랫폼에서 사용할 수 있습니다. 호스트 및 VM 수준 PMem 통계는 메모리 모드로 구성된 Cascade Lake 및 Ice Lake 호스트에서만 사용할 수 있습니다.

vSphere에서 가상 그래픽 구성

12

지원되는 그래픽 구현의 경우 그래픽 설정을 편집할 수 있습니다.

vSphere는 여러 가지 그래픽 구현을 지원합니다.

- VMware는 AMD, Intel 및 NVIDIA에서 3D 그래픽 솔루션을 지원합니다.
- NVIDIA GRID가 지원됩니다.
- 단일 NVIDIA VIB가 vSGA 및 vGPU 구현을 모두 지원할 수 있습니다.
- Intel 및 NVIDIA에 대한 vCenter GPU 성능 차트를 제공합니다.
- Horizon View VDI 데스크톱에 대한 그래픽을 사용하도록 설정합니다.
- 최대 16개의 vGPU 디바이스를 사용하여 VM을 생성할 수 있습니다. ESXi 및 vCenter에서 HWv21을 사용하도록 설정해야 합니다.

호스트 그래픽 설정을 구성하고 VM별로 vGPU 그래픽 설정을 사용자 지정할 수 있습니다.

참고 이 장에서 "메모리"는 물리적 RAM을 나타냅니다.

다음으로 아래 항목을 읽으십시오.

- GPU 통계 보기
- 가상 시스템에 NVIDIA GRID vGPU 추가
- 호스트 그래픽 구성
- 그래픽 디바이스 구성
- vGPU가 있는 VM 마이그레이션
- vGPU 크기 구성

GPU 통계 보기

호스트 그래픽 카드의 세부 정보를 볼 수 있습니다.

GPU 온도, 활용률 및 메모리 사용량을 확인할 수 있습니다.

참고 GPU 드라이버가 호스트에 설치된 경우에만 이러한 통계가 표시됩니다.

절차

- 1 vSphere Client에서 호스트로 이동합니다.
- 2 **모니터** 탭을 클릭하고 **성능**을 클릭합니다.
- 3 **고급**을 클릭하고 드롭다운 메뉴에서 **GPU**를 선택합니다.

가상 시스템에 NVIDIA GRID vGPU 추가

ESXi 호스트에 NVIDIA GRID GPU 그래픽 디바이스가 있는 경우 NVIDIA GRID 가상 GPU(vGPU) 기술을 사용하여 가상 시스템을 구성할 수 있습니다.

NVIDIA GRID GPU 그래픽 디바이스는 CPU에 과부하를 주지 않은 상태로 복잡한 그래픽 작업을 고성능으로 실행 및 최적화하기 위해 설계되었습니다.

사전 요구 사항

- 적절한 드라이버가 포함된 NVIDIA GRID GPU 그래픽 디바이스가 호스트에 설치되어 있는지 확인합니다. "vSphere 업그레이드" 설명서를 참조하십시오.
- 가상 시스템이 ESXi 6.0 이상과 호환되는지 확인합니다.

절차

- 1 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
- 2 **가상 하드웨어** 탭에서 **새 디바이스 추가**를 선택하고 드롭다운 메뉴에서 **새 PCI 디바이스**를 선택합니다.
- 3 **새 PCI 디바이스**를 확장하고 가상 시스템을 연결할 NVIDIA GRID vGPU 패스스루 디바이스를 선택합니다.

참고 전체 메모리 예약이 자동으로 적용되며 PCI 디바이스에 필요합니다.

- 4 GPU 프로파일을 선택합니다.
GPU 프로파일은 vGPU 유형을 나타냅니다.
- 5 **확인**을 클릭합니다.

결과

가상 시스템이 디바이스에 액세스할 수 있습니다.

호스트 그래픽 구성

호스트 단위로 그래픽 옵션을 사용자 지정할 수 있습니다.

사전 요구 사항

가상 시스템의 전원을 꺼야 합니다.

절차

- 1 호스트를 선택하고 **구성 > 하드웨어 > 그래픽**을 선택합니다.
- 2 **호스트 그래픽**에서 **편집**을 선택합니다.
- 3 **호스트 그래픽 설정 편집** 창에서 다음을 선택합니다.

옵션	설명
공유됨	VMware 공유 가상 그래픽
Shared Direct	벤더 공유 패스스루 그래픽

- 4 공유한 패스스루 GPU 할당 정책을 선택합니다.
 - a GPU 전체에 VM 분산(최고 성능)
 - b 최대 수준까지 GPU에서 VM 그룹화(GPU 통합)
- 5 **확인**을 클릭합니다.

다음에 수행할 작업

확인을 클릭한 후 호스트에서 Xorg를 다시 시작해야 합니다.

그래픽 디바이스 구성

비디오 카드에 대한 그래픽 유형을 편집할 수 있습니다.

사전 요구 사항

가상 시스템의 전원을 꺼야 합니다.

절차

- 1 **그래픽 디바이스**에서 그래픽 카드를 선택하고 **편집**을 클릭합니다.
 - a VMware 공유 가상 그래픽에 대해 **공유됨**을 선택합니다.
 - b 벤더 공유 패스스루 그래픽에 대해 **Shared Direct**를 선택합니다.
- 2 **확인**을 클릭합니다.

결과

디바이스를 선택하면 해당 디바이스를 사용하고 있는 가상 시스템(활성화된 경우)이 표시됩니다.

다음에 수행할 작업

확인을 클릭한 후 호스트에서 Xorg를 다시 시작해야 합니다.

vGPU가 있는 VM 마이그레이션

vGPU가 있는 VM을 마이그레이션할 수 있습니다.

vSphere 8.0 U2부터 DRS vGPU를 사용하여 VM을 자동으로 마이그레이션할 수 있습니다. DRS 클러스터 고급 옵션이 설정되고 VM에 대한 예상 VM 디바이스의 스텐 시간이 VM 디바이스의 vMotion 스텐 시간 제한보다 낮으면 DRS는 VM 마이그레이션을 자동화합니다.

이 기능을 사용하도록 설정하려면 인프라가 다음 요구 사항을 충족하는지 확인합니다.

- 정상인 vSphere 수명 주기 서비스([KB 91891](#) 참조)
- vCenter 통해서만 VM의 vGPU 디바이스 구성
- 정상인 vMotion 네트워크([vMotion NIC 설정](#))

이러한 요구 사항이 충족되면 다음과 같은 DRS 클러스터 고급 옵션을 추가합니다.

DRS 클러스터 고급 옵션	값
PassthroughDrnsAutomation	1
LBMaxVmotionPerHost	1

스텐 시간이 "vMotion 스텐 시간 제한"(기본값 100초)을 초과하는 vGPU VM의 경우 VI 관리자는 다음과 같은 DRS 클러스터 고급 옵션을 추가할 수 있습니다.

DRS 클러스터 고급 옵션	값
VmDevicesStunTimeTolerated	<클러스터에 있는 모든 VM의 예상 스텐 시간보다 긴 시간(초 단위)>(기본값 100초)

또는 **VM 구성 > > VM 옵션 탭 > 고급 섹션**에서 "vMotion 스텐 시간 제한"을 수정할 수 있습니다.

이전 릴리스의 경우:

유지 보수 모드 제거의 경우 [KB 88271 DRS 클러스터에서 호스트 유지 관리 모드에 대한 vGPU 가상 시스템 자동화된 마이그레이션](#)을 참조하십시오.

VM 배치 문제가 발생하는 경우 DRS 자동화를 **부분적으로 자동화됨**으로 줄이십시오. 자세한 내용은 [클러스터 설정 편집](#)을 참조하십시오.

vGPU가 있는 VM을 다른 호스트로 수동으로 마이그레이션할 수 있습니다. 자세한 내용은 vCenter Server 및 호스트 관리 가이드에서 vMotion을 사용하여 vGPU 가상 시스템 마이그레이션 을 참조하십시오.

vGPU 크기 구성

vCenter에서 vGPU 크기를 조정할 수 있습니다.

하나의 물리적 GPU의 vGPU 프로파일에는 서로 다른 유형 또는 크기가 있을 수 있습니다. 예를 들어 단일 디바이스에서 16GB 계산 및 4GB 그래픽을 갖춘 vGPU VM 프로파일을 동시에 실행할 수 있습니다. 이를 통해 vGPU 워크로드의 유연성이 향상되고 vGPU 워크로드가 혼합된 디바이스를 더 잘 활용할 수 있습니다.

디바이스별로 이기종 크기를 활성화하거나 비활성화할 수 있습니다. 이 기능을 활성화하면 일부 프로파일의 최대 인스턴스 수가 줄어듭니다. 예를 들어 A16-1Q 1Q 프로파일은 기능이 비활성화되면 24개의 인스턴스를 지원하지만 활성화되면 16개의 인스턴스를 지원합니다. 워크로드에 대한 용량을 최대화하기 위해 일부 디바이스에서 기능을 비활성화하는 것이 좋습니다.

절차

- 1 ESXi 호스트를 찾습니다.
- 2 **구성 > 하드웨어 > 그래픽 > 그래픽 디바이스**에서 vGPU 크기를 조정할 수 있습니다.
- 3 **vGPU 모드**를 선택하고 **SameSize** 또는 **MixedSize**를 선택합니다. 기본값은 **SameSize**입니다.
 - **SameSize**는 크기가 동종임을 나타냅니다.
 - **MixedSize**는 크기가 이기종일 수 있음을 나타냅니다.

결과

그래픽 디바이스를 편집할 때 다음 메시지가 표시됩니다.

설정은 호스트 또는 xorg 서비스를 다시 시작한 후에 적용됩니다.

이 작업은 vGPU 크기를 편집할 때 필요합니다.

vSphere를 사용하여 스토리지 I/O 리소스 관리

13

vSphere Storage I/O Control을 사용하면 클러스터 전체의 스토리지 I/O 우선 순위를 지정하여 워크로드를 효율적으로 통합하고 과도한 프로비저닝에 따른 추가 비용을 줄일 수 있습니다.

Storage I/O Control은 공유 및 제한 구조를 확장하여 스토리지 I/O 리소스를 처리합니다. I/O 정체 기간 동안 가상 시스템에 할당된 스토리지 I/O 양을 제어할 수 있으므로 I/O 리소스 할당 시 중요도가 높은 가상 시스템이 중요도가 낮은 가상 시스템보다 우선하도록 할 수 있습니다.

데이터스토어에서 Storage I/O Control을 사용하도록 설정하면 ESXi는 호스트에서 해당 데이터스토어와 통신할 때 관측되는 디바이스 지연 시간을 모니터링하기 시작합니다. 디바이스 지연 시간이 임계값을 초과하면 데이터스토어가 정체되는 것으로 간주되어 해당 데이터스토어에 액세스하는 각 가상 시스템이 해당 공유에 비례하여 I/O 리소스를 할당받게 됩니다. 가상 시스템당 공유 수를 설정합니다. 필요에 맞게 각 가상 시스템의 공유 수를 조정할 수 있습니다.

I/O 필터 프레임워크(VAIO)를 사용하면 VMware 및 해당 파트너가 각 VMDK의 I/O를 가로채는 필터를 개발할 수 있고, VAIO는 VMDK 세분성에 대해 필요한 기능을 제공합니다. VAIO가 SPBM(스토리지 정책 기반 관리)과 함께 연동하므로 사용자는 VMDK에 연결된 스토리지 정책을 통해 필터 기본 설정을 설정할 수 있습니다.

Storage I/O Control 구성은 두 단계 프로세스로 이루어집니다.

- 1 데이터스토어에 대해 Storage I/O Control을 사용하도록 설정합니다.
- 2 각 가상 시스템에 허용되는 스토리지 I/O 공유 수와 IOPS(초당 I/O 작업 수)에 대한 상한을 설정합니다.

기본적으로 모든 가상 시스템 공유는 무제한 IOPS와 함께 보통(1000)으로 설정됩니다.

참고 Storage DRS가 지원되는 데이터스토어 클러스터에서는 Storage I/O Control이 기본적으로 사용하도록 설정됩니다.

참고 vSphere 8.0 U1은 VM 재구성 중에 LWD I/O 필터의 원활한 무중단 추가 및 무중단 제거를 지원합니다.

참고 이 장에서 "메모리"는 물리적 RAM을 나타냅니다.

다음으로 아래 항목을 읽으십시오.

- 가상 시스템 스토리지 정책
- I/O 필터
- Storage I/O Control 요구 사항

- Storage I/O Control 리소스 공유 및 제한
- Storage I/O Control 공유 및 제한 보기
- Storage I/O Control 공유 모니터링
- Storage I/O Control 리소스 공유 및 제한 설정
- Storage I/O Control 사용
- Storage I/O Control 임계값 설정
- 스토리지 프로파일과 Storage DRS 통합

가상 시스템 스토리지 정책

가상 시스템 스토리지 정책은 가상 시스템 프로비저닝에 필수적입니다. 이 정책을 통해 가상 시스템에 제공되는 스토리지 유형, 가상 시스템이 스토리지에 배치되는 방법 및 가상 시스템에 제공되는 데이터 서비스를 제어할 수 있습니다.

vSphere에 기본 스토리지 정책이 포함됩니다. 그러나 사용자가 새 정책을 정의하고 할당할 수 있습니다.

스토리지 정책은 VM 스토리지 정책 인터페이스를 사용하여 생성합니다. 정책 정의 시 가상 시스템에서 실행할 애플리케이션에 대한 다양한 스토리지 요구 사항을 지정합니다. 또한 스토리지 정책을 사용하여 가상 디스크에 대해 캐싱 또는 복제 같은 특정 데이터 서비스를 요청할 수도 있습니다.

가상 시스템을 생성, 복제 또는 마이그레이션할 때 스토리지 정책을 적용합니다. 스토리지 정책을 적용하면 SPBM(스토리지 정책 기반 관리) 메커니즘이 가상 시스템을 일치하는 데이터스토어에 배치하고, 특정 스토리지 환경에서는 필요한 서비스 수준을 보장하기 위해 스토리지 리소스 내에 가상 시스템 스토리지 개체를 프로비저닝 및 할당하는 방법을 결정합니다. 또한 SPBM은 가상 시스템에 대해 요청된 데이터 서비스를 사용하도록 설정합니다. vCenter Server는 정책 규정 준수 여부를 모니터링하고 가상 시스템이 할당된 스토리지 정책을 위반하면 경고를 보냅니다.

자세한 내용은 "vSphere Storage" 를 참조하십시오.

I/O 필터

가상 디스크에 연결된 I/O 필터는 기본 스토리지 토폴로지와는 관계없이 가상 시스템 I/O 경로에 직접 액세스할 수 있습니다.

VMware는 특정 범주의 I/O 필터를 제공합니다. 또한 타사 벤더가 I/O 필터를 생성할 수 있습니다. 일반적으로 I/O 필터는 패키지 형식으로 배포되며, 패키지에는 필터 구성 요소를 vCenter Server 및 ESXi 호스트 클러스터에 배포하는 설치 관리자가 제공됩니다.

I/O 필터가 ESXi 클러스터에 배포된 후에는 vCenter Server가 클러스터 내의 각 호스트에 대해 I/O 필터 스토리지 제공자(또는 VASA 제공자라고도 함)를 자동으로 구성하고 등록합니다. 스토리지 제공자는 vCenter Server와 통신하고, I/O 필터가 제공하는 데이터 서비스가 VM 스토리지 정책 인터페이스에 표시되도록 합니다. 이러한 데이터 서비스는 VM 정책에 대한 공통 규칙을 정의할 때 참조할 수 있습니다. 이 정책에 가상 디스크를 연결한 후에는 가상 디스크에서 I/O 필터가 사용 가능하도록 설정됩니다.

자세한 내용은 "vSphere Storage" 를 참조하십시오.

Storage I/O Control 요구 사항

Storage I/O Control에는 몇 가지 요구 사항 및 제한이 있습니다.

- Storage I/O Control을 사용하도록 설정된 데이터스토어는 단일 vCenter Server 시스템으로 관리되어야 합니다.
- Storage I/O Control은 Fibre Channel, iSCSI 및 NFS 연결 스토리지에서 지원됩니다. RDM(원시 디바이스 매핑)은 지원되지 않습니다.
- Storage I/O Control은 다중 익스텐트가 포함된 데이터스토어를 지원하지 않습니다.
- 자동화된 스토리지 계층화 기능이 있는 어레이로 백업되는 데이터스토어에서 Storage I/O Control을 사용하려면 먼저 "VMware 스토리지/SAN 호환성 가이드" 를 검토하여 자동 계층화 기능이 있는 스토리지 어레이가 Storage I/O Control과의 호환성 검증이 이루어졌는지 확인하십시오.

자동화된 스토리지 계층화는 사용자가 설정한 정책과 현재 I/O 패턴을 기준으로 LUN/볼륨이나 일부 LUN/볼륨을 다른 유형의 스토리지 미디어(SSD, FC, SAS, SATA)로 마이그레이션하는 어레이(또는 어레이 그룹)의 기능입니다. 서로 다른 유형의 스토리지 미디어 간에 수동으로 데이터를 마이그레이션하는 기능이 있는 어레이를 비롯하여, 이러한 자동 마이그레이션/계층화 기능이 없는 어레이에는 별도의 특수한 인증이 필요하지 않습니다.

Storage I/O Control 리소스 공유 및 제한

각 가상 시스템에 허용되는 스토리지 I/O 공유 개수와 IOPS(초당 I/O 작업 수)에 대한 상한을 할당합니다. 데이터스토어에서 스토리지 I/O 정체가 감지되면 해당 데이터스토어에 액세스하는 가상 시스템의 I/O 워크로드가 각 가상 시스템의 가상 시스템 공유에 비례하여 조정됩니다.

스토리지 I/O 공유는 메모리 및 CPU 리소스 할당(리소스 할당 공유 참조)에 사용되는 공유와 비슷합니다. 이러한 공유는 스토리지 I/O 리소스 배포와 관련하여 가상 시스템의 상대적 중요성을 나타냅니다. 리소스 경합 시 가상 시스템의 공유 값이 높을수록 스토리지 어레이에 대해 더 많은 액세스 권한을 가집니다. 스토리지 I/O 리소스를 할당할 때 가상 시스템에 대해 허용되는 IOPS를 제한할 수 있습니다. 기본적으로 IOPS에는 제한이 없습니다.

리소스 제한을 설정할 때의 장점과 단점은 리소스 할당 제한에 설명되어 있습니다. 가상 시스템에 대해 설정하려는 제한이 IOPS 대신 초당 MB를 기준으로 하는 경우에는 초당 MB를 해당 가상 시스템의 일반적인 I/O 크기를 기반으로 하는 IOPS로 변환할 수 있습니다. 예를 들어 IO 크기가 64KB인 백업 애플리케이션을 초당 10MB로 제한하려면 제한을 160 IOPS로 설정합니다.

Storage I/O Control 공유 및 제한 보기

데이터스토어에서 실행 중인 모든 가상 시스템에 대한 공유 및 제한을 볼 수 있습니다. 이 정보를 확인하면 가상 시스템이 실행 중인 클러스터에 관계없이 데이터스토어에 액세스하고 있는 모든 가상 시스템의 설정을 비교할 수 있습니다.

절차

- 1 vSphere Client에서 데이터스토어를 찾습니다.
- 2 **VM** 탭을 클릭합니다.

이 탭에는 데이터스토어에서 실행 중인 각 가상 시스템과 관련 공유 값 및 데이터스토어 공유의 백분율이 표시됩니다.

Storage I/O Control 공유 모니터링

Storage I/O Control이 해당 공유를 기반으로 데이터스토어에 액세스하는 가상 시스템의 I/O 워크로드를 처리하는 방법을 모니터링하려면 데이터스토어 **성능** 탭을 사용합니다.

데이터스토어 성능 차트를 통해 다음 정보를 모니터링할 수 있습니다.

- 데이터스토어에서의 평균 지연 시간 및 집계 IOPS
- 호스트 간의 지연 시간
- 호스트 간의 대기열 크기
- 호스트 간의 읽기/쓰기 IOPS
- 가상 시스템 디스크 간의 읽기/쓰기 지연 시간
- 가상 시스템 디스크 간에 IOPS 읽기/쓰기

절차

- 1 vSphere Client에서 데이터스토어를 찾습니다.
- 2 **모니터** 탭에서 **성능**을 클릭합니다.
- 3 **고급**을 선택합니다.

Storage I/O Control 리소스 공유 및 제한 설정

가상 시스템에 상대적 공유 양을 할당하여 중요도에 따라 가상 시스템에 스토리지 I/O 리소스를 할당합니다.

가상 시스템 워크로드가 그다지 비슷하지 않은 경우 공유에서 I/O 작업과 관련한 할당이나 초당 메가바이트를 지정하지 않을 수도 있습니다. 공유 양이 많은 가상 시스템에서는 공유 양이 적은 가상 시스템에 비해 더 많은 동시 I/O 작업을 스토리지 디바이스 또는 데이터스토어에서 진행할 수 있습니다. 두 가상 시스템의 처리량은 워크로드에 따라 달라질 수 있습니다.

사전 요구 사항

VM 스토리지 정책을 생성하고 VM 스토리지 정책의 공통 규칙을 정의하는 방법에 대한 자세한 내용은 "vSphere Storage" 를 참조하십시오.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
 - a 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - b **VM** 탭을 클릭합니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭한 후 **설정 편집**을 클릭합니다.
- 3 **가상 하드웨어** 탭을 클릭하고 목록에서 가상 하드 디스크를 선택합니다. **하드 디스크**를 확장합니다.
- 4 드롭다운 메뉴에서 **VM 스토리지 정책**을 선택합니다.

스토리지 정책을 선택한 경우 **공유 및 제한 - IOPS**를 수동으로 구성하지 마십시오.
- 5 **공유**에서 드롭다운 메뉴를 클릭하고 가상 시스템에 할당할 상대적 공유 양(낮음, 보통 또는 높음)을 선택합니다.

사용자 지정을 선택하여 사용자 정의 공유 값을 입력할 수 있습니다.
- 6 **제한 - IOPS**에서 드롭다운 메뉴를 클릭하고 가상 시스템에 할당할 스토리지 리소스 상한을 입력합니다.

IOPS는 초당 I/O 작업 수입니다. 기본적으로 IOPS에는 제한이 없습니다. 낮음(500), 보통(1000) 또는 높음(2000)을 선택하거나, 사용자 지정을 선택하여 사용자 정의 공유 수를 입력할 수 있습니다.
- 7 **확인**을 클릭합니다.

Storage I/O Control 사용

Storage I/O Control을 사용하도록 설정하면 ESXi에서는 데이터스토어 지연 시간을 모니터링하고 데이터스토어 평균 지연 시간이 임계값을 초과할 경우 I/O 로드를 조절합니다.

절차

- 1 vSphere Client에서 데이터스토어를 찾습니다.
- 2 **구성** 탭을 클릭합니다.
- 3 **설정**을 클릭하고 **일반**을 클릭합니다.
- 4 **데이터스토어 기능의 편집**을 클릭합니다.
- 5 **Storage I/O Control 사용** 확인란을 선택합니다.
- 6 **확인**을 클릭합니다.

결과

데이터스토어 기능에서 데이터스토어에 대해 Storage I/O Control을 사용하도록 설정됩니다.

Storage I/O Control 임계값 설정

데이터스토어에 대한 정체 임계값은 Storage I/O Control에서 공유에 따라 가상 시스템 워크로드에 중요도를 할당하기 전에 데이터스토어에 허용되는 지연 시간 상한 값입니다.

대부분의 환경에서는 임계값 설정을 조정할 필요가 없습니다.

경고 두 개의 서로 다른 데이터스토어에서 동일한 스펜들을 공유하는 경우 Storage I/O Control이 제대로 기능하지 않을 수 있습니다.

정체 임계값 설정을 변경할 경우 다음을 고려하여 값을 설정하십시오.

- 대개 값이 클수록 집계 처리량이 높아지고 분리는 약해집니다. 전체 평균 지연 시간이 임계값보다 높지 않는 한 임계치 조절이 실행되지 않습니다.
- 지연 시간보다 처리량이 중요한 경우 이 값을 너무 낮게 설정하지 마십시오. 예를 들어 Fibre Channel 디스크의 경우 20ms보다 작은 값으로 설정하면 최대 디스크 처리량을 줄일 수 있습니다. 매우 높은 값(50ms 초과)은 종합적인 처리량의 별다른 향상 없이 매우 긴 지연 시간만 허용할 수 있습니다.
- 값이 작을수록 디바이스 지연 시간은 짧아지고 가상 시스템 I/O 성능 분리가 강력해집니다. 강력한 분리는 컨트롤 공유가 더욱 빈번하게 시행된다는 의미입니다. 짧은 디바이스 지연 시간은 높은 공유를 가지는 가상 시스템의 I/O 지연 시간을 줄여 주지만 낮은 공유를 가지는 가상 시스템의 I/O 지연 시간은 더 길어집니다.
- 매우 작은 값(20ms 미만)으로 설정하면 디바이스 지연 시간 및 I/O 간의 분리가 줄어들 수 있습니다. 이 경우 집계 데이터스토어 처리량이 감소할 수 있다는 단점이 있습니다.
- 아주 높은 값이나 아주 낮은 값으로 설정하면 분리 상태가 나빠집니다.

사전 요구 사항

Storage I/O Control이 활성화되어 있는지 확인합니다.

절차

- 1 vSphere Client에서 데이터스토어를 찾습니다.
- 2 구성 탭을 클릭하고 **설정**을 클릭합니다.
- 3 **일반**을 클릭합니다.
- 4 **데이터스토어 기능의 편집**을 클릭합니다.
- 5 **Storage I/O Control 사용** 확인란을 선택합니다.

Storage I/O Control은 데이터스토어가 최대 처리량의 90%에서 작동 중일 때 예상되는 지연 시간에 해당되는 지연 시간 임계값을 자동으로 설정합니다.

- 6 (선택 사항) **정체 임계값**을 조정합니다.
 - ◆ **최대 처리량 비율** 드롭다운 메뉴에서 값을 선택합니다.

최대 처리량 비율 값은 데이터스토어가 해당 비율의 예상되는 최대 처리량을 사용 중일 때 예상되는 지연 시간 임계값을 나타냅니다.

◆ 수동 드롭다운 메뉴에서 값을 선택합니다.

값은 5ms에서 100ms 사이여야 합니다. 정체 임계값을 잘못 설정하면 데이터스토어의 가상 시스템 성능이 저하될 수 있습니다.

7 (선택 사항) 정체 임계값 설정을 기본값(30ms)으로 복원하려면 **기본값으로 재설정**을 클릭합니다.

8 **확인**을 클릭합니다.

스토리지 프로파일과 Storage DRS 통합

SPBM(스토리지 정책 기반 관리)을 통해 Storage DRS에 의해 적용되는 가상 시스템 정책을 지정할 수 있습니다. 데이터스토어 클러스터에 기능 프로파일이 서로 다른 데이터스토어 집합이 있을 수 있습니다. 스토리지 프로파일 이 가상 시스템에 연결되어 있으면 Storage DRS가 기본 데이터스토어 기능을 기반으로 배치를 적용할 수 있습니다.

스토리지 프로파일과 Storage DRS 통합의 일부로 Storage DRS 클러스터 수준의 고급 옵션

`EnforceStorageProfiles`가 도입되었습니다. 고급 옵션 `EnforceStorageProfiles`가 정수 값 0, 1 또는 2 중 하나를 적용합니다. 기본값은 0입니다. 옵션이 0으로 설정되면 Storage DRS 클러스터에서 스토리지 프로파일 또는 정책이 적용되지 않음을 나타냅니다. 옵션이 1로 설정되면 Storage DRS 클러스터에서 스토리지 프로파일 또는 정책이 유동적으로 적용됨을 나타냅니다. DRS 소프트 규칙과 유사합니다. Storage DRS는 스토리지 프로파일 또는 정책을 최적의 수준으로 준수합니다. 필요한 경우 Storage DRS가 스토리지 프로파일 규정 준수를 위반합니다. 스토리지 프로파일 적용이 1로 설정된 경우에만 Storage DRS 선호도 규칙이 스토리지 프로파일보다 더 높은 우선 순위를 갖습니다. 옵션이 2로 설정되면 Storage DRS 클러스터에서 스토리지 프로파일 또는 정책이 고정적으로 적용됨을 나타냅니다. DRS 하드 규칙과 유사합니다. Storage DRS가 스토리지 프로파일 또는 정책 규정 준수를 위반하지 않습니다. 스토리지 프로파일이 선호도 규칙보다 더 높은 우선 순위를 갖습니다. Storage DRS에서 다음 장애가 발생합니다. 반선호도 규칙 위반을 수정할 수 없습니다.

사전 요구 사항

기본적으로 Storage DRS는 가상 시스템에 연결된 스토리지 정책을 적용하지 않습니다. 필요에 따라 `EnforceStorageProfiles` 옵션을 구성하십시오. 옵션은 기본(0), 소프트(1) 또는 하드(2)입니다.

절차

- 1 vSphere Client에 관리자로 로그인합니다.
- 2 vSphere Client에서 Storage DRS 클러스터를 클릭한 후 **관리 > 설정 > Storage DRS**를 선택합니다.
- 3 **편집 > 고급 옵션**을 클릭합니다.
- 4 [옵션] 머리글 아래의 영역을 클릭하고 `EnforceStorageProfiles`를 입력합니다.
- 5 이전에 입력한 고급 옵션 이름의 오른쪽에 있는 [값] 머리글 아래의 영역을 클릭하고 값 0, 1 또는 2 중 하나를 입력합니다.
- 6 **추가**를 선택합니다.

7 확인을 클릭합니다.

리소스 풀은 리소스를 유연성 있게 관리하기 위한 논리적 추상화입니다. 리소스 풀을 계층 구조로 그룹화하여 사용할 수 있는 CPU 및 메모리 리소스를 계층적으로 분할하는 데 사용할 수 있습니다.

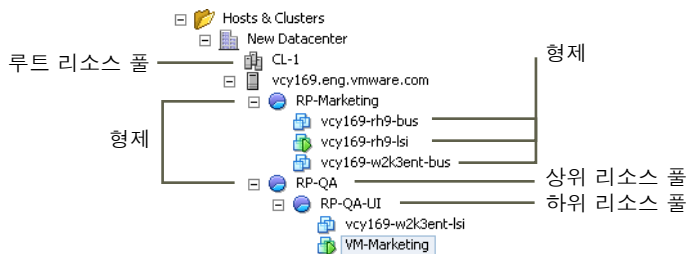
각 독립 실행형 호스트와 각 DRS 클러스터에는 해당 호스트 또는 클러스터의 리소스를 그룹화하는 보이지 않는 루트 리소스 풀이 있습니다. 루트 리소스 풀은 호스트(또는 클러스터)와 루트 리소스 풀의 리소스가 항상 동일하므로 표시되지 않습니다.

사용자는 루트 리소스 풀 또는 사용자가 만든 하위 리소스 풀의 하위 리소스 풀을 만들 수 있습니다. 각 하위 리소스 풀은 상위 리소스 풀 중 일부를 소유하므로 연속적으로 보다 작은 단위의 계산 기능을 나타내기 위한 하위 리소스 풀의 계층일 수 있습니다.

리소스 풀은 하위 리소스 풀, 가상 시스템 또는 둘 모두를 포함할 수 있습니다. 공유 리소스의 계층을 만들 수 있습니다. 더 높은 수준의 리소스 풀을 상위 리소스 풀이라고 하며, 동일한 수준의 리소스 풀 및 가상 시스템을 형제라고 합니다. 클러스터 자체는 루트 리소스 풀을 나타냅니다. 하위 리소스 풀을 만들지 않은 경우에는 루트 리소스 풀만 존재합니다.

다음 예제에서는 RP-QA가 RP-QA-UI의 상위 리소스 풀입니다. RP-Marketing 및 RP-QA는 형제입니다. RP-Marketing 바로 아래의 가상 시스템 세 개도 형제입니다.

그림 14-1. 리소스 풀 계층의 상위, 자식 및 형제



각 리소스 풀에 대해 예약, 제한, 공유 및 예약이 확장 가능한지 여부를 지정합니다. 그러면 리소스 풀의 리소스를 하위 리소스 풀 및 가상 시스템에서 사용할 수 있습니다.

참고 이 장에서 "메모리"는 물리적 RAM을 나타냅니다.

다음으로 아래 항목을 읽으십시오.

- 리소스 풀을 사용하는 원인은 무엇입니까?
- 리소스 풀 생성

- 리소스 풀 편집
- 리소스 풀에 가상 시스템 추가
- 리소스 풀에서 가상 시스템 제거
- 리소스 풀 제거
- 리소스 풀 승인 제어
- 확장 가능한 예약 예제 1
- 확장 가능한 예약 예제 2

리소스 풀을 사용하는 원인은 무엇입니까?

리소스 풀을 사용하여 호스트 또는 클러스터의 리소스에 대한 제어를 위임할 수 있는데, 리소스 풀은 클러스터의 모든 리소스를 구획화하는 데 사용했을 때 그 이점이 분명합니다. 호스트 또는 클러스터의 직계 하위로 여러 리소스 풀을 생성하여 구성합니다. 그러면 리소스 풀에 대한 제어를 다른 개인 또는 조직에 위임할 수 있습니다.

리소스 풀을 사용할 때의 이점은 다음과 같습니다.

- 유연한 계층적 구성 - 리소스 풀을 추가, 제거 또는 재구성하거나 필요에 따라 리소스 할당을 변경합니다.
- 풀 간의 분리, 풀 내에서의 공유 - 최상위 관리자는 리소스 풀을 부서 수준 관리자가 사용 가능하게 만들 수 있습니다. 한 부서 리소스 풀의 내부적 할당 변경은 관련 없는 다른 리소스 풀에 부당한 영향을 미치지 않습니다.
- 액세스 제어 및 위임 - 최상위 수준 관리자가 리소스 풀을 부서 수준 관리자가 사용할 수 있도록 만든 경우 부서 수준 관리자는 현재의 공유, 예약 및 제한 설정에 따라 리소스 풀이 권한을 가진 리소스의 경계 내에서 모든 가상 시스템 생성 및 관리를 수행할 수 있습니다. 일반적으로 위임은 사용 권한 설정과 함께 이루어집니다.
- 리소스를 하드웨어에서 분리 - DRS에 대해 사용하도록 설정된 클러스터를 사용하는 경우 모든 호스트의 리소스가 항상 클러스터에 할당됩니다. 즉, 관리자는 리소스에 기여하는 실제 호스트와 독립적으로 리소스 관리를 수행할 수 있습니다. 세 개의 2GB 호스트를 두 개의 3GB 호스트로 바꿀 때 리소스 할당을 변경할 필요가 없습니다.

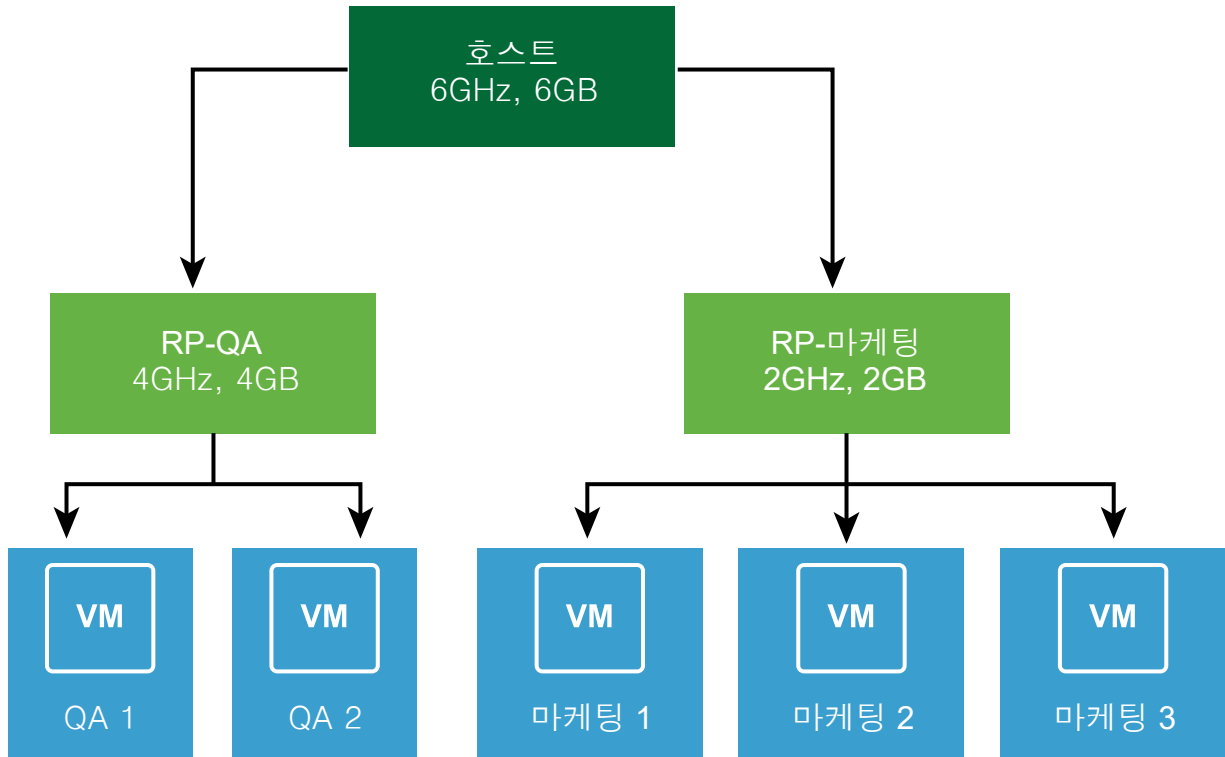
이러한 분리를 통해 관리자는 개별 호스트보다 전체 컴퓨팅 용량에 대해 더 많이 고려할 수 있습니다.

- 다계층 서비스를 실행 중인 가상 시스템 집합의 관리 - 리소스 풀의 다계층 서비스에 대해 가상 시스템을 그룹화합니다. 각 가상 시스템에 대해 리소스를 설정할 필요가 없습니다. 대신, 포함하는 리소스 풀에 대한 설정을 변경하여 가상 시스템 집합에 대한 전체 리소스 할당을 제어할 수 있습니다.

예를 들어 여러 가상 시스템이 있는 호스트를 가정하겠습니다. 마케팅 부서에서 세 개의 가상 시스템을 사용하고 QA 부서에서 두 개의 가상 시스템을 사용합니다. QA 부서에 필요한 CPU 및 메모리의 양이 더 많기 때문에 관리자는 그룹마다 하나씩 리소스 풀을 생성합니다. 관리자는 QA 부서 사용자가 자동화된 테스트를 실행할 수 있도록 QA 부서 풀의 **CPU 공유는 높음**으로 설정하고 마케팅 부서 풀에 대해서는 **일반**으로 설정합니다. CPU 및 메모리 리소스의 양이 더 적은 두 번째 리소스 풀은 사용량이 많지 않은 마케팅 직원의 작업 부하에 충분합니다. QA 부서가 할당을 완전히 사용하지 않을 때마다 마케팅 부서에서 가용 리소스를 사용할 수 있습니다.

다음 그림의 숫자는 리소스 풀에 대한 유효 할당을 보여 줍니다.

그림 14-2. 리소스 풀에 리소스 할당



리소스 풀 생성

ESXi 호스트, 리소스 풀 또는 DRS 클러스터의 하위 리소스 풀을 생성할 수 있습니다.

참고 클러스터에 호스트를 추가한 경우에는 해당 호스트의 하위 리소스 풀을 생성할 수 없습니다. 클러스터에 DRS가 설정되어 있으면 해당 클러스터의 하위 리소스 풀을 생성할 수 있습니다.

하위 리소스 풀을 만들 때 리소스 풀 특성 정보에 프롬프트됩니다. 시스템에서는 사용할 수 없는 리소스를 할당하지 못하도록 승인 제어를 사용합니다. VM을 추가 또는 제거할 때 공유의 크기를 동적으로 조정하려는 경우 확장 가능한 공유를 선택할 수 있습니다.

참고 공유는 상위 수준에서 크기가 조정됩니다. 확장 가능한 공유를 사용하여 상위에서 생성된 모든 하위 리소스 풀은 기본적으로 확장 가능한 공유를 가집니다.

사전 요구 사항

vSphere Client는 vCenter Server 시스템에 연결되어 있습니다.

절차

- 1 vSphere Client에서 리소스 풀의 상위 개체(호스트, 다른 리소스 풀 또는 DRS 클러스터)를 선택합니다.
- 2 개체를 마우스 오른쪽 버튼으로 클릭하고 **새 리소스 풀**을 선택합니다.
- 3 리소스 풀을 식별할 수 있는 이름을 입력합니다.
- 4 확장 가능한 공유를 사용하도록 설정하려는 경우 확인란을 선택합니다.
- 5 CPU와 메모리 리소스 할당 방법을 지정합니다.

리소스 풀의 CPU 리소스는 호스트가 리소스 풀을 위해 예약해 둔 보장된 물리적 리소스입니다. 일반적인 경우 기본값을 승인하고 호스트가 리소스 할당을 처리하도록 합니다.

옵션	설명
공유	상위의 총 리소스와 관련하여 이 리소스 풀에 공유를 지정합니다. 형제 리소스 풀은 예약이나 한도로 바인딩된 상대적 공유 값에 따라서 리소스를 공유합니다. <ul style="list-style-type: none"> ■ 각각이 1:2:4 비율의 공유값을 지정하도록 낮음, 정상 또는 높음을 선택합니다. ■ 각 가상 시스템에 특정 공유 개수(비례 가중치)를 지정하려면 사용자 지정을 선택합니다.
예약	이 리소스 풀의 메모리 할당이나 보장된 CPU를 지정합니다. 기본값을 0으로 합니다. 0이 아닌 예약은 상위(호스트나 리소스 풀)의 예약되지 않은 리소스에서 차감됩니다. 가상 시스템과 리소스 풀의 연결 상태에 관계없이 리소스는 예약된 것으로 간주됩니다.
확장 가능한 예약	이 확인란이 선택된 경우(기본값), 승인 제어 중에 확장 가능한 예약이 고려됩니다. 리소스 풀에서 가상 시스템의 전원을 켜고 가상 시스템의 결합 예약이 리소스 풀의 예약보다 큰 경우에, 리소스 풀은 상위 리소스를 사용할 수 있습니다.
제한	리소스 풀의 CPU나 메모리 할당의 상한 값을 지정합니다. 일반적으로 기본값(제한 없음)을 승인할 수 있습니다. 제한을 지정하려면 제한 없음 확인란을 선택 취소합니다.

6 확인을 클릭합니다.

결과

리소스 풀을 생성한 후에는 리소스 풀에 가상 시스템을 추가할 수 있습니다. 가상 시스템의 공유는 상위 리소스 풀이 동일한 다른 가상 시스템(또는 리소스 풀)에 대해 상대적입니다.

예제: 리소스 풀 생성

호스트가 제공하는 6GHz CPU와 3GB 메모리를 마케팅 부서 및 QA 부서가 공유해야 한다고 가정합니다. 또한 두 부서 중 하나(QA)에 우선 순위를 높게 두어 리소스를 균등하지 않게 공유하려고 합니다. 이와 같이 하려면 각 부서에 대해 리소스 풀을 생성하고 **공유** 특성을 사용하여 리소스 할당 우선 순위를 지정할 수 있습니다.

이 예에서는 ESXi 호스트를 상위 리소스로 하여 리소스 풀을 생성하는 방법을 보여 줍니다.

1 **새 리소스 풀** 대화상자에서 QA 부서의 리소스 풀 이름(예: RP-QA)을 입력합니다.

2 RP-QA의 CPU 및 메모리 리소스에 대해 **공유**를 **높음**으로 지정합니다.

3 두 번째 리소스 풀인 RP-Marketing을 생성합니다.

CPU 및 메모리 공유를 **보통**으로 둡니다.

4 **확인**을 클릭합니다.

리소스 경합이 발생하면 RP-QA에는 4GHz 및 2GB 메모리가 할당되고 RP-Marketing에는 2GHz 및 1GB가 할당됩니다. 그렇지 않은 경우에는 이보다 더 많은 양을 할당받을 수 있습니다. 그러면 해당 리소스 풀의 가상 시스템에서 리소스를 사용할 수 있습니다.

리소스 풀 편집

리소스 풀을 생성한 후에 리소스 풀의 CPU 및 메모리 리소스 설정을 편집할 수 있습니다.

절차

1 vSphere Client에서 리소스 풀을 찾습니다.

2 **작업** 드롭다운 메뉴에서 **리소스 설정 편집**을 선택합니다.

3 (선택 사항) **리소스 풀 생성**에 설명된 대로 선택한 리소스 풀의 모든 특성을 변경할 수 있습니다.

- 확장 가능한 공유를 사용하도록 설정하려는 경우 **확인란**을 선택합니다.

참고 공유는 상위 수준에서 크기가 조정됩니다. 확장 가능한 공유를 사용하여 상위에서 생성된 모든 하위 리소스 풀은 기본적으로 확장 가능한 공유를 가집니다.

- **CPU**에서 CPU 리소스 설정을 선택합니다.
- ◆ **메모리**에서 메모리 리소스 설정을 선택합니다.

4 **확인**을 클릭하여 변경 내용을 저장합니다.

리소스 풀에 가상 시스템 추가

가상 시스템을 생성할 때 생성 프로세스의 일부로 리소스 풀 위치를 지정할 수 있습니다. 또한 기존 가상 시스템을 리소스 풀에 추가할 수도 있습니다.

다음 내용은 가상 시스템을 새 리소스 풀로 이동할 경우에 적용됩니다.

- 가상 시스템의 예약 및 제한은 변경되지 않습니다.
- 가상 시스템의 공유가 높음, 중간 또는 낮음인 경우 새 리소스 풀에 사용되는 총 공유 수를 반영하도록 %Shares가 조정됩니다.
- 가상 시스템에 사용자 지정 공유가 할당되어 있는 경우 해당 공유 값이 유지됩니다.

참고 공유는 리소스 풀에 상대적으로 할당되기 때문에 가상 시스템을 새 리소스 풀로 이동한 후에는 새 리소스 풀의 상대 값에 대해 일관되도록 가상 시스템의 공유를 수동으로 변경해야 할 수 있습니다. 총 공유 중 가상 시스템에 할당되는 비율이 매우 크거나 매우 작으면 주의 메시지가 나타납니다.

- **모니터** 아래의 **리소스 예약** 탭에서 리소스 풀의 예약되거나 예약되지 않은 CPU 및 메모리 리소스에 대해 표시되는 정보는 가상 시스템과 관련된 예약이 있는 경우 이를 반영하도록 변경됩니다.

참고 가상 시스템의 전원이 꺼지거나 일시 중단되면 해당 가상 시스템을 이동할 수 있지만 이 경우에는 리소스 풀의 사용 가능한 전체 리소스(예: 예약/예약되지 않은 CPU 및 메모리)는 아무런 영향을 받지 않습니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
 - a 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - b **VM** 탭을 클릭합니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **마이그레이션**을 클릭합니다.
 - 가상 시스템을 다른 호스트로 이동할 수 있습니다.
 - 가상 시스템의 스토리지를 다른 데이터스토어로 이동할 수 있습니다.
 - 가상 시스템을 다른 호스트로 이동한 후 해당 스토리지를 다른 데이터스토어로 이동할 수 있습니다.
- 3 가상 시스템을 실행할 리소스 풀을 선택합니다.
- 4 선택 사항을 검토하고 **마침**을 클릭합니다.

결과

가상 시스템의 전원이 켜져 있고 대상 리소스 풀에 가상 시스템의 예약을 보장할 수 있을 만큼 CPU나 메모리가 충분하지 않으면 승인 제어가 이동을 허용하지 않습니다. [오류] 대화상자에 사용 가능하고 요청된 리소스가 표시되기 때문에 조정을 통해 문제가 해결될 수 있는지 고려할 수 있습니다.

리소스 풀에서 가상 시스템 제거

가상 시스템을 다른 리소스 풀로 이동하거나 삭제하여 리소스 풀에서 가상 시스템을 제거할 수 있습니다.

리소스 풀에서 가상 시스템을 제거하면 해당 리소스 풀과 관련된 총 공유 수가 감소하여 남아 있는 공유가 더 많은 리소스를 나타냅니다. 예를 들어 6GHz를 사용할 수 있고 공유가 **일반**으로 설정된 세 개의 가상 시스템이 포함되어 있는 풀이 있다고 가정하겠습니다. 가상 시스템이 CPU 바인딩되어 있고 각각 2GHz씩 동일하게 할당받는다고 가정합니다. 이 가상 시스템 중 하나를 다른 리소스 풀로 이동하면 나머지 두 개의 가상 시스템은 각각 3GHz씩 동일하게 할당받습니다.

절차

- 1 vSphere Client에서 리소스 풀을 찾습니다.
- 2 리소스 풀에서 가상 시스템을 제거하려면 다음 방법 중 하나를 선택합니다.
 - 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **다음으로 이동...**을 선택하여 가상 시스템을 다른 리소스 풀로 이동합니다.
가상 시스템을 이동하기 전에 전원을 차단할 필요는 없습니다.
 - 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **디스크에서 삭제**를 선택합니다.
가상 시스템을 완전히 제거하기 전에 가상 시스템의 전원을 꺼야 합니다.

리소스 풀 제거

인벤토리에서 리소스 풀을 제거할 수 있습니다.

절차

- 1 vSphere Client에서 리소스 풀을 마우스 오른쪽 버튼으로 클릭하고 **삭제**를 선택합니다.
[확인] 대화상자가 나타납니다.
- 2 리소스 풀을 제거하려면 **예**를 클릭합니다.

리소스 풀 승인 제어

리소스 풀의 가상 시스템 전원을 켜거나 하위 리소스 풀을 생성하려고 하면 풀의 제한을 위반하지 않도록 시스템에서 추가 승인 제어를 수행합니다.

가상 시스템의 전원을 켜거나 리소스 풀을 생성하기 전에 vSphere Client의 **리소스 예약** 탭을 통해 사용 가능한 리소스가 충분한지 확인하십시오. CPU 및 메모리의 **사용 가능한 예약** 값은 예약되지 않은 리소스를 나타냅니다.

사용 가능한 CPU 및 메모리 리소스가 계산되는 방법과 작업이 수행되는지 여부는 **예약 유형**에 따라 다릅니다.

표 14-1. 예약 유형

예약 유형	설명
고정	시스템은 선택한 리소스 풀에 예약되지 않은 리소스가 충분히 있는지 여부를 확인합니다. 충분하면 작업을 수행할 수 있고, 충분하지 않으면 메시지가 나타나고 작업을 수행할 수 없습니다.
확장 가능 (기본값)	시스템은 선택한 리소스 풀 및 해당 리소스 풀의 직접적인 상위 리소스 풀에서 사용 가능한 리소스를 고려합니다. 상위 리소스 풀에 확장 가능한 예약 옵션이 선택되어 있으면 상위 리소스 풀에서 리소스를 빌릴 수 있습니다. 확장 가능한 예약 옵션이 선택되어 있는 한 현재 리소스 풀의 상위 리소스 풀에서 리소스를 빌리는 동작이 반복적으로 일어납니다. 이 옵션을 선택된 상태로 두면 유연성은 더 높아지는 반면에 보호 수준이 낮아집니다. 하위 리소스 풀 소유자는 예상보다 더 많은 리소스를 예약할 수 있습니다.

시스템은 미리 구성된 **예약** 또는 **제한** 설정을 위반하는 것을 허용하지 않습니다. 리소스 풀을 재구성하거나 가상 시스템의 전원을 켜 때마다 시스템에서 모든 매개 변수를 검증하므로 모든 서비스 수준 보증이 여전히 충족됩니다.

확장 가능한 예약 예제 1

이 예제에서는 확장 가능한 예약이 포함된 리소스 풀의 작동 방식을 보여 줍니다.

관리자가 P라는 풀을 관리하고 다른 두 사용자나 두 그룹을 위한 두 개의 하위 리소스 풀인 S1과 S2를 정의한다고 가정합니다.

사용자가 예약이 있는 가상 시스템의 전원을 켜려고 한다는 점은 관리자가 알고 있지만 각 사용자가 예약해야 할 크기는 알지 못합니다. S1과 S2의 예약을 확장 가능하게 설정하면 관리자가 P 풀에 대한 일반 예약을 더 유연하게 공유하고 상속할 수 있습니다.

확장 가능한 예약이 없으면 관리자가 명시적으로 S1 및 S2에 특정 크기를 할당해야 합니다. 이러한 특정 할당은 특히 심층적인 리소스 풀 계층에서 유연성이 떨어져 리소스 풀 계층에서 예약 설정 과정을 복잡하게 만들 수 있습니다.

확장 가능한 예약으로 인해 엄격한 분리가 손실될 수 있습니다. S1은 모든 P 예약을 사용할 수 있으므로 S2에서 CPU 또는 메모리를 직접 사용할 수 없습니다.

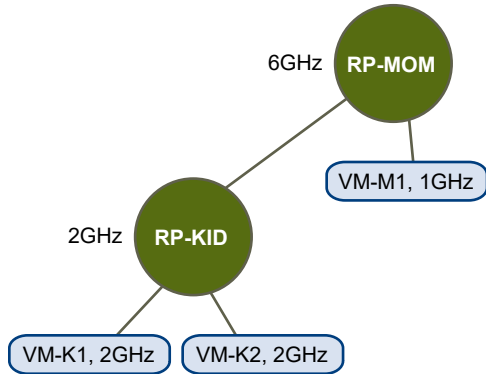
확장 가능한 예약 예제 2

확장 가능한 할당 기능이 있는 리소스 풀이 어떻게 작업하는 지를 보여줍니다.

그림과 같이 다음 시나리오가 있다고 가정합니다.

- 상위 풀 RP-MOM에는 6GHz 예약이 있으며 1GHz가 예약된 하나의 시스템 VM-M10이 실행 중입니다.
- 2GHz 예약이 있고 **확장 가능한 예약**이 선택되어 있는 하위 리소스 풀 RP-KID를 생성합니다.
- 각각 2GHz 예약이 있는 두 가상 시스템 VM-K1 및 VM-K2를 하위 리소스 풀에 추가하고 전원을 켭니다.
- VM-K1은 2GHz가 있는 RP-KID에서 직접 리소스를 예약할 수 있습니다.
- VM-K2에는 로컬 리소스를 사용할 수 없으므로 상위 리소스 풀 RP-MOM에서 리소스를 빌려 옵니다. RP-MOM에서는 6GHz에서 1GHz(가상 시스템에 예약됨)와 2GHz(RP-KID에 예약됨)를 뺀 3GHz가 예약되지 않은 상태입니다. 3GHz를 사용할 수 있는 상태에서 2GHz 가상 시스템의 전원을 켤 수 있습니다.

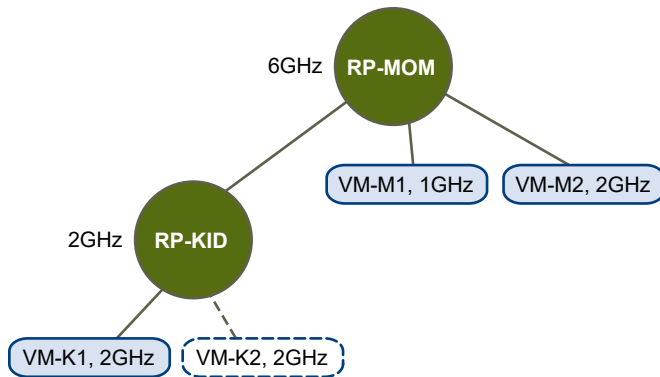
그림 14-3. 확장 가능한 리소스 풀을 사용한 승인 제어: 전원 켜기 성공



이제 VM-M1 및 VM-M2와 관련된 또 다른 시나리오를 고려해 봅니다.

- 총 3GHz가 예약된 RP-MOM에서 두 가상 시스템의 전원을 켭니다.
- 2GHz를 로컬로 사용할 수 있기 때문에 계속 RP-KID에서 VM-K1의 전원을 켤 수 있습니다.
- VM-K2의 전원을 켜려고 시도하면 RP-KID에 미예약 CPU 용량이 없으므로 RP-KID에서 상위 항목을 검사합니다. RP-MOM에서는 1GHz의 미예약 용량만 사용할 수 있습니다. (5GHz의 RP-MOM은 이미 사용 중인데, 5GHz 중 3GHz는 로컬 가상 시스템에 예약되어 있고 2GHz는 RP-KID에 예약되어 있습니다.) 따라서 2GHz 예약이 필요한 VM-K2의 전원을 켤 수 없습니다.

그림 14-4. 확장 가능한 리소스 풀을 사용한 승인 제어: 전원 켜기 차단



vCLS(vSphere 클러스터 서비스)는 기본적으로 활성화되고 모든 vSphere 클러스터에서 실행됩니다. vCLS는 vCenter Server를 사용할 수 없게 되는 경우, 클러스터에서 실행되는 워크로드의 리소스와 상태를 유지하기 위해 클러스터 서비스를 계속 사용할 수 있도록 합니다. DRS 및 HA를 구성하고 실행하려면 계속 vCenter Server가 필요합니다.

vSphere 8.0 U3에서는 내장된 vCLS에 새로운 기능이 도입되었습니다. 원래 vCLS 구성 요소는 Photon OS를 실행하는 완전한 가상 시스템이었습니다. vSphere 8.0 U3의 경우 vCLS는 이제 때때로 PodCRX라고도 하는 vSphere 포드 기술을 기반으로 합니다. 이 문서에서 PodCRX는 vCLS VM을 뜻합니다. vCLS용 API 인터페이스는 변경되지 않았지만 이러한 vCLS 구성 요소의 배포 및 관리는 변경되었습니다. 이로 인해 이름 변경이 필요했습니다. Platform Service Controller가 외장형에서 내장형으로 전환되었을 때 만들어진 선례에 따라 vCLS는 이제 vSphere 8.0 U3 이전 릴리스에서 외부 vCLS를 뜻합니다. vSphere 8.0 U3 이상부터 vCLS를 이제 내장된 vCLS라고 합니다.

vCLS는 vCenter Server 업그레이드의 일부로 업그레이드됩니다. 내장된 vCLS의 경우 vCLS 업그레이드는 클러스터의 ESXi 호스트 업그레이드와 관련이 있습니다. 내장된 vCLS는 VC가 8.0 U3이고 하나 이상의 8.0 U3 ESXi 호스트가 있는 클러스터에서 활성화됩니다. 8.0 U3 VC는 클러스터 구성에 따라 외부 vCLS 및 내장된 vCLS를 실행할 수 있습니다.

vCenter는 ESX 8.0 U3 이상을 실행하는 지원 호스트 클러스터에 내장된 vCLS를 배포합니다. 이전 릴리스를 사용하는 호스트 클러스터의 경우 vCenter는 외부 vCLS를 배포합니다. vCenter 인벤토리를 사용하면 ESX 롤링 업그레이드 중에 여러 버전의 ESX가 공존할 수 있습니다. 혼합 버전 클러스터의 경우 vCenter는 클러스터에서 사용 가능한 호스트가 이를 지원할 때마다 내장된 vCLS를 사용합니다. 이로 인해 호스트가 지원되는 버전으로 업그레이드한 후 유지 보수 모드를 종료할 때 클러스터가 외부 vCLS에서 내장된 vCLS로 업그레이드되는 지점이 생성됩니다. vCenter는 첫 번째 내장된 vCLS VM을 사용할 수 있을 때까지 기다린 후 외부 vCLS를 비활성화하고 해당 VM을 제거하므로 이러한 업그레이드가 원활하게 수행됩니다. 또한 지원되는 모든 호스트를 사용할 수 없게 되고 지원되지 않는 호스트만 남게 되면 vCenter는 외부 vCLS로 되돌릴 수 있습니다. 그러나 이 단계는 호스트를 사용할 수 없게 되는 것을 기반으로 하기 때문에 내장된 vCLS 해체와 외부 vCLS 배포 사이에 DRS를 사용할 수 없는 기간이 발생할 수 있습니다.

vCLS는 에이전트 가상 시스템을 사용하여 클러스터 서비스 상태를 유지합니다. vCLS 에이전트 가상 시스템(vCLS VM)은 클러스터에 호스트를 추가할 때 생성됩니다. 외부 vCLS에서는 클러스터 내에 분산된 각 vSphere 클러스터에서 실행하려면 최대 3개의 vCLS VM이 필요합니다. 외부 vCLS는 1개 또는 2개의 호스트만 포함하는 클러스터에서도 활성화됩니다. 이러한 클러스터에서 vCLS VM 수는 각각 1개, 2개입니다. 내장된 vCLS에서는 각 vSphere 클러스터에서 실행하려면 최대 2개의 vCLS VM이 필요합니다.

새로운 반선호도 규칙이 자동으로 적용됩니다. 외부 vCLS에서는 3분마다 검사가 수행되고, 단일 호스트에 여러 vCLS VM이 있으면 서로 다른 호스트에 자동으로 재분산됩니다. 내장된 vCLS에서는 1분마다 검사가 수행됩니다.

vSphere DRS 또는 vSphere HA와 같은 클러스터 서비스가 클러스터에서 활성화되지 않은 경우에도 모든 클러스터에서 vCLS VM이 실행됩니다. vCLS VM의 수명 주기 작업은 ESX Agent Manager 및 워크로드 제어부와 같은 vCenter Server 서비스에서 관리됩니다. vCLS VM은 NIC를 지원하지 않습니다.

vCLS에서 활성화된 클러스터에는 ESXi 버전이 vCenter Server와 호환되는 경우 서로 다른 버전의 ESXi 호스트가 포함될 수 있습니다. vCLS는 vSphere Lifecycle Manager 클러스터에서 작동합니다.

다음으로 아래 항목을 읽으십시오.

- 내장된 vCLS
- vSphere DRS 및 vCLS VM
- 외부 vCLS에 대한 데이터스토어 선택
- 외부 vCLS 데이터스토어 배치
- vSphere 클러스터 서비스 모니터링
- vSphere 클러스터 서비스의 상태 유지
- 클러스터를 철회 모드로 전환
- 외부 vCLS의 암호 검색
- vCLS VM 반선호도 정책
- vCLS VM 반선호도 정책 생성 또는 삭제

내장된 vCLS

vSphere 8.0 U3 내장된 vCLS가 도입되면서 새로운 기능과 향상된 기능을 제공합니다.

외부 vCLS와 내장된 vCLS의 차이점

외부 vCLS에서 **quorum**이라는 용어는 DRS가 작동하는 데 필요한 전원이 켜진 vCLS VM의 최소 수(1)를 나타냅니다. 내장된 vCLS에서는 **minimum vCLS VM count**라는 용어를 사용하는데, 해당 용도와 값 1은 변경되지 않습니다. DRS 가용성을 하기 위해 두 가지 유형의 vCLS 모두 VM에 이중화를 제공합니다. 하나의 vCLS VM이 있는 호스트에 장애가 발생하면 DRS 기능을 보존하는 데 사용할 수 있는 다른 vCLS VM이 있습니다. 중복 vCLS VM을 배포하기에 충분한 사용 가능한 호스트 및 리소스가 있다고 가정하여 클러스터에 배포된 총 vCLS VM 수를 원하는 이중화 수라고 합니다. 외부 vCLS의 경우 원하는 이중화 수는 3입니다. 내장된 vCLS의 경우 이 수가 2로 줄어듭니다.

vCLS VM이란?

vSphere 8.0 업데이트 3에 도입된 vCLS VM은 vSphere 포드 기술을 기반으로 합니다. vCLS 서비스는 호스트 또는 클러스터에 ESXi 호스트의 가상 시스템으로 배포됩니다. 그러나 이는 아주 최소한의 운영 체제를 사용하는 특수 VM이며 컨테이너 런타임을 사용하여 vCLS VM을 vSphere에서 실행할 수 있습니다. 새 vCLS VM은 안전한 고성능 런타임을 제공합니다.

vCLS VM 전원 끄기

내장된 vCLS VM의 전원을 끌 수 있습니다. 이러한 VM의 전원이 꺼지면 vSphere가 이 조건을 실패로 해석한 후 계속해서 VM을 다시 시작하고 hostd에 vCLS VM을 다시 등록합니다.

vCLS VM의 전원이 꺼지면 vSphere가 해당 VM을 다시 시작할 때까지 기다립니다. 시간 초과 후 vSphere가 해당 VM을 다시 시작하지 않으면 vCLS VM 재구성이 시작됩니다.

유지 보수 모드로 전환되는 호스트와 같이 시스템에 의해 내장된 vCLS VM이 제거될 때마다 vMotion을 사용하는 대신 다른 호스트에서 제거되고 교체됩니다.

참고 이 동작은 vCLS가 VM 전원 끄기를 감지하고 해당 VM의 전원 켜기를 즉시 진행하는 외부 vCLS 동작과는 다릅니다.

vCLS VM

내장된 vCLS의 vCLS VM에 대한 가시성 및 액세스는 외부 vCLS와 동등합니다.

- vCLS VM은 VC에 표시됩니다.
- VC 인벤토리 계층에서 vCLS VM은 **vCLS**라는 전용 VM 폴더에 상주합니다.
- vCLS VM에 대한 일부 작업을 사용할 수 없습니다.
- vCLS VM에 대한 쿼리 요청을 사용할 수 있습니다.
- 내장된 vCLS VM의 이름 지정은 외부 vCLS VM: **vCLS-{UUID}**와 동일합니다. 여기서 UUID는 ESX 호스트의 UUID입니다(summary.hardware.uuid).

VM ExtraConfig

vCLS VM을 식별하기 위해 내장된 vCLS는 VM ExtraConfig의 특정 옵션을 계속 사용합니다.

내장된 vCLS VM을 식별하는 추가 구성 옵션은 **vCLSCRX.agent**이고 값은 **true**입니다.

참고 외부 vCLS VM 추가 구성 옵션(**HDCS.agent**) 외에 이 추가 구성 옵션이 설정됩니다. 내장된 vCLS VM과 외부 vCLS VM은 둘 다 **HDCS.agent**를 사용하여 대부분의 작업에서 제외되도록 감지되어야 할 만큼 유사하지만 일부 작업은 여전히 **vCLSCRX.agent**를 사용하여 구별해야 할 정도로 다릅니다.

내장된 vCLS는 이러한 VM을 관리하는 확장을 내부/시스템 vpxd 확장으로 변경합니다.

- 내장된 vCLS VM의 VM 구성에서 **managedBy** 속성이 해당 VC 확장의 키로 설정됩니다. 이 속성을 사용하면 내장된 vCLS VM을 식별하고 외부 vCLS VM과 구별할 수 있습니다. 이 속성이 vSphere에 표시되기 때문입니다.

- `config.managedBy.extensionKey` 속성은 `VirtualCenter`로 설정됩니다.
- `config.managedBy.type` 속성은 `vcls-entity`로 설정됩니다.

연결이 끊긴 vCLS VM

VC에서 vCLS VM의 연결이 끊길 수 있습니다. 다양한 시나리오에서 이 문제가 발생할 수 있습니다.

- ESX 호스트에서 vCLS VM 실행
- 호스트에서 vpxa를 중지하여 연결 문제를 시뮬레이션합니다. 연결 상태가 `notResponding`으로 전환됩니다.
- 호스트에서 vCLS VM을 중지합니다.
- 호스트에서 vpxa를 다시 시작합니다. 연결 상태가 `connected`로 전환됩니다.

위의 단계 후에는 vCLS VM이 VC 인벤토리에 있지만 ESX 호스트에는 없습니다. 따라서 이 VM은 `orphaned`로 표시됩니다. HdcsManager는 연결이 끊긴 vCLS VM을 감지하고 삭제하는 작업을 담당합니다. 외부 vCLS에서 vCLS는 연결이 끊긴 vCLS VM을 감지하고 삭제합니다.

클러스터 가상 속성 `vclsVmType`

vSphere는 클러스터가 외부 vCLS에서 관리되는 vCLS VM을 실행 중인지 아니면 vSphere에서 관리되는 내장된 vCLS를 실행 중인지에 대한 정보를 제공합니다. 이 속성의 값은 내장된 vCLS VM 클러스터의 경우 `Embedded`이고, 외부 vCLS VM 클러스터의 경우 `external`입니다.

vCLS VM 정리

vCLS VM을 실행하는 클러스터 ESX 호스트가 VC와의 연결이 끊어지고 VC 인벤토리에서 제거되는 경우 vCLS VM은 해당 ESX 호스트에서 계속 실행됩니다. 즉, 관리되지 않는 ESX 호스트가 vCLS VM을 실행하게 될 수 있습니다.

vCLS VM을 실행 중인 관리되지 않는 ESX 호스트가 독립형 호스트로 추가되면 vpxd는 예기치 않은 vCLS VM의 존재를 감지하고 중지합니다.

vCLS VM을 실행 중인 관리되지 않는 ESX 호스트가 VC 클러스터에 추가되면 vCLS는 클러스터에 호스트를 추가하기 전에 vCLS VM을 중지하지 않습니다. 호스트를 추가하면 vCLS 재구성 워크플로가 트리거됩니다. 이 워크플로는 관련 없는 vCLS VM을 중지합니다.

vCLS 비활성화

참고 외부 vCLS에서는 vCLS를 비활성화하는 기능(전역적으로 또는 클러스터당)을 **Retreat Mode**라고 합니다.

외부 vCLS는 클러스터별로 vCLS VM 배포를 비활성화할 수 있는 기능을 구현합니다. 어떤 의미에서 이 기능은 예기치 않은 상황이 발생했을 때 외부 vCLS 기능을 비활성화하기 위한 중지 스위치입니다. 이 기능이 유용할 수 있는 몇 가지 샘플 시나리오:

- 클러스터에 DRS와 HA가 모두 비활성화되었습니다. 이 경우 vCLS VM은 아무 쓸모가 없으며 VC 인벤토리에 시스템 VM이 존재할 필요가 없습니다.

- 클러스터에 DRS가 비활성화되었으며 HA가 활성화되었습니다. 이 경우 vCLS VM은 VM을 최적으로 페일오버하는 데 일부 도움이 되지만 미미한 이점 때문에 VC 인벤토리에 시스템 VM이 존재하는 것을 원치 않을 수 있습니다.
- 클러스터에 DRS가 활성화되었습니다. 이 경우 외부 vCLS 배포를 일시적으로 비활성화하여 일부 일시적 구성 또는 일시적 런타임 문제를 해결할 수 있습니다. 예를 들어 VSAN 데이터스토어를 사용할 수 있게 될 때까지 외부 vCLS를 비활성화하여 vCLS VM이 해당 VSAN 데이터스토어에 배포되도록 할 수 있습니다.

내장된 vCLS는 클러스터에 대해 vCLS를 활성화하거나 비활성화하기 위해 동일한 기능을 유지합니다.

vSphere DRS 및 vCLS VM

vSphere DRS는 vSphere 클러스터 내에서 실행되는 워크로드의 상태를 유지 관리하는 데 필요한 vSphere의 중요 기능입니다. DRS는 vCLS VM의 가용성에 따라 다릅니다.

참고 vCLS VM에 문제가 있는 클러스터에서 DRS를 활성화하려고 하면 **클러스터 요약** 페이지에 주의 메시지가 표시됩니다.

참고 DRS가 켜져 있지만 vCLS VM에 문제가 있는 경우 DRS가 작동하려면 이러한 문제를 해결해야 합니다. **클러스터 요약** 페이지에 주의 메시지가 표시됩니다.

DRS가 작동하지 않는다고 해서 DRS가 비활성화된 것은 아닙니다. vCLS 상태는 vCLS VM이 실행되고 있지 않으며 이로 인해 DRS의 첫 번째 인스턴스를 건너뛰는 경우에만 DRS가 활성화된 클러스터에서 **비정상**으로 바뀝니다. vCLS 상태는 하나 이상의 vCLS VM이 실행되고 있지 않을 때 비DRS 비활성화 클러스터에서 **성능 저하**됨으로 유지됩니다. vCenter에는 실수로 성능 저하 상태가 되는 것을 방지하기 위한 조치가 있습니다. 예를 들어 클러스터 유지 보수 모드 권장 사항에는 모든 vCLS 지원 호스트를 한 번에 유지 보수 모드로 전환하는 옵션이 포함되지 않습니다.

외부 vCLS에 대한 데이터스토어 선택

vCLS VM에 대한 데이터스토어는 클러스터 내의 호스트에 연결된 모든 데이터스토어 순위 지정에 따라 자동으로 선택됩니다.

데이터스토어에 연결된 사용 가능한 예약된 DRS 슬롯이 있는 클러스터에 호스트가 있는 경우 데이터스토어가 선택될 가능성이 높습니다. 알고리즘은 로컬 데이터스토어를 선택하기 전에 가능하면 공유 데이터스토어에 vCLS VM을 배치하려고 시도합니다. 사용 가능한 공간이 더 많은 데이터스토어가 선호되며 알고리즘이 동일한 데이터스토어에 둘 이상의 vCLS VM을 배치하지 않으려고 합니다. vCLS VM의 데이터스토어는 배포하고 전원을 켜 후에 만 변경할 수 있습니다.

vCLS VM에 대한 VMDK를 다른 데이터스토어로 이동하거나 다른 스토리지 정책을 연결하려는 경우 vCLS VM을 재구성할 수 있습니다. 이 작업을 수행할 때 주의 메시지가 표시됩니다.

Storage vMotion을 수행하여 vCLS VM을 다른 데이터스토어로 마이그레이션할 수 있습니다. 워크로드 VM과 별도로 그룹화하려는 경우, 예를 들어 데이터 센터에서 실행되는 모든 VM에 대한 특정 메타데이터 전략이 있는 경우, vCLS VM에 태그를 지정하거나 사용자 지정 특성을 연결할 수 있습니다.

참고 데이터스토어가 유지 보수 모드로 전환되고 데이터스토어가 vCLS VM을 호스팅하는 경우 수동으로 Storage vMotion을 vCLS VM에 적용하여 새 위치로 이동하거나 클러스터를 철회 모드로 전환해야 합니다. 주의 메시지가 표시됩니다.

유지 보수 모드 시작 작업이 시작되지만 데이터스토어에 상주하는 1개의 가상 시스템이 있기 때문에 이 작업을 마칠 수 없습니다. 계속 진행하기로 결정한 경우에는 언제든지 [최근 작업]에서 작업을 취소할 수 있습니다.

선택한 데이터스토어가 전원을 끌 수 없는 vSphere 클러스터 서비스 VM을 저장할 수 있습니다. vSphere 클러스터 서비스의 상태를 보장하려면 유지 보수를 위해 이 데이터스토어를 다운시키기 전에 클러스터 내의 다른 데이터스토어로 이러한 VM을 수동으로 vMotion해야 합니다. KB 문서 79892를 참조하십시오.

계속하려면

모든 가상 시스템의 스토리지를 마이그레이션하고 마이그레이션 후 다시 유지 보수 모드로 전환하도록 허용을 선택합니다.

참고 내장된 vCLS는 데이터스토어를 활용하지 않으며 배치 설정을 무시합니다.

외부 vCLS 데이터스토어 배치

기본 vCLS VM 데이터스토어 배치를 재정의할 수 있습니다.

vCLS(vSphere 클러스터 서비스) VM 데이터스토어 위치는 기본 데이터스토어 선택 논리에 의해 선택됩니다. 클러스터의 기본 vCLS VM 데이터스토어 배치를 재정의하려면 클러스터로 이동하고 **구성 > 클러스터 서비스 > 데이터스토어**에서 **추가**를 클릭하여 허용되는 데이터스토어 집합을 지정할 수 있습니다. 일부 데이터스토어는 vCLS에 대해 선택할 수 없습니다. vCLS를 구성할 수 없는 SRM 또는 vSAN 유지 보수 모드와 같은 솔루션에 의해 차단되어 있기 때문입니다. 사용자는 vCLS VM에 대해 솔루션이 차단된 데이터스토어를 추가하거나 제거할 수 없습니다.

vSphere 클러스터 서비스 모니터링

vCLS VM에서 사용하는 리소스와 해당 상태를 모니터링할 수 있습니다.

vCLS VM은 **호스트 및 클러스터** 탭의 인벤토리 트리에 표시되지 않습니다. 데이터 센터 내 모든 클러스터의 vCLS VM은 **vCLS**라는 별도의 VM 및 템플릿 폴더 내에 배치됩니다. 이 폴더와 vCLS VM은 vSphere Client의 **VM 및 템플릿** 탭에만 표시됩니다. 이러한 VM은 일반 워크로드 VM과 다른 아이콘으로 식별됩니다. vCLS VM의 **요약** 탭에서 vCLS VM의 용도에 대한 정보를 볼 수 있습니다.

모니터 탭에서 vCLS VM에 사용되는 리소스를 모니터링할 수 있습니다.

클러스터의 **요약** 탭에 표시되는 **클러스터 서비스** 포틀릿에서 vCLS의 상태를 모니터링할 수 있습니다.

표 15-1. vCLS의 상태

상태	색상 코딩	요약
정상	녹색	실행 중인 vCLS VM이 하나 이상 있으면 클러스터의 호스트 수에 관계없이 상태가 정상으로 유지됩니다.
성능 저하됨	노란색	3분(180초) 미만으로 실행되는 vCLS VM이 없으면 성능 저하됨 상태입니다.
비정상	빨간색	3분 이상 실행되는 vCLS VM이 없으면 DRS 지원 클러스터에서 비정상 상태입니다.

vSphere 클러스터 서비스의 상태 유지

vCLS VM은 항상 전원이 켜져 있습니다. vSphere DRS는 이러한 VM의 가용성에 의존하기 때문입니다. 이러한 VM은 시스템 VM으로 취급해야 합니다. 관리자만 vCLS VM에 대해 선택적 작업을 수행할 수 있습니다. 클러스터 서비스의 장애를 방지하려면 vCLS VM에서 구성 또는 작업을 수행하지 마십시오.

vCLS VM은 실수로 삭제되지 않도록 보호됩니다. 클러스터 VM 및 폴더는 관리자를 포함한 사용자가 수정하지 못하도록 보호됩니다.

관리자 SSO 그룹에 속한 사용자만 다음 작업을 수행할 수 있습니다.

- vCLS VM에 대한 읽기 전용 액세스
- vCLS VM에 태그 및 사용자 지정 특성 사용

vCLS VM의 정상적인 작동을 방해할 수 있는 작업:

- vCLS VM의 전원 상태 변경
- vCLS VM의 리소스 재구성(예: CPU, 메모리, 디스크 크기, 디스크 배치 변경)
- VM 암호화
- vCLS VM의 vMotion 트리거
- BIOS 변경
- 인벤토리에서 vCLS VM 제거
- 디스크에서 vCLS VM 삭제
- vCLS VM의 FT 사용
- vCLS VM 복제
- PMem 구성
- vCLS VM을 다른 폴더로 이동
- vCLS VM 이름 변경
- vCLS 폴더 이름 변경

- vCLS VM에서 DRS 규칙 및 재정의 사용
- vCLS VM에서 HA 승인 제어 정책 사용
- vCLS VM에서 HA 재정의 사용
- vCLS VM을 리소스 풀로 이동
- 스냅샷에서 vCLS VM 복구

vCLS VM에서 중단 작업을 수행하면 주의 대화 상자가 나타납니다.

문제 해결:

전원 상태를 포함한 vCLS VM의 상태는 VMware ESX Agent Manager 및 워크로드 제어부 서비스를 통해 관리됩니다. vCLS VM의 전원 켜기 오류가 발생하거나 vCLS VM의 쿼럼이 부족하여 클러스터에 대한 DRS의 첫 번째 인스턴스를 건너뛰면, 클러스터 요약 페이지에 오류 상태 문제를 해결하는 데 도움이 되는 기술 자료 문서 링크와 함께 배너가 나타납니다.

vCLS VM은 시스템 VM으로 취급되므로 해당 VM을 백업하거나 스냅샷을 생성할 필요가 없습니다. 이러한 VM의 상태는 vCenter Server 서비스에서 관리됩니다.

클러스터를 철회 모드로 전환

데이터스토어가 유지 보수 모드로 전환되고 데이터스토어가 vCLS VM을 호스팅하는 경우 vCLS VM을 새 위치로 수동으로 Storage vMotion 작업하거나 클러스터를 철회 모드로 전환해야 합니다.

이 작업에서는 클러스터를 철회 모드로 전환하는 방법을 설명합니다.

절차

- 1 vSphere Client에 로그인합니다.
- 2 vCLS를 비활성화해야 하는 클러스터로 이동합니다.
- 3 vCenter Server **구성** 탭으로 이동합니다.
- 4 **구성**에서 **일반**을 선택합니다.
- 5 기본 **시스템 관리** 옵션 또는 vCLS를 비활성화하는 **철회 모드**를 선택합니다.
- 6 **확인**을 클릭합니다.

결과

클러스터에 DRS가 활성화된 경우 작동이 중지되고 **클러스터 요약**에 추가 주의가 표시됩니다. DRS가 활성화되어 있더라도 vCLS가 철회 모드에서 제거되어 재구성될 때까지 DRS가 작동하지 않습니다.

vSphere HA는 호스트 장애 시나리오 동안 최적의 배치를 수행하지 않습니다. HA는 DRS에 따라 배치 권장 사항을 수행합니다. HA는 여전히 VM의 전원을 켤 수 있지만 이러한 VM의 전원이 덜 최적의 호스트에서 켜질 수 있습니다.

외부 vCLS의 암호 검색

vCLS VM에 로그인하기 위한 암호를 검색할 수 있습니다.

클러스터 서비스 상태를 보장하려면 vCLS VM에 액세스하지 마십시오. 이 문서는 vCLS VM에 대한 명시적 진단을 위한 것입니다.

절차

1 SSH를 사용하여 vCenter Server Appliance에 로그인합니다.

2 다음 Python 스크립트를 실행합니다.

```
/usr/lib/vmware-wcp/decrypt_clustervm_pw.py
```

3 암호에 대한 출력을 읽습니다.

```
pwd-script-output
Read key from file
Connected to PSQL
PWD: (password displayed here)
```

결과

검색된 암호를 사용하여 vCLS VM에 로그인할 수 있습니다.

vCLS VM 반선호도 정책

vSphere는 vCLS VM과 다른 워크로드 VM 그룹 사이에 반선호도를 지원합니다.

계산 정책은 vSphere DRS(Distributed Resource Scheduler)가 리소스 풀의 호스트에 VM을 배치하는 방식을 지정하는 방법을 제공합니다. vSphere 계산 정책 편집기를 사용하여 계산 정책을 생성하고 삭제합니다. 계산 정책을 생성하거나 삭제할 수 있지만 수정할 수는 없습니다. 정책 정의에 사용된 범주 태그를 삭제하면 정책도 삭제됩니다. vSphere **VM 요약** 페이지를 열면 VM에 적용되는 계산 정책과 각 정책의 규정 준수 상태를 볼 수 있습니다. vCLS VM 그룹에 대해 반선호도를 갖는 워크로드 VM 그룹에 대한 계산 정책을 생성할 수 있습니다. vCLS 반선호도 정책은 워크로드 VM 그룹에 대해 단일 사용자 표시 태그를 포함할 수 있으며 다른 vCLS VM 그룹은 내부적으로 인식됩니다.

vCLS VM 반선호도 정책 생성 또는 삭제

vCLS VM 반선호도 정책은 VM 범주와 vCLS 시스템 VM 간의 관계를 설명합니다.

vCLS VM 반선호도 정책은 vCLS VM과 애플리케이션 VM을 동일한 호스트에 배치하는 것을 억제합니다. 이러한 종류의 정책은 중요한 워크로드를 실행하는 가상 시스템과 vCLS VM을 동일한 호스트에서 실행하지 않으려는 경우에 유용할 수 있습니다. SAP HANA와 같은 중요한 워크로드를 실행하기 위한 몇 가지 모범 사례에는 전용 호스트가 필요합니다. 정책이 생성되면 배치 엔진은 정책 VM이 실행되고 있지 않은 호스트에 vCLS VM을 배치하려고 시도합니다.

vCLS VM 반선호도 정책 적용은 여러 가지 방법으로 영향을 받을 수 있습니다.

- 정책이 서로 다른 호스트의 여러 VM에 적용되고 vCLS VM을 배포하기에 충분한 호스트를 가질 수 없는 경우 vCLS VM은 정책 VM이 없는 호스트에 통합됩니다.
- 프로비저닝 작업에서 대상 호스트를 지정하는 경우 해당 규격은 정책을 위반하더라도 항상 적용됩니다. DRS는 이후 업데이트 적용 주기에 vCLS VM을 규정 준수 호스트로 이동하려고 시도합니다.

절차

- 1 vCLS VM 반선호도 정책에 포함하려는 각 VM 그룹에 대한 범주 및 태그를 생성합니다.
- 2 포함하려는 VM에 태그를 지정합니다.
- 3 vCLS VM 반선호도 정책을 생성합니다.
 - a vSphere에서 **정책 및 프로파일 > 계산 정책**을 클릭합니다.
 - b **추가**를 클릭하여 **새 계산 정책** 마법사를 엽니다.
 - c 정책 **이름**을 입력하고 **정책 유형** 드롭다운 컨트롤에서 **vCLS VM 반선호도**를 선택합니다.
정책 **이름**은 고유해야 합니다.
 - d 정책의 **설명**을 제공한 다음 **VM 태그**를 사용하여 정책이 적용되는 **범주 및 태그**를 선택합니다.
범주와 연결된 VM 태그가 여러 개가 아닌 경우 태그 **범주**를 선택하면 마법사에서 VM 태그가 채워집니다.
 - e **생성**을 클릭하여 정책을 생성합니다.
- 4 (선택 사항) 계산 정책을 삭제하려면 vSphere를 열고 **정책 및 프로파일 > 계산 정책**을 클릭하여 각 정책을 카드로 표시합니다. [삭제]를 클릭하여 정책을 삭제합니다.

vSphere DRS 클러스터 생성

16

클러스터는 ESXi 호스트와 공유 리소스 및 공유 관리 인터페이스를 사용하는 연결된 가상 시스템의 모음입니다. 클러스터 수준 리소스 관리의 이점을 얻으려면 먼저 클러스터를 생성하고 DRS를 활성화해야 합니다.

클러스터에서 vSphere FT(Fault Tolerance) 가상 시스템을 사용할 경우 DRS의 동작은 EVC(향상된 vMotion 호환성)의 활성화 여부에 따라 달라집니다.

표 16-1. vSphere FT 가상 시스템 및 EVC를 사용할 경우의 DRS 동작

EVC	DRS(로드 밸런싱)	DRS(초기 배치)
사용	사용(기본 및 보조 VM)	사용(기본 및 보조 VM)
사용 안 함	사용 안 함(기본 및 보조 VM)	사용 안 함(기본 VM) 완전히 자동화됨(보조 VM)

다음으로 아래 항목을 읽으십시오.

- 승인 제어 및 초기 배치
- 단일 가상 시스템 전원 켜기
- 그룹 전원 켜기
- 가상 시스템 마이그레이션
- DRS 마이그레이션 임계값
- 마이그레이션 권장 사항
- DRS 클러스터 요구 사항
- 공유 스토리지 요구 사항
- 공유 VMFS 볼륨 요구 사항
- 프로세서 호환성 요구 사항
- DRS 클러스터에 대한 vMotion 요구 사항
- 가상 플래시로 DRS 구성
- 클러스터 생성
- 클러스터 설정 편집

- 가상 시스템에 대한 사용자 지정 자동화 수준 설정
- DRS 비활성화
- 리소스 풀 트리 복원
- vSAN 확장된 클러스터의 DRS 인식
- vGPU DRS 배치
- VM용 DRS 오버헤드 메모리 관리

승인 제어 및 초기 배치

DRS 지원 클러스터에서 가상 시스템 그룹이나 단일 가상 시스템의 전원을 켜면 vCenter Server는 승인 제어를 수행하고, 클러스터에 가상 시스템을 지원하는 데 충분한 리소스가 있는지 확인합니다.

클러스터의 리소스가 부족하여 단일 가상 시스템의 전원 또는 그룹 전원 켜기 시도에 포함된 가상 시스템의 전원을 켤 수 없으면 메시지가 나타납니다. 그렇지 않으면 DRS는 각 가상 시스템에 대해 가상 시스템을 실행할 호스트에 대한 권장 사항을 생성하고 다음 작업 중 하나를 수행합니다.

- 배치 권장 사항을 자동으로 실행합니다.
- 배치 권장 사항을 표시하여 사용자가 수락 또는 재정의를 선택하도록 합니다.

참고 독립형 호스트나 비 DRS 클러스터의 가상 시스템에 초기 배치 권장 사항이 제공되지 않습니다. 전원이 켜질 때 현재 있는 호스트에 배치됩니다.

- DRS는 네트워크 대역폭을 고려합니다. 호스트 네트워크 포화 상태를 계산하여 DRS는 더 나은 배치 결정을 내릴 수 있습니다. 그러면 환경에 대한 보다 포괄적인 이해를 통해 가상 시스템의 성능 저하를 방지할 수 있습니다.

단일 가상 시스템 전원 켜기

DRS 클러스터에서 단일 가상 시스템의 전원을 켜면 초기 배치 권장 사항을 받아 볼 수 있습니다.

단일 가상 시스템의 전원을 켤 때 두 가지의 초기 배치 권장 사항이 제공됩니다.

- 단일 가상 시스템의 전원이 켜져 있으며 사전 요구 사항 단계가 필요하지 않습니다.
가상 시스템에 대해 상호 배타적인 초기 배치 권장 사항 목록이 제공됩니다. 하나만 선택할 수 있습니다.

- 단일 가상 시스템의 전원이 켜져 있지만 사전 요구 사항 작업이 필요합니다.

이러한 작업에는 대기 모드에 있는 호스트의 전원 켜기 또는 한 호스트에서 다른 호스트로 다른 가상 시스템 마이그레이션 등이 포함됩니다. 이 경우 제공되는 권장 사항은 여러 줄이며, 그 각각은 사전 요구 사항 작업을 나타냅니다. 사용자는 전체 권장 사항을 따르거나 가상 시스템 전원 켜기를 취소할 수 있습니다.

그룹 전원 켜기

동시에 여러 가상 시스템의 전원을 켤 수 있습니다(그룹 전원 켜기).

그룹 전원 켜기 시도에 선택된 가상 시스템이 동일한 DRS 클러스터에 있을 필요는 없습니다. 여러 클러스터에서 선택할 수 있지만 동일한 데이터 센터 내에 있는 시스템만 선택해야 합니다. 또한 비 DRS 클러스터 또는 독립 실행형 호스트에 있는 가상 시스템도 포함할 수 있습니다. 이러한 가상 시스템은 전원이 자동으로 켜지므로 초기 배치 권장 사항에 포함되지 않습니다.

그룹 전원 켜기 시도를 위한 초기 배치 권장 사항은 각 클러스터 기반으로 제공됩니다. 그룹 전원 켜기 시도를 위한 모든 배치 관련 작업이 자동 모드인 경우 초기 배치 권장 사항이 제공되지 않고 가상 시스템의 전원이 켜집니다. 하나 이상의 가상 시스템에 대한 배치 관련 작업이 수동 모드인 경우 모든 가상 시스템(자동 모드인 가상 시스템 포함)에 대한 전원 켜기가 수동입니다. 이러한 작업은 초기 배치 권장 사항에 포함됩니다.

전원을 켜 가상 시스템이 속한 각 DRS 클러스터에 대해 모든 사전 요구 사항을 포함하는 단일 권장 사항이 제공되거나 아무런 권장 사항도 제공되지 않습니다. 이러한 모든 클러스터 관련 권장 사항은 **전원 켜기 권장 사항** 탭 아래에 함께 표시됩니다.

자동이 아닌 그룹 전원 켜기를 시도하며 초기 배치 권장 사항이 적용되지 않는 가상 시스템(독립형 호스트 또는 비 DRS 클러스터에 있는 가상 시스템)이 포함되는 경우 vCenter Server에서 이러한 가상 시스템에 대한 전원 켜기를 자동으로 시도합니다. 이 전원 켜기가 성공적으로 수행되면 해당 가상 시스템이 **전원 켜기 시작됨** 탭 아래에 나열됩니다. 전원을 켜지 못한 가상 시스템은 모두 **전원 켜기 실패** 탭 아래에 나열됩니다.

예제: 그룹 전원 켜기

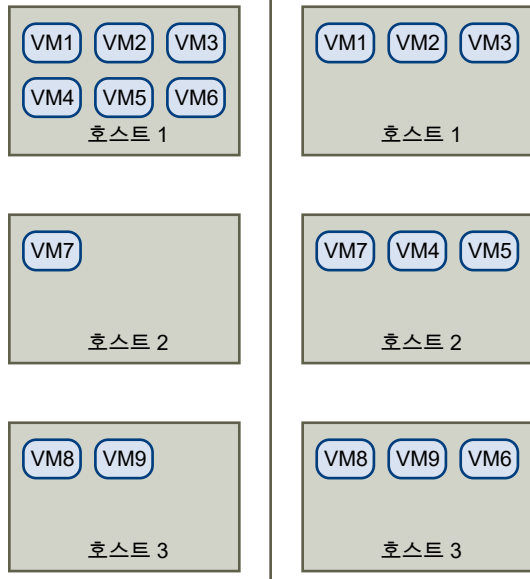
사용자는 그룹 전원 켜기를 시도할 세 가상 시스템을 동일한 데이터 센터에서 선택합니다. 처음 두 개의 가상 시스템(VM1 및 VM2)은 동일한 DRS 클러스터(Cluster1)에 있고 세 번째 가상 시스템(VM3)은 독립 실행형 호스트에 있습니다. VM1은 자동 모드이고 VM2는 수동 모드입니다. 이 시나리오에서 사용자에게는 VM1 및 VM2의 전원 켜기 작업으로 구성된, Cluster1에 대한 초기 배치 권장 사항이 **전원 켜기 권장 사항** 탭 아래에 제공됩니다. VM3의 전원 켜기가 자동으로 시도되고 성공하면 **전원 켜기 시작** 탭 아래에 나열됩니다. 이 시도가 실패하면 **전원 켜기 실패** 탭 아래에 나열됩니다.

가상 시스템 마이그레이션

클러스터의 로드가 균형을 이루도록 DRS가 초기 배치를 수행하더라도 가상 시스템 로드 및 리소스 가용성의 변화로 인해 클러스터는 불균형 상태가 될 수 있습니다. 이러한 불균형을 바로 잡기 위해 DRS는 마이그레이션 권장 사항을 생성합니다.

클러스터에서 DRS가 사용되도록 설정되면 로드가 보다 균일하게 분산되므로 이러한 불균형 정도를 줄일 수 있습니다. 예를 들어, 다음 그림의 왼쪽에 있는 세 개의 호스트는 불균형 상태입니다. 호스트 1, 호스트 2 및 호스트 3이 동일한 용량을 가지고 있고 모든 가상 시스템이 동일한 구성 및 로드를 가진다고 가정해 보겠습니다(예약이 설정된 경우 예약 포함). 하지만 호스트 1에 6개의 가상 시스템이 있기 때문에 호스트 2와 호스트 3에는 충분한 리소스가 있는 반면 호스트 1의 리소스는 과용될 수 있습니다. DRS는 호스트 1에서 호스트 2와 호스트 3으로 가상 시스템을 마이그레이션(또는 마이그레이션을 권장)합니다. 다이어그램의 오른쪽에는 이에 대한 결과로 호스트에 대해 적절히 로드 밸런싱된 구성이 나와 있습니다.

그림 16-1. 로드 밸런싱



클러스터가 불균형 상태가 되면 DRS는 기본 자동화 수준에 따라 권장 사항을 제공하거나 가상 시스템을 마이그레이션합니다.

- 클러스터나 관련된 가상 시스템이 수동으로 설정되었거나 부분적으로 자동화된 경우에는 vCenter Server가 리소스를 조정하기 위해 자동으로 작업을 수행하지 않습니다. 대신 요약 페이지에 마이그레이션 권장 사항이 있다고 표시되고 DRS 권장 사항 페이지에는 클러스터에서 리소스를 가장 효과적으로 사용하기 위해 수행할 변경 사항이 표시됩니다.
- 클러스터 및 관련된 가상 시스템이 모두 완전히 자동화된 경우에는 vCenter Server가 클러스터 리소스를 효과적으로 사용할 수 있도록 실행 중인 가상 시스템을 호스트 간에 마이그레이션합니다.

참고 자동 마이그레이션 설정이 사용되는 경우에도 사용자가 명시적으로 개별 가상 시스템을 마이그레이션할 수 있지만 vCenter Server가 이러한 가상 시스템을 다른 호스트로 이동하여 클러스터 리소스를 최적화할 수 있습니다.

기본적으로 자동화 수준은 전체 클러스터에 대해 지정됩니다. 필요하다면 개별 가상 시스템에 사용자 지정 자동화 수준을 지정할 수도 있습니다.

DRS 마이그레이션 임계값

DRS 마이그레이션 임계값을 사용하면 생성된 이후에 적용되는 권장 사항(권장 사항에 관련된 가상 시스템이 완전 자동화된 모드인 경우) 또는 표시되는 권장 사항(수동 모드인 경우)을 지정할 수 있습니다. 이 임계값은 VM 만족도 개선을 위해 DRS가 마이그레이션을 얼마나 적극적으로 권장하는지에 대한 척도입니다.

임계값 슬라이더를 이동하여 일반에서 적극적까지의 다섯 가지 설정 중 하나를 사용할 수 있습니다. 강도 설정이 높을수록 VM 만족도 개선을 위해 DRS가 더 자주 마이그레이션을 권장할 수 있습니다. [일반] 설정은 우선 순위가 1인 권장 사항(필수 권장 사항)만 생성합니다.

권장 사항에 우선 순위 수준이 지정되면 이 수준과 사용자가 설정한 마이그레이션 임계값이 비교됩니다. 우선 순위 수준이 임계값 설정보다 작거나 같으면 권장 사항이 적용되거나(관련 가상 시스템이 완전 자동화 모드인 경우), 사용자 확인을 위해 표시됩니다(수동 또는 부분 자동 모드인 경우).

DRS 점수

각 마이그레이션 권장 사항은 실행 효율성을 측정하는 VM 만족도 메트릭을 사용하여 계산됩니다. 이 메트릭은 vSphere Client에서 클러스터의 요약 탭에 DRS 점수로 표시됩니다. DRS 로드 밸런싱 권장 사항은 VM의 DRS 점수 향상을 시도합니다. 클러스터 DRS 점수는 클러스터에서 전원이 켜진 모든 VM의 VM DRS 점수에 대한 가중 평균입니다. 클러스터 DRS 점수는 측정기 구성 요소에 표시됩니다. 채워진 색상의 색상은 VM DRS 점수 히스토그램에서 해당하는 막대와 일치하는 값에 따라 달라집니다. 히스토그램의 막대는 해당 범위에 DRS 점수가 있는 VM의 백분율을 표시합니다. 클러스터의 [모니터] 탭을 선택하고 vSphere DRS를 선택하여 서버 측 정렬 및 필터링을 사용하는 목록을 볼 수 있습니다. 이 목록은 클러스터의 VM을 DRS 점수에 따라 오름차순으로 정렬하여 표시합니다.

마이그레이션 권장 사항

기본값인 수동 모드 또는 부분적으로 자동화된 모드를 사용하여 클러스터를 생성할 경우 vCenter Server는 DRS 권장 사항 페이지에 마이그레이션 권장 사항을 표시합니다.

시스템은 규칙을 적용하고 클러스터의 리소스를 조정하는 데 필요한 만큼의 권장 사항을 제공합니다. 각 권장 사항에는 이동할 가상 시스템, 현재(소스) 호스트 및 대상 호스트, 그리고 권장 이유가 포함됩니다. 다음 중 하나가 원인일 수 있습니다.

- 평균 CPU 로드 또는 예약을 조정합니다.
- 평균 메모리 로드 또는 예약을 조정합니다.
- 리소스 풀 예약을 충족시킵니다.
- 선호도 규칙을 충족시킵니다.
- 호스트가 유지 보수 모드 또는 대기 모드로 전환되는 중입니다.

참고 vSphere DPM(분산 전원 관리) 기능을 사용하는 경우에는 마이그레이션 권장 사항에 추가적으로 DRS 호스트 전원 상태 권장 사항이 제공됩니다.

DRS 클러스터 요구 사항

DRS 클러스터에 추가한 호스트에서 클러스터 기능을 사용하려면 호스트가 특정 요구 사항을 충족해야 합니다.

참고 vSphere DRS는 vSphere 클러스터 내에서 실행되는 워크로드의 상태를 유지 관리하는 데 필요한 vSphere의 중요 기능입니다. vSphere 7.0 업데이트 1부터 DRS는 vCLS VM의 가용성에 따라 달라집니다. 자세한 내용은 [장 15 vSphere 클러스터 서비스](#)의 내용을 참조하십시오.

공유 스토리지 요구 사항

DRS 클러스터에는 특정 공유 스토리지 요구 사항이 있습니다.

관리 호스트가 공유 스토리지를 사용하는지 확인합니다. 공유 스토리지는 대개 SAN에 있지만 NAS 공유 스토리지를 사용하여 구현할 수도 있습니다.

다른 공유 스토리지에 대한 정보는 "vSphere 스토리지" 설명서를 참조하십시오.

공유 VMFS 볼륨 요구 사항

DRS 클러스터에는 특정 공유 VMFS 볼륨 요구 사항이 있습니다.

공유된 VMFS 볼륨을 사용하도록 모든 관리 호스트를 구성합니다.

- 모든 가상 시스템의 디스크를 소스 및 대상 호스트에서 액세스할 수 있는 VMFS 볼륨에 배치합니다.
- VMFS 볼륨이 가상 시스템의 모든 가상 디스크를 저장할 수 있을 정도로 충분히 크지 확인합니다.
- 소스 및 대상 호스트의 모든 VMFS 볼륨이 볼륨 이름을 사용하며 모든 가상 시스템이 해당 볼륨 이름을 사용하여 가상 디스크를 지정하는지 확인합니다.

참고 가상 시스템 스왑 파일도 .vmdk 가상 디스크 파일과 마찬가지로 소스 및 대상 호스트에서 액세스할 수 있는 VMFS에 있어야 합니다. 모든 소스 및 대상 호스트가 ESX Server 3.5 이상이며 호스트-로컬 스왑을 사용하는 경우에는 이 요구 사항이 적용되지 않습니다. 이 경우에는 비공유 스토리지의 스왑 파일을 사용하는 vMotion이 지원됩니다. 스왑 파일은 기본적으로 VMFS에 배치되지만 관리자가 고급 가상 시스템 구성 옵션을 사용하여 파일 위치를 재정의할 수 있습니다.

프로세서 호환성 요구 사항

DRS 클러스터에는 특정한 프로세서 호환성 요구 사항이 있습니다.

DRS 기능이 제한되지 않도록 하려면 클러스터에 있는 소스 및 대상 호스트의 프로세서 호환성을 최대화해야 합니다.

vMotion은 가상 시스템의 실행 중인 아키텍처 상태를 기본 ESXi 호스트 간에 전송합니다. vMotion 호환성은 대상 호스트의 프로세서가 소스 호스트의 프로세서가 일시 중단되었던 위치에 있는 동등한 명령을 사용하여 실행을 재개할 수 있어야 한다는 의미입니다. 프로세서 클럭 속도와 캐시 크기는 다를 수 있지만 프로세서는 vMotion을 사용한 마이그레이션에 호환되는 동일한 벤더 클래스(Intel 대 AMD) 및 동일한 프로세서 제품군이어야 합니다.

프로세서 제품군은 프로세서 벤더가 정의합니다. 프로세서의 모델, 단계 수준 및 확장된 기능을 비교하면 같은 제품군 내에서 서로 다른 프로세서 버전을 구분할 수 있습니다.

때때로 프로세서 벤더가 동일한 프로세서 제품군 내에서 중대한 아키텍처 변경 사항을 도입할 경우가 있습니다(예: 64비트 확장 및 SSE3). VMware에서는 vMotion을 사용한 마이그레이션이 성공하리라고 보장할 수 없는 경우 이러한 예외를 식별합니다.

vCenter Server는 vMotion을 사용하여 마이그레이션된 가상 시스템이 프로세서 호환성 요구 사항을 충족할 수 있도록 도와주는 기능을 제공합니다. 이러한 기능에는 다음이 포함됩니다.

- EVC(향상된 vMotion 호환성) – EVC를 사용하여 클러스터에 있는 호스트의 vMotion 호환성이 보장되도록 도울 수 있습니다. EVC를 사용하면 호스트의 실제 CPU가 달라도 클러스터의 모든 호스트가 가상 시스템에 동일한 CPU 기능 세트를 제공합니다. 호환되지 않는 CPU로 인한 vMotion의 마이그레이션 오류를 방지합니다.

[클러스터 설정] 대화상자에서 EVC를 구성합니다. 클러스터에 있는 호스트가 특정 요구 사항을 충족해야 클러스터가 EVC를 사용할 수 있습니다. EVC 및 EVC 요구 사항에 대한 자세한 내용은 "vCenter Server 및 호스트 관리" 설명서를 참조하십시오.

- CPU 호환성 마스크 – vCenter Server는 가상 시스템에서 사용할 수 있는 CPU 기능을 대상 호스트의 CPU 기능과 비교하여 vMotion을 사용한 마이그레이션을 허용할 것인지 여부를 결정합니다. 개별 가상 시스템에 CPU 호환성 마스크를 적용함으로써 가상 시스템으로부터 특정 CPU 기능을 숨기고 잠재적으로 vMotion을 사용한 마이그레이션이 CPU와의 비호환으로 인해 실패하지 않도록 막을 수 있습니다.

DRS 클러스터에 대한 vMotion 요구 사항

DRS 클러스터에는 특정 vMotion 요구 사항이 있습니다.

DRS 마이그레이션 권장 사항을 사용할 수 있으려면 클러스터의 호스트가 vMotion 네트워크의 일부여야 합니다. 호스트가 vMotion 네트워크에 있지 않은 경우에도 DRS는 초기 배치 권장 사항을 생성할 수 있습니다.

vMotion에 대해 구성하려면 클러스터의 각 호스트가 다음 요구 사항을 충족해야 합니다.

- vMotion은 MSCS(Microsoft Cluster Service)를 사용하여 클러스터된 애플리케이션의 마이그레이션 또는 원시 디스크를 지원하지 않습니다.
- vMotion을 사용하려면 vMotion을 사용하도록 설정된 모든 관리 호스트 간에 전용 기가비트 이더넷 마이그레이션 네트워크가 필요합니다. 관리 호스트에서 vMotion을 사용하도록 설정한 경우 관리 호스트에 대한 고유 네트워크 ID 개체를 구성하고 이를 전용 마이그레이션 네트워크에 연결합니다.

가상 플래시로 DRS 구성

DRS는 가상 플래시 예약이 있는 가상 시스템을 관리할 수 있습니다.

가상 플래시 용량은 호스트에서 vSphere Client로 정기적으로 보고되는 통계로 나타납니다. DRS는 실행될 때마다 보고된 최신 용량 값을 사용합니다.

호스트당 하나의 가상 플래시 리소스를 구성할 수 있습니다. 즉, 가상 시스템의 전원이 켜져 있는 동안에는 DRS가 지정된 호스트의 여러 가상 플래시 리소스 중에서 선택할 필요가 없습니다.

DRS는 가상 시스템을 시작하는 데 사용 가능한 가상 플래시 용량이 충분한 호스트를 선택합니다. DRS가 가상 시스템의 가상 플래시 예약을 충족할 수 없으면 가상 시스템의 전원을 켤 수 없습니다. DRS는 가상 플래시 예약이 있는 전원이 켜진 가상 시스템을 현재 호스트와 함께 소프트 선호도를 갖는 것으로 간주합니다. DRS는 호스트를 유지 보수 모드로 설정하거나 초과 사용된 호스트의 로드를 줄이려는 경우 등 반드시 필요한 경우를 제외하고는 vMotion에 이러한 가상 시스템을 사용하는 것을 권장하지 않습니다.

클러스터 생성

클러스터는 호스트 그룹입니다. 클러스터에 호스트를 추가하면 호스트의 리소스가 클러스터 리소스의 일부가 됩니다. 클러스터는 클러스터 내 모든 호스트의 리소스를 관리합니다.

클러스터는 vSphere HA(High Availability) 및 vSphere DRS(Distributed Resource Scheduler) 솔루션을 사용하도록 설정합니다.

참고 vSphere DRS는 vSphere 클러스터 내에서 실행되는 워크로드의 상태를 유지 관리하는 데 필요한 vSphere의 중요 기능입니다. vSphere 7.0 업데이트 1부터 DRS는 vCLS VM의 가용성에 따라 달라집니다. 자세한 내용은 [장 15 vSphere 클러스터 서비스](#)의 내용을 참조하십시오.

사전 요구 사항

- 클러스터 개체를 생성할 수 있는 충분한 권한이 있는지 확인합니다.
- 인벤토리에 데이터 센터가 있는지 확인합니다.
- vSAN을 사용하려면 vSphere HA를 구성하기 전에 vSAN을 사용하도록 설정해야 합니다.

절차

- 1 vSphere Client에서 데이터 센터를 찾습니다.
- 2 데이터 센터를 마우스 오른쪽 버튼으로 클릭하고 **새 클러스터**를 선택합니다.
- 3 클러스터의 이름을 입력합니다.
- 4 DRS 및 vSphere HA 클러스터 기능을 선택합니다.

옵션	설명
이 클러스터에 DRS를 사용하려면	<ol style="list-style-type: none"> a DRS 설정 확인란을 선택합니다. b 자동화 수준과 마이그레이션 임계값을 선택합니다.
이 클러스터에 HA를 사용하려면	<ol style="list-style-type: none"> a vSphere HA 설정 확인란을 선택합니다. b 호스트 모니터링 및 승인 제어를 사용할지 여부를 선택합니다. c 승인 제어를 설정한 경우 정책을 지정합니다. d VM 모니터링 옵션을 선택합니다. e 가상 시스템 모니터링 감도를 지정합니다.

5 EVC(향상된 vMotion 호환성) 설정을 선택합니다.

EVC를 사용하면 호스트의 실제 CPU가 달라도 클러스터의 모든 호스트가 가상 시스템에 동일한 CPU 기능 세트를 제공합니다. 호환되지 않는 CPU로 인해 vMotion을 사용한 마이그레이션이 실패하는 것을 방지합니다.

6 확인을 클릭합니다.

결과

클러스터가 인벤토리에 추가됩니다.

다음에 수행할 작업

클러스터에 호스트와 리소스 풀을 추가합니다.

참고 **클러스터 요약** 페이지에서 vSphere 클러스터 서비스 상태를 표시하는 **클러스터 서비스**를 볼 수 있습니다.

클러스터 설정 편집

DRS 클러스터에 호스트를 추가하면 호스트의 리소스가 클러스터의 리소스의 일부가 됩니다. DRS 클러스터를 사용하면 이 리소스 집계 외에도 클러스터 전체의 리소스 풀을 지원하고 클러스터 수준 리소스 할당 정책을 적용할 수 있습니다.

다음과 같은 클러스터 수준 리소스 관리 기능도 사용할 수 있습니다.

로드 밸런싱

클러스터에 있는 모든 호스트와 가상 시스템의 CPU 및 메모리 리소스의 배포 및 사용량이 계속 모니터링됩니다. DRS는 클러스터의 리소스 풀 및 가상 시스템, 현재 요구량, 불균형 대상의 특성을 기준으로 이러한 메트릭과 이상적인 리소스 사용량을 비교합니다. 그런 다음 DRS는 그에 따라 권장 사항을 제공하거나 가상 시스템 마이그레이션을 수행합니다. **가상 시스템 마이그레이션**의 내용을 참조하십시오. 클러스터에 포함된 가상 시스템의 전원을 켜면 DRS가 가상 시스템을 적합한 호스트에 배치하거나 권장 사항을 제시하여 올바른 로드 밸런싱을 유지합니다. **승인 제어 및 초기 배치**의 내용을 참조하십시오.

전원 관리

vSphere DPM(Distributed Power Management) 기능이 사용되도록 설정되어 있으면 DRS는 최근 기간별 요구량을 포함하여 클러스터 및 호스트 수준의 용량과 클러스터의 가상 시스템을 실행하는 데 필요한 용량을 비교합니다. 그런 다음 DRS는 남은 용량이 충분하면 호스트를 대기 모드로 전환하거나 호스트를 대기 전원 모드로 전환하도록 권장합니다. 용량이 필요한 경우 DRS는 호스트의 전원을 켭니다. 호스트 전원 상태에 대한 권장 사항에 따라 가상 시스템을 호스트 간에 마이그레이션해야 할 수도 있습니다. **전원 리소스 관리**의 내용을 참조하십시오.

선호도 규칙

선호도 규칙을 할당하여 클러스터 내의 호스트에서 가상 시스템의 배치를 제어할 수 있습니다. **vSphere DRS와 선호도 규칙 사용**의 내용을 참조하십시오.

사전 요구 사항

클러스터는 특별한 라이선스 없이도 생성할 수 있지만 vSphere DRS 또는 vSphere HA에서 클러스터를 사용하도록 설정하려면 라이선스가 필요합니다.

참고 vSphere DRS는 vSphere 클러스터 내에서 실행되는 워크로드의 상태를 유지 관리하는 데 필요한 vSphere의 중요 기능입니다. vSphere 7.0 업데이트 1부터 DRS는 vCLS VM의 가용성에 따라 달라집니다. 자세한 내용은 [장 15 vSphere 클러스터 서비스](#)의 내용을 참조하십시오.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **서비스**를 클릭합니다.
- 3 **vSphere DRS**에서 **편집**을 클릭합니다.
- 4 **DRS 자동화**에서 DRS에 대한 기본 자동화 수준을 선택합니다.

자동화 수준	작업
수동	<ul style="list-style-type: none"> ■ 초기 배치: 권장 호스트가 표시됩니다. ■ 마이그레이션: 권장 사항이 표시됩니다.
부분적으로 자동화됨	<ul style="list-style-type: none"> ■ 초기 배치: 자동입니다. ■ 마이그레이션: 권장 사항이 표시됩니다.
완전히 자동화됨	<ul style="list-style-type: none"> ■ 초기 배치: 자동입니다. ■ 마이그레이션: 권장 사항이 자동으로 실행됩니다.

- 5 DRS의 **마이그레이션 임계값**을 설정합니다.
- 6 **Predictive DRS** 확인란을 선택합니다. 실시간 메트릭 외에, DRS는 vRealize Operations 서버에서 제공하는 예측 메트릭에 응답합니다. 이 기능을 지원하는 vRealize Operations 버전에서 **Predictive DRS**를 구성해야 합니다.
- 7 **가상 시스템 자동화** 확인란을 선택하여 개별 가상 시스템 자동화 수준을 사용하도록 설정합니다.
개별 가상 시스템에 대한 재정의는 [VM 재정의] 페이지에서 설정할 수 있습니다.
- 8 **추가 옵션**에서 확인란을 선택하여 기본 정책 중 하나를 적용합니다.

옵션	설명
VM 분포	가용성을 위해 호스트 전체에 더 많은 짝수 개의 가상 시스템을 분산하십시오. 이것은 DRS 로드 밸런싱에 대한 보조적인 옵션입니다.
로드 밸런싱을 위한 메모리 메트릭	활성 메모리가 아닌 가상 시스템에서 사용된 메모리를 기반으로 로드 밸런싱합니다. 이 설정은 호스트 메모리가 오버 커밋되지 않은 클러스터에 대해서만 권장됩니다. 참고 이 설정은 더 이상 지원되지 않으며 vCenter 7.0에서 표시되지 않습니다.
CPU 오버 커밋	클러스터의 CPU 오버 커밋을 제한합니다.
확장 가능한 공유	이 클러스터의 리소스 풀에 대해 확장 가능한 공유를 사용하도록 설정합니다.

- 9 **전원 관리**에서 자동화 수준을 선택합니다.
- 10 DPM을 사용하도록 설정한 경우 **DPM 임계값**을 설정합니다.
- 11 **확인**을 클릭합니다.

다음에 수행할 작업

참고 **클러스터 요약** 페이지에서 vSphere 클러스터 서비스 상태를 표시하는 **클러스터 서비스**를 볼 수 있습니다.

vSphere Client에서 DRS에 대한 메모리 활용률을 볼 수 있습니다. 자세한 내용은 다음을 참조하십시오.



(분산 리소스 스케줄러 메모리 활용률 보기)

가상 시스템에 대한 사용자 지정 자동화 수준 설정

DRS 클러스터를 생성한 후 개별 가상 시스템의 자동화 수준을 사용자 지정하여 클러스터의 기본 자동화 수준을 재정의할 수 있습니다.

예를 들어 완전히 자동화된 클러스터에서 특정한 가상 시스템을 **수동**으로 선택하거나 또는 수동 클러스터의 특정 가상 시스템에 **부분 자동화됨**을 선택할 수 있습니다.

가상 시스템이 **사용 안 함**으로 설정된 경우에는 vCenter Server가 가상 시스템을 마이그레이션하지 않거나 가상 시스템에 마이그레이션 권고사항을 제공하지 않습니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **서비스**를 클릭합니다.
- 3 [서비스]에서 **vSphere DRS**를 선택하고 **편집**을 클릭합니다. DRS 자동화를 확장합니다.
- 4 **개별 가상 시스템 자동화 수준 사용** 확인란을 선택합니다.
- 5 개별 가상 시스템 재정의의 임시로 비활성화하려면 **개별 가상 시스템 자동화 수준 사용** 확인란의 선택을 취소합니다.
확인란을 다시 선택하면 가상 시스템 설정이 복원됩니다.
- 6 클러스터에서 모든 vMotion 작업을 임시로 일시 중단하려면 **개별 가상 시스템 자동화 수준 사용** 확인란의 선택을 취소하고 클러스터를 수동 모드로 둡니다.
- 7 하나 이상의 가상 시스템을 선택합니다.
- 8 **자동화 수준** 열을 클릭하고 드롭다운 메뉴에서 자동화 수준을 선택합니다.

옵션	설명
수동	배치 및 마이그레이션 권장 사항이 표시되지만 권장 사항을 수동으로 적용하기 전까지 실행하지 않습니다.
완전히 자동화됨	배치와 마이그레이션 권장 사항이 자동으로 실행됩니다.

옵션	설명
부분적으로 자동화됨	초기 배치가 자동으로 수행됩니다. 마이그레이션 권장 사항이 표시되지만 실행하지 않습니다.
사용 안 함	vCenter Server는 가상 시스템을 마이그레이션하지 않고 마이그레이션 권고사항을 제공하지 않습니다.

9 확인을 클릭합니다.

결과

참고 vSphere vApp 및 vSphere Fault Tolerance 같은 다른 VMware 제품 또는 기능은 DRS 클러스터에서 가상 시스템의 자동화 수준을 재정의할 수 있습니다. 자세한 사항은 제품 설명서를 참조하십시오.

DRS 비활성화

클러스터에서 DRS를 해제할 수 있습니다.

DRS가 비활성화된 경우:

- DRS 선호도 규칙이 제거되지는 않지만 DRS가 재활성화될 때까지 적용되지 않습니다.
- 호스트 및 VM 그룹이 제거되지는 않지만 DRS가 재활성화될 때까지 적용되지 않습니다.
- 리소스 풀이 클러스터에서 영구적으로 제거됩니다. 리소스 풀의 손실을 방지하려면 로컬 시스템에 리소스 풀 트리의 스냅샷을 저장합니다. DRS를 활성화하면 스냅샷을 사용하여 리소스 풀을 복원할 수 있습니다.

참고 워크로드 관리가 활성화된 경우 DRS를 비활성화하지 마십시오. 그렇게 하면 WCP 서비스를 복구할 수 없게 됩니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 구성 탭을 클릭하고 **서비스**를 클릭합니다.
- 3 vSphere DRS에서 **편집**을 클릭합니다.
- 4 vSphere DRS **설정** 확인란의 선택을 취소합니다.
- 5 **확인**을 클릭하여 DRS를 해제합니다.
- 6 (선택 사항) 리소스 풀을 저장하는 옵션을 선택합니다.
 - 리소스 풀 트리 스냅샷을 로컬 시스템에 저장하려면 **예**를 클릭합니다.
 - 리소스 풀 트리 스냅샷을 저장하지 않고 DRS를 해제하려면 **아니요**를 클릭합니다.

결과

DRS가 해제됩니다.

참고 vSphere DRS는 vSphere 클러스터 내에서 실행되는 워크로드의 상태를 유지 관리하는 데 필요한 vSphere의 중요 기능입니다. vSphere 7.0 업데이트 1부터 DRS는 vCLS VM의 가용성에 따라 달라집니다. 자세한 내용은 [장 15 vSphere 클러스터 서비스](#)의 내용을 참조하십시오.

리소스 풀 트리 복원

이전에 저장된 리소스 풀 트리 스냅샷을 복원할 수 있습니다.

사전 요구 사항

- vSphere DRS를 설정해야 합니다.
- 스냅샷이 생성된 것과 동일한 클러스터에서만 스냅샷을 복원할 수 있습니다.
- 클러스터에 다른 리소스 풀이 없습니다.
- 백업 및 복원은 항상 동일한 버전의 vCenter 및 ESXi에서 수행해야 합니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 클러스터를 마우스 오른쪽 버튼으로 클릭하고 **리소스 풀 트리 복원**을 선택합니다.
- 3 **찾아보기**를 클릭하고 로컬 시스템에서 스냅샷 파일을 찾습니다.
- 4 **열기**를 클릭합니다.
- 5 **확인**을 클릭하여 리소스 풀 트리를 복원합니다.

vSAN 확장된 클러스터의 DRS 인식

vSAN 확장된 클러스터의 DRS 인식은 DRS를 사용하도록 설정된 확장된 클러스터에서 사용할 수 있습니다. vSAN 확장된 클러스터에는 읽기 지역성이 있으며, VM은 해당 위치에서 로컬 사이트의 데이터를 읽습니다. 원격 사이트에서 읽기를 가져오면 VM 성능에 영향을 줄 수 있습니다. 이제 DRS는 vSAN 확장된 클러스터의 DRS 인식을 통해 VM 읽기 지역성을 완전히 인식하기 때문에 읽기 지역성을 완전히 충족할 수 있는 사이트에 VM을 배치합니다. 이 기능은 자동이며, 구성 가능한 옵션은 없습니다. vSAN 확장된 클러스터의 DRS 인식은 기존 선호도 규칙과 함께 작동합니다. VMware Cloud on AWS에서도 작동합니다.

vSphere HA 및 vSphere DRS를 사용하는 vSAN 확장된 클러스터는 장애가 발생할 경우 장애 도메인 2개에 분산된 데이터 복사본 2개와 세 번째 장애 도메인의 감시 노드를 통해 복원력을 제공합니다. 2개의 활성 장애 도메인이 데이터 복제를 제공하기 때문에 양쪽 장애 도메인 모두에 최신 데이터 복사본이 있습니다.

vSAN 확장된 클러스터는 2개의 장애 도메인 내에서 워크로드를 이동하는 자동화된 방법을 제공합니다. 전체 사이트 장애가 발생할 경우, VM은 vSphere HA에 의해 보조 사이트에서 다시 시작됩니다. 이렇게 하면 중요한 프로덕션 워크로드에 다운타임이 없습니다. 기본 사이트가 다시 온라인 상태가 되면 DRS는 소프트 선호도 호스트를 사용하여 VM을 기본 사이트로 즉시 재조정합니다. 이러한 프로세스로 인해 VM 데이터 구성 요소가 아직 재구축되고 있는 동안에도 VM이 보조 사이트에서 읽고 쓰기를 수행하기 때문에 VM 성능이 저하될 수 있습니다.

vSphere 7.0 U2 이전 릴리스에서는 기본 사이트로 재동기화가 진행되는 동안 VM이 마이그레이션되지 않도록 DRS를 완전히 자동화된 모드에서 부분적으로 자동화된 모드로 변경하는 것이 좋습니다. 재동기화가 완료된 후에만 DRS를 완전히 자동화된 모드로 설정하십시오.

vSAN 확장된 클러스터의 DRS 인식을 통해 vSAN 확장된 클러스터의 장애 복구를 위해 완전히 자동화된 읽기 지역성 솔루션이 추가됩니다. 읽기 지역성 정보는 VM에 전체 액세스 권한이 있는 호스트를 나타내며 DRS는 vSAN 확장된 클러스터의 호스트에 VM을 배치할 때 이 정보를 사용합니다. DRS는 사이트 복구 단계 중에 vSAN 재동기화를 아직 진행 중일 때 VM이 기본 사이트로 장애 복구되는 것을 방지합니다. DRS는 데이터 구성 요소가 전체 읽기 지역성에 도달하면 VM을 기본 선호 사이트로 자동으로 마이그레이션합니다. 이렇게 하면 전체 사이트 장애가 발생할 경우 완전 자동 모드에서 DRS를 작동할 수 있습니다.

부분 사이트 장애가 발생할 경우 데이터 구성 요소 손실이 [허용되는 실패 수]보다 크거나 같기 때문에 VM의 읽기 지역성이 손실되면 vSphere DRS는 매우 높은 읽기 대역폭을 소비하는 VM을 파악하고 이것을 보조 사이트로 재조정하려고 시도합니다. 이렇게 하면 부분 사이트 장애가 발생해도 읽기 워크로드가 과도한 VM은 줄어들지 않습니다. 기본 사이트가 다시 온라인 상태가 되고 데이터 구성 요소의 재동기화가 완료되면 VM은 선호도가 있는 사이트로 다시 이동됩니다.

vGPU DRS 배치

DRS는 클러스터의 호스트 전반에 vGPU VM을 분산합니다.

DRS는 클러스터의 호스트 전반에 너비 우선 방식으로 vGPU VM을 분산합니다. VM에 대한 부분 vGPU 프로파일 할당에는 동종 프로파일 상호 제외 규칙이 적용될 수 있습니다.

- vGPU VM을 원하는 호스트로 수동으로 마이그레이션하여 사용되지 않은 물리적 GPU 용량을 확보합니다.
- 클러스터의 모든 vGPU VM에서 동일한 vGPU 프로파일 구성을 사용합니다.
- 호스트 "GPU 통합"을 사용하도록 설정합니다. 자세한 내용은 [호스트 그래픽 구성](#)을 참조하십시오.
- DRS 자동화가 활성 상태인 경우 클러스터 또는 VM을 **부분적으로 자동화된** 모드로 전환하는 것이 좋습니다. 자세한 내용은 [클러스터 설정 편집](#)을 참조하십시오.

VM용 DRS 오버헤드 메모리 관리

vSphere 8.0 U3 DRS에서는 재구성 중인 VM에 대한 오버헤드 메모리 관리가 향상되었습니다.

VMware vSphere에서 오버헤드 메모리는 ESXi가 VM(가상 시스템)을 관리하는 데 사용되는 메모리 양을 나타냅니다. 이 메모리는 ESXi가 기능을 수행하는 데 필요하며 VM에 할당된 게스트 메모리와는 별개입니다. 오버헤드 메모리의 양은 vCPU(가상 CPU) 수, VM에 할당된 메모리 양, VM의 구성 및 하드웨어 버전을 비롯한 여러 요인에 따라 달라집니다. vCPU가 많고 메모리 할당이 클수록 오버헤드 메모리 사용량이 높아질 수 있습니다.

vSphere에서 DRS는 ESXi 메모리 관리와 함께 작동하여 VM에 최적의 오버헤드 메모리 사용량을 보장합니다. DRS는 VM 오버헤드 메모리 제한을 설정하여 오버헤드 메모리를 관리하며, ESXi는 제한 내에서 오버헤드 메모리를 사용할 수 있습니다.

VMware vSphere에서 VM을 재구성하면 ESXi가 VM을 관리하는 데 필요한 오버헤드 메모리에 직접적인 영향을 미칠 수 있습니다. vCPU 수, 할당된 RAM 양을 수정하거나 네트워크 어댑터 또는 디스크 컨트롤러와 같은 가상 하드웨어를 추가하는 등 VM 구성을 변경하면 오버헤드 메모리 요구 사항이 변경될 수 있습니다. 예를 들어 VM 예약을 250GB에서 0GB로 재구성하려면 약 25MB의 오버헤드 메모리 사용량이 필요합니다. ESXi는 가상 페이지와 물리적 페이지 간의 페이지 테이블 매핑을 관리하기 위해 25MB의 추가 오버헤드 메모리를 할당합니다. vSphere는 이러한 변경 사항을 모니터링하고 관리합니다. 그러나 이전 vSphere 릴리스는 이러한 오버헤드 메모리 증가를 수용하도록 사용자 지정되지 않았습니다. 새 오버헤드 메모리 증가가 오버헤드 제한을 초과하면 재구성이 실패할 수 있습니다.

vSphere 8.0 U3에서 DRS는 재구성 전에 VM의 오버헤드 메모리 제한을 사전 예방적으로 업데이트합니다. DRS는 VM의 리소스 규격, IO 필터, 오버헤드 메모리에 영향을 미치는 기타 요소를 비롯한 다양한 요소를 확인합니다. DRS는 재구성 후 업데이트된 VM 규격으로 인해 예상되는 오버헤드 메모리 증가를 새 오버헤드 제한이 수용하도록 보장하여 VM 성능과 안정성을 최적화합니다.

DRS의 향상된 오버헤드 메모리 관리는 VM이 재구성되기 전에 오버헤드 메모리 제한을 지능적으로 관리하여 재구성 실패를 방지하는 데 도움이 되며 재구성 실패 위험을 크게 줄여줍니다. 이러한 사전 예방적 접근 방식은 보다 안정적인 경험을 보장합니다. VM 성능과 안정성을 최적화하여 특히 중요한 재구성 프로세스 중에 가상 환경이 중단 없이 효율적으로 실행될 수 있습니다. 이러한 향상된 기능은 기존 vSphere 환경에 원활하게 통합되는 동시에 성능과 안정성을 향상시킵니다.

ROBO Enterprise 라이선스를 사용한 vSphere DRS 유지 보수 모드 기능

17

VMware의 대규모 ROBO(Remote Office Branch Office) Enterprise 라이선스는 호스트가 유지 보수 모드로 전환될 때 자동 VM 제거를 지원합니다.

ROBO Enterprise 클러스터에서 DRS는 기본적으로 비활성화되기 때문에 DRS 구성을 변경할 수 없습니다. ROBO Enterprise 클러스터의 호스트가 유지 보수 모드로 전환되면 DRS에 의해 호스트에서 VM이 자동으로 제거됩니다. 호스트에서 VM을 제거하기 전에 DRS는 VM이 배치된 위치를 추적하기 위해 VM-호스트 선호도 매핑을 생성합니다. 호스트가 유지 보수 모드에 있으면 호스트에서 실행되던 VM이 호스트로 다시 마이그레이션됩니다. VM-호스트 선호도 매핑은 마이그레이션 후에 지워집니다.

다음으로 아래 항목을 읽으십시오.

- ROBO Enterprise 라이선스로 제공되는 DRS 유지 보수 모드의 제한 사항
- ROBO Enterprise 라이선스로 DRS 유지 보수 모드 사용
- ROBO Enterprise 라이선스로 DRS 유지 보수 모드 문제 해결

ROBO Enterprise 라이선스로 제공되는 DRS 유지 보수 모드의 제한 사항

ROBO Enterprise 라이선스로 제공되는 DRS 기능은 전체 DRS 기능이 아닙니다.

ROBO Enterprise 클러스터에서 유지 보수 모드를 시작하기 전에 다음과 같은 몇 가지 제한 사항을 알고 있어야 합니다. ROBO Enterprise 클러스터에는 DRS가 기본적으로 비활성화됩니다. DRS가 지원되는 라이선스에서 ROBO Enterprise 라이선스로 마이그레이션한 경우에는 시스템에 선호도 또는 반선호도 규칙이 있는 VM이 있을 수 있습니다. 선호도 또는 반선호도 규칙이 있는 VM을 비활성화하거나 삭제하지 않으면 ROBO Enterprise 유지 보수 모드 작업이 비활성화됩니다. DRS가 완전 자동화된 모드로 설정되어 있지 않으면 ROBO Enterprise 유지 보수 모드 작업이 비활성화됩니다. 호스트 유지 보수 워크플로를 통해 VM을 자동으로 제거하려면 DRS 자동화 수준을 완전히 자동화된 모드로 설정해야 합니다. VM이 완전히 자동화된 DRS 모드를 재정의하는 경우에는 VM을 수동으로 삭제해야 합니다.

ROBO Enterprise 라이선스로 DRS 유지 보수 모드 사용

vSphere는 ROBO Enterprise 라이선스로 제한된 DRS 유지 보수 모드 기능을 지원합니다.

사전 요구 사항

- 클러스터의 모든 호스트에 ROBO Enterprise 라이선스가 설치되어 있는지 확인합니다. 그렇지 않은 경우 라이선스를 설치해야 합니다.
- DRS 규칙이 구성되어 있고 활성화되어 있는지 확인합니다. 그렇다면 ROBO Enterprise 유지 보수 모드 작업을 사용하려면 비활성화하거나 삭제해야 합니다.

절차

- 1 DRS 유지 보수 모드를 ROBO Enterprise 라이선스로 사용하려면 클러스터의 각 호스트에 ROBO Enterprise 라이선스가 설치되어 있어야 합니다.
 - 라이선스가 설치되어 있지 않으면 2단계로 이동합니다.
 - 라이선스가 설치되어 있으면 3단계로 이동합니다.
- 2 ROBO Enterprise 라이선스를 설치합니다.
 - a vSphere Client에서 호스트를 찾습니다.
 - b 구성 탭에서 **라이선싱**을 선택합니다.
 - c **라이선스 할당**을 클릭합니다.
 - d ROBO Enterprise 라이선스 키를 입력하고 **확인**을 클릭합니다.클러스터의 모든 호스트에 대해 이 단계를 반복해야 합니다.
- 3 클러스터에서 호스트를 선택하고 마우스 오른쪽 버튼을 클릭하여 **유지 보수 모드 시작**을 선택하고 **확인**을 클릭합니다.

호스트의 VM이 자동으로 제거됩니다.

결과

호스트가 유지 보수 모드를 종료하면 VM은 자동으로 호스트에 다시 마이그레이션됩니다. 호스트가 원래 상태로 복원됩니다. 단, 호스트가 오버로드되면 DRS는 VM을 원래 호스트로 다시 마이그레이션할 수 없습니다. DRS가 호스트를 원래 상태로 복원하려고 하지만 호스트가 오버로드되도록 만들 수 없습니다.

다음에 수행할 작업

ROBO Enterprise 라이선스로 DRS 유지 보수 모드를 비활성화하려면, vpxd.cfg 파일을 편집하면 됩니다. vpxd.cfg 파일을 엽니다. <cluster> 옵션에서 <roboMMEabled>true</roboMMEabled>를 <roboMMEabled>>false</roboMMEabled>로 변경합니다. 이것은 런타임 구성이므로 구성을 업데이트한 후 vpxd를 다시 시작할 필요가 없습니다.

ROBO Enterprise 라이선스로 DRS 유지 보수 모드 문제 해결

ROBO Enterprise 클러스터에서 유지 보수 모드를 사용할 때 문제가 발생하는 경우 다음 사항을 고려하십시오.

ROBO Enterprise 클러스터에서 유지 보수 모드가 올바르게 작동하려면 다음을 수행합니다.

- 클러스터의 모든 호스트에 ROBO Enterprise 라이선스가 설치되어 있는지 확인합니다. 그렇지 않은 경우 라이선스를 설치해야 합니다.
- DRS 규칙이 구성되어 있고 활성화되어 있는지 확인합니다. 그렇다면 ROBO Enterprise 유지 보수 모드 작업을 사용하려면 비활성화하거나 삭제해야 합니다.
- 호환성 검사가 실패하면 다른 호스트가 VM과 호환되는지 확인합니다.

DRS 클러스터를 사용하여 vSphere로 리소스 관리

18

DRS 클러스터를 생성한 후 이를 사용자 지정하여 리소스를 관리하는 데 사용할 수 있습니다.

DRS 클러스터와 DRS 클러스터에 포함된 리소스를 사용자 지정하려면 선호도 규칙을 구성하고 호스트 및 가상 시스템을 추가하거나 제거합니다. 클러스터의 설정 및 리소스가 정의된 경우에는 유효한 클러스터로 유지되도록 해야 합니다. 올바른 DRS 클러스터를 사용하여 전원 리소스를 관리하고 vSphere HA와 상호 작용할 수도 있습니다.

참고 이 장에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- 클러스터에 호스트 추가
- 클러스터에 가상 시스템 추가
- 클러스터에서 가상 시스템 제거
- 클러스터에서 호스트 제거
- DRS 클러스터 유효성
- 전원 리소스 관리
- vSphere DRS와 선호도 규칙 사용
- VM-호스트 그룹 생성
- 가상 시스템 그룹 생성
- VM-VM 선호도 규칙 생성
- VM-VM 선호도 규칙 충돌
- VM 호스트 선호도 규칙 생성
- VM-호스트 선호도 규칙 사용
- vSphere DRS 없이 선호도 규칙 사용

클러스터에 호스트 추가

클러스터에 호스트를 추가하는 절차는 호스트가 같은 vCenter Server에서 관리하는 호스트(관리 호스트)인지 아니면 서버에서 관리하지 않는 호스트인지에 따라 다릅니다.

호스트를 추가하면 해당 호스트에 배포된 가상 시스템이 클러스터의 일부가 되고 DRS는 일부 가상 시스템을 클러스터의 다른 호스트에 마이그레이션하도록 권장할 수 있습니다.

참고 vSphere DRS는 vSphere 클러스터 내에서 실행되는 워크로드의 상태를 유지 관리하는 데 필요한 vSphere의 중요 기능입니다. vSphere 7.0 업데이트 1부터 DRS는 vCLS VM의 가용성에 따라 달라집니다. 자세한 내용은 [장 15 vSphere 클러스터 서비스](#)의 내용을 참조하십시오.

클러스터에 관리 호스트 추가

vCenter Server가 이미 관리하고 있는 독립 실행형 호스트를 DRS 클러스터에 추가하면 호스트의 리소스가 클러스터에 연결됩니다.

기존 가상 시스템과 리소스 풀을 클러스터의 루트 리소스 풀에 연결할지 아니면 리소스 풀 계층을 이식할지 선택할 수 있습니다.

참고 호스트에 하위 리소스 풀이나 가상 시스템이 없으면 호스트의 리소스가 클러스터에 추가되지만 최상위 리소스 풀이 있는 리소스 풀 계층은 생성되지 않습니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 호스트를 마우스 오른쪽 버튼으로 클릭하고 **다음으로 이동...**을 선택합니다.
- 3 클러스터를 선택합니다.
- 4 **확인**을 클릭하여 변경 사항을 적용합니다.
- 5 호스트의 가상 시스템과 리소스 풀에서 수행할 작업을 선택합니다.
 - **클러스터의 루트 리소스 풀에 이 호스트의 모든 가상 시스템을 추가합니다.**
vCenter Server가 호스트의 기존 리소스 풀을 모두 제거하고 호스트의 계층에 있는 모든 가상 시스템이 루트에 연결됩니다. 공유 할당은 리소스 풀에 상대적이기 때문에 리소스 풀 계층을 삭제하는 이 옵션을 선택한 후에는 가상 시스템의 공유를 수동으로 변경해야 할 수 있습니다.
 - **이 호스트의 가상 시스템 및 리소스 풀을 위한 리소스 풀을 생성합니다.**
vCenter Server가 클러스터의 직속 하위 리소스 풀이 되는 최상위 리소스 풀을 생성하고 호스트의 모든 하위 항목을 이 새 리소스 풀에 추가합니다. 새로 생성하는 최상위 리소스 풀의 이름을 지정할 수 있으며, 기본값은 <host_name>에서 이식됩니다.

결과

호스트가 클러스터에 추가됩니다.

클러스터에 관리되지 않는 호스트 추가

관리되지 않는 호스트를 클러스터에 추가할 수 있습니다. 이러한 호스트는 현재 클러스터와 동일한 vCenter Server 시스템에서 관리되고 있지 않으므로 vSphere Client에 표시되지 않습니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 클러스터를 마우스 오른쪽 버튼으로 클릭하고 **호스트 추가**를 선택합니다.
- 3 호스트 이름, 사용자 이름 및 암호를 입력하고 **다음**을 클릭합니다.
- 4 요약 정보를 확인한 후 **다음**을 클릭합니다.
- 5 기존 라이선스 키나 새 라이선스 키를 할당하고 **다음**을 클릭합니다.
- 6 (선택 사항) 원격 사용자가 호스트에 직접 로그인하지 못하도록 잠금 모드를 설정할 수 있습니다.
잠금 모드를 설정하지 않을 경우 나중에 호스트 설정에서 보안 프로파일을 편집하여 이 옵션을 구성할 수 있습니다.
- 7 호스트의 가상 시스템과 리소스 풀에서 수행할 작업을 선택합니다.
 - **클러스터의 루트 리소스 풀에 이 호스트의 모든 가상 시스템을 추가합니다.**
vCenter Server가 호스트의 기존 리소스 풀을 모두 제거하고 호스트의 계층에 있는 모든 가상 시스템이 루트에 연결됩니다. 공유 할당은 리소스 풀에 상대적이기 때문에 리소스 풀 계층을 삭제하는 이 옵션을 선택한 후에는 가상 시스템의 공유를 수동으로 변경해야 할 수 있습니다.
 - **이 호스트의 가상 시스템 및 리소스 풀을 위한 리소스 풀을 생성합니다.**
vCenter Server가 클러스터의 직속 하위 리소스 풀이 되는 최상위 리소스 풀을 생성하고 호스트의 모든 하위 항목을 이 새 리소스 풀에 추가합니다. 새로 생성하는 최상위 리소스 풀의 이름을 지정할 수 있으며, 기본값은 <host_name>에서 이식됩니다.
- 8 설정을 검토하고 **마침**을 클릭합니다.

결과

호스트가 클러스터에 추가됩니다.

클러스터에 가상 시스템 추가

클러스터에 가상 시스템을 추가하는 방법에는 여러 가지가 있습니다.

- 호스트를 클러스터에 추가하면 해당 호스트의 모든 가상 시스템이 클러스터에 추가됩니다.
- 가상 시스템을 생성하면 **새 가상 시스템** 마법사가 가상 시스템을 배치할 위치를 묻는 메시지를 표시합니다. 이 마법사에서 독립 실행형 호스트 또는 클러스터를 선택할 수 있으며 호스트나 클러스터 내의 리소스 풀을 선택할 수 있습니다.

- **가상 시스템 마이그레이션** 마법사를 사용하여 가상 시스템을 독립 실행형 호스트에서 클러스터로 마이그레이션하거나 클러스터에서 다른 클러스터로 마이그레이션할 수 있습니다. 이 마법사를 시작하려면 가상 시스템 이름을 마우스 오른쪽 버튼으로 클릭하고 **마이그레이션**을 선택합니다.

가상 시스템을 클러스터로 이동

가상 시스템을 클러스터로 이동할 수 있습니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
 - a 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - b **VM** 탭을 클릭합니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **다음으로 이동...**을 선택합니다.
- 3 클러스터를 선택합니다.
- 4 **확인**을 클릭합니다.

클러스터에서 가상 시스템 제거

클러스터에서 가상 시스템을 제거할 수 있습니다.

클러스터에서 가상 시스템을 제거할 수 있는 방법은 두 가지가 있습니다.

- 클러스터에서 호스트를 제거하면 다른 호스트로 마이그레이션하지 않은 전원이 꺼진 모든 가상 시스템도 제거됩니다. 유지 보수 모드에 있거나 연결이 끊어진 호스트만 제거할 수 있습니다. DRS 클러스터에서 호스트를 제거하면 클러스터가 오버 커밋되므로 노란색이 될 수 있습니다.
- **마이그레이션** 마법사를 사용하여 가상 시스템을 클러스터에서 독립형 호스트로 또는 클러스터에서 다른 클러스터로 마이그레이션할 수 있습니다. 이 마법사를 시작하려면 가상 시스템 이름을 마우스 오른쪽 버튼으로 클릭하고 **마이그레이션**을 선택합니다.

가상 시스템을 클러스터 외부로 이동

가상 시스템을 클러스터 밖으로 이동할 수 있습니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
 - a 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - b **VM** 탭을 클릭합니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **마이그레이션**을 선택합니다.
- 3 **데이터스토어 변경**을 선택하고 **다음**을 클릭합니다.
- 4 데이터스토어를 선택하고 **다음**을 클릭합니다.

5 마침을 클릭합니다.

가상 시스템이 DRS 클러스터 규칙 그룹의 구성원인 경우 vCenter Server에서 마이그레이션을 계속하도록 허용하기 전에 주의를 표시합니다. 주의를 종속 가상 시스템은 자동으로 마이그레이션되지 않음을 나타냅니다. 마이그레이션을 계속하려면 주의를 확인해야 합니다.

클러스터에서 호스트 제거

DRS 클러스터에서 호스트를 제거하면 리소스 풀 계층 구조와 가상 시스템에 영향이 있으며 잘못된 클러스터를 생성할 수 있습니다. 호스트를 제거하기 전에 영향을 받는 개체를 고려해야 합니다.

- 리소스 풀 계층 구조 호스트를 클러스터에 추가할 때 DRS 클러스터를 사용하고 호스트 리소스 풀을 이식하기로 결정한 경우에도 클러스터에서 호스트를 제거하면 호스트에는 루트 리소스 풀만 유지됩니다. 이러한 경우 클러스터에 계층 구조가 유지됩니다. 호스트별 리소스 풀 계층 구조를 생성할 수 있습니다.

참고 클러스터에서 호스트를 제거할 때 먼저 호스트를 유지 보수 모드로 전환해야 합니다. 클러스터에서 호스트를 제거하기 전에 호스트와의 연결을 끊으면 호스트에는 클러스터 계층 구조가 반영되는 리소스 풀이 유지됩니다.

- 가상 시스템 클러스터에서 호스트를 제거하려면 호스트가 유지 보수 모드에 있어야 하며, 호스트를 유지 보수 모드로 전환하려면 전원이 켜진 모든 가상 시스템을 해당 호스트 밖으로 마이그레이션해야 합니다. 호스트를 유지 보수 모드로 전환하도록 요청할 때 해당 호스트에서 전원이 꺼진 모든 가상 시스템을 클러스터의 다른 호스트로 마이그레이션할지 여부를 묻는 메시지가 나타납니다.
- 잘못된 클러스터 클러스터에서 호스트를 제거하면 클러스터에서 사용할 수 있는 리소스가 줄어듭니다. 클러스터에 있는 리소스가 클러스터의 모든 가상 시스템과 리소스 풀을 예약하는 데 충분하면 클러스터는 줄어든 크기의 리소스를 반영하도록 리소스 할당을 조정합니다. 클러스터에 있는 리소스가 모든 리소스 풀을 예약하는 데는 충분하지 않지만 모든 가상 시스템을 예약하는 데 충분하면 경보가 실행되며 클러스터가 노란색으로 표시됩니다. DRS는 계속해서 실행됩니다.

유지 보수 모드로 호스트 전환

추가 메모리를 설치하는 등의 서비스가 필요한 경우 호스트를 유지 보수 모드로 전환합니다. 호스트는 사용자 요청에 의해서만 유지 보수 모드로 전환하거나 유지 보수 모드를 마침니다.

유지 보수 모드로 전환되는 호스트에서 실행 중인 가상 시스템은 다른 호스트로 마이그레이션(수동으로 또는 DRS를 통해 자동으로)하거나 종료해야 합니다. 실행 중인 모든 가상 시스템이 다른 호스트로 마이그레이션되거나 전원이 꺼질 때까지 호스트는 **유지 보수 모드로 전환되고 있음** 상태에 있습니다. 가상 시스템의 전원을 켜거나 유지 보수 모드로 전환되는 호스트로 가상 시스템을 마이그레이션할 수 없습니다.

호스트에서 실행 중인 가상 시스템이 더 이상 없으면 호스트의 아이콘이 **유지 보수 중**임을 나타내도록 변경되며 호스트의 요약 패널이 새로운 상태를 표시합니다. 유지 보수 모드인 호스트에서는 가상 시스템을 배포하거나 가상 시스템의 전원을 켤 수 없습니다.

참고 DRS는 호스트가 요청된 모드로 전환된 후 vSphere HA 페일오버 수준을 위반할 경우 유지 보수 모드 또는 대기 모드로 전환되는 호스트의 가상 시스템을 마이그레이션하도록 권장하지 않습니다(또는 완전히 자동화된 모드로 수행함).

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 호스트를 마우스 오른쪽 버튼으로 클릭하고 **유지 보수 모드 > 유지 보수 모드 시작**을 선택합니다.
 - 호스트가 부분적으로 자동화된 DRS 클러스터 또는 수동 DRS 클러스터의 일부인 경우 **클러스터 > 모니터 > DRS > 권장 사항**으로 이동하여 **권장 사항 적용**을 클릭합니다.
 - 호스트가 자동화된 DRS 클러스터에 속해 있으면 해당 호스트가 유지 보수 모드로 전환될 때 가상 시스템이 다른 호스트로 마이그레이션됩니다.
- 3 해당되는 경우 **예**를 클릭합니다.

결과

유지 보수 모드 > 유지 보수 모드 종료를 선택하기 전까지 호스트는 유지 보수 모드에 있습니다.

클러스터에서 호스트 제거

클러스터에서 호스트를 제거할 수 있습니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 호스트를 마우스 오른쪽 버튼으로 클릭하고 **유지 보수 모드 > 유지 보수 모드 시작**을 선택합니다.
호스트가 유지 보수 모드에 있으면 호스트를 다른 인벤토리 위치(최상위 데이터 센터 또는 다른 클러스터)로 이동합니다.
- 3 호스트를 마우스 오른쪽 버튼으로 클릭하고 **다음으로 이동...**을 선택합니다.
- 4 호스트의 새 위치를 선택하고 **확인**을 클릭합니다.

결과

호스트를 이동하면 호스트의 리소스가 클러스터에서 제거됩니다. 호스트의 리소스 풀 계층 구조를 클러스터로 이식한 경우 해당 계층 구조는 클러스터에 남아 있습니다.

다음에 수행할 작업

클러스터에서 호스트를 제거한 후 다음 작업을 수행할 수 있습니다.

- vCenter Server에서 호스트를 제거합니다.

- vCenter Server에서 호스트를 독립 실행형 호스트로 실행합니다.
- 호스트를 다른 클러스터로 이동합니다.

대기 모드 사용

호스트 시스템이 대기 모드로 전환될 경우 전원이 꺼집니다.

보통 호스트는 vSphere DPM 기능에 의해 대기 모드로 전환되어 전원 사용을 최적화합니다. 호스트를 수동으로 대기 모드로 전환할 수 있습니다. 하지만 DRS는 다음 번에 실행될 때 사용자의 변경 사항을 실행 취소(또는 실행 취소를 권장)할 수 있습니다. 호스트를 강제로 전원이 꺼진 상태로 두려면 유지 보수 모드로 전환하고 전원을 끄십시오.

VM BestEffortRestart 정책 구성

이 정책은 유지 보수 모드 중 VM의 철수 동작을 제어합니다.

유지 보수 모드로 전환할 때는 호스트가 유지 보수에 대해 준비가 되어 있는지 확인해야 합니다. 호스트를 업그레이드하기 위해 이렇게 하는 경우가 많습니다. 업그레이드 수행 여부에 관계없이 실행 중인 모든 VM을 철수하여 호스트를 유지 보수 모드로 전환해야 합니다. 이렇게 하려면 전원이 켜진 VM을 처리하는 방법을 결정해야 합니다. 업그레이드를 수행하는 경우 호스트가 유지 보수 모드로 전환될 수 있는지 확인해야 합니다. vMotion을 수행할 수 없는 VM이 있으면 유지 보수 모드로 전환하기 전에 해당 VM의 전원을 끄면 됩니다.

계산 정책을 사용하면 유지 보수 모드로 전환할 때 VM의 전원을 꺼서 VM을 철수하는 이와 같은 상황을 자동화할 수 있습니다. 그런 다음, 원하는 모든 VM에 태그를 지정할 수 있으며 시스템이 유지 보수 모드로 전환될 때 이러한 VM의 전원이 꺼집니다.

BestEffortRestart 정책

유지 보수 모드로 전환되는 동안 **BestEffortRestart** 정책은 VM을 종료하려고 시도하며, 실패할 경우 VM의 전원이 꺼집니다. 정책은 호스트가 VM을 종료할 때까지 기다리는 대신 원래 호스트가 여전히 유지 보수 모드에 있는 동안 VM에 최적의 호스트를 찾습니다. **BestEffortRestart**는 유지 보수 모드로 전환되는 동안 최적의 호스트에서 VM의 전원이 켜질 작업을 생성하여 해당 VM이 가능한 한 빨리 전원이 켜진 상태가 되도록 합니다. 어떤 이유로든 VM의 전원을 켤 수 없는 경우 3분마다 실행되는 업데이트 적용 주기가 있습니다. 이 VM을 실행하기 위해 더 이상 유지 보수 모드를 종료할 필요가 없습니다.

호스트를 클러스터 외부로 이동하면서 VM은 클러스터에 남아 있고 호스트와 함께 사라지지 않도록 하려면 **BestEffortRestart**를 사용하면 됩니다. 이 경우 호스트가 클러스터 외부로 이동할 때 VM이 클러스터에 남아 있기를 원합니다. 따라서 유지 보수 모드로 전환하는 동안 VM의 전원을 꺼야 할 뿐만 아니라 이 VM에 가장 적합한 호스트(현재 호스트 제외)를 찾아서 전원을 켜야 합니다.

vGPU VM 및 기타 패스스루 디바이스에 **BestEffortRestart**를 사용하면 이점을 얻을 수 있습니다. vGPU VM에는 마이그레이션 비용이 많이 들고 기본 vMotion 제한 시간인 100초를 초과할 수 있는 대용량 메모리 프레임 버퍼가 있습니다. 이러한 vMotion은 시간 초과되어 vGPU VM이 원치 않는 상태로 남을 수 있습니다. 대신 유지 보수 모드로 전환되는 동안 vGPU VM의 전원을 끌 수 있습니다. 이상적으로 패스스루 VM(vGPU VM 포함)은 유지 보수 모드로 전환되는 동안 다른 호스트에서 전원을 켜야 합니다. 그러면 exitMM이 완료될 때까지 기다릴 필요가 없습니다.

이 정책이 제대로 작동하려면 DRS를 활성화해야 합니다. 호스트 또는 VM 수준에서 DRS가 비활성화되면 유지 보수 모드로 전환되는 동안 VM의 전원이 꺼집니다.

BestEffortRestart 정책이 삭제되면 정책이 삭제되는 시기에 따라 여러 가지 결과가 발생할 수 있습니다.

- VM에 대한 철수 작업 전에 정책이 제거되면 이 VM은 다른 VM과 동일하게 처리됩니다. **BestEffortRestart** 정책에 대한 연결이 없습니다.
- 철수 작업 후에 정책이 삭제되면 VM의 전원이 꺼질 수 있습니다. 이 VM은 더 이상 **BestEffortRestart** 정책과 연결되어 있지 않으므로 VM에 업데이트가 적용되지 않습니다.
- VM에 대한 철수 작업 전에 **BestEffortRestart** 정책에 연결된 태그가 삭제되면 이 VM은 다른 VM과 동일하게 처리됩니다. **BestEffortRestart** 정책에 대한 연결이 없습니다.

VM은 유지 보수 모드 이전과 동일한 상태여야 합니다. 유지 보수 모드 전에 VM의 전원이 켜져 있었다면 결국 VM의 전원이 켜지게 됩니다.

BestEffortRestart 정책이 성공하면 모든 **BestEffortRestart** 정책 VM의 전원이 꺼지고 최적의 호스트에서 전원이 켜집니다.

DRS 클러스터 유효성

vSphere Client에는 DRS 클러스터가 유효한지, 오버 커밋되었는지(노란색) 또는 유효하지 않은지(빨간색)가 표시됩니다.

DRS 클러스터가 오버 커밋되거나 유효하지 않은 이유는 몇 가지가 있습니다.

- 호스트에 장애가 발생할 경우 클러스터가 오버 커밋될 수 있습니다.
- vCenter Server를 사용할 수 없는 상태에서 vSphere Client를 사용하여 가상 시스템 전원을 켜면 클러스터가 유효하지 않은 상태가 됩니다.
- 가상 시스템에서 페일오버가 진행 중일 때 사용자가 상위 리소스 풀의 예약을 줄이면 클러스터가 유효하지 않은 상태가 됩니다.
- vCenter Server를 사용할 수 없는 상태에서 vSphere Client를 사용하여 호스트나 가상 시스템을 변경하면 해당 변경 내용이 적용됩니다. 그러나 vCenter Server를 다시 사용할 수 있게 되면 클러스터가 클러스터 요구 사항을 더 이상 충족하지 않기 때문에 빨간색이나 노란색으로 바뀔 수 있습니다.

클러스터 유효성 시나리오에서는 다음과 같은 용어를 이해하고 있어야 합니다.

예약

리소스 풀에 사용하도록 사용자가 입력한 고정된 할당량입니다.

사용된 예약

각 하위 리소스 풀에 대한 예약 합계 또는 사용된 예약(둘 중 더 큰 값)이며 반복적으로 추가됩니다.

미예약

음수가 아닌 이 숫자는 리소스 풀 유형에 따라 다릅니다.

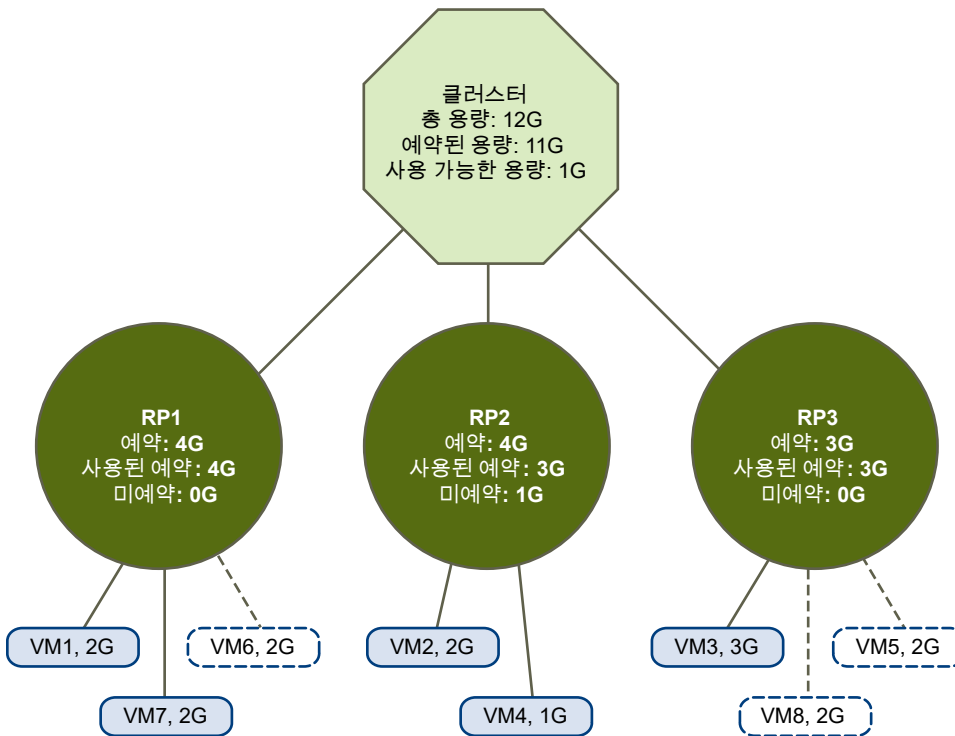
- 확장 불가능한 리소스 풀: 예약에서 사용된 예약을 뺀 값입니다.
- 확장 가능한 리소스 풀: (예약에서 사용된 예약을 뺀 값)에 상위 리소스 풀에서 빌려올 수 있는 예약되지 않은 리소스를 더한 값입니다.

유효한 DRS 클러스터

유효한 클러스터는 모든 예약을 충족하고 실행 중인 모든 가상 시스템을 지원할 수 있을 만큼 충분한 리소스를 가지고 있어야 합니다.

다음 그림에서는 고정 리소스 풀을 가진 유효한 클러스터의 예와 해당 CPU 및 메모리 리소스가 계산되는 방식을 보여 줍니다.

그림 18-1. 고정 리소스 풀을 가진 유효한 클러스터



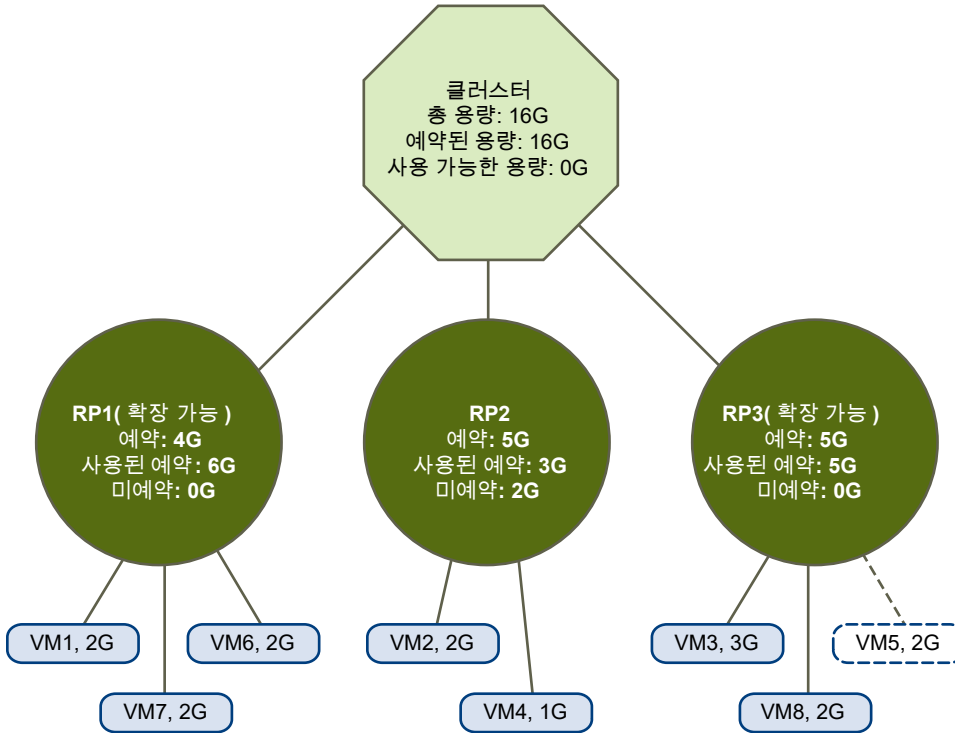
클러스터는 다음과 같은 특징을 가지고 있습니다.

- 총 리소스가 12GHz인 클러스터가 있습니다.
- 유형이 고정(확장 가능한 예약이 선택되지 않음)인 세 개의 리소스 풀이 있습니다.
- 세 개의 리소스 풀이 결합된 총 예약은 11GHz(4+4+3GHz)입니다. 클러스터의 **예약된 용량** 필드에 총계가 표시됩니다.
- RP1은 4GHz의 예약으로 생성되었습니다. 두 개의 가상 시스템입니다. 각각이 2GHz인 (VM1 및 VM7)의 전원이 켜졌습니다(**사용된 예약**: 4GHz). 추가 가상 시스템의 전원을 켤 수 있는 리소스가 남아 있지 않습니다. VM6은 전원이 켜져 있지 않습니다. 이 가상 시스템은 예약을 사용하지 않습니다.

- RP2는 4GHz를 예약하여 생성되었습니다. 1GHz 및 2GHz인 두 가상 시스템의 전원이 켜졌습니다(**사용된 예약: 3GHz**). 1GHz가 예약되지 않은 채 남아 있습니다.
- RP3은 3GHz의 예약으로 생성되었습니다. 3GHz인 가상 시스템의 전원이 켜졌습니다. 추가 가상 시스템의 전원을 켜는 데 사용할 수 있는 리소스가 없습니다.

다음 그림에서는 **확장 가능** 예약 유형을 사용하는 리소스 풀(RP1 및 RP3)을 가진 유효한 클러스터의 예를 보여 줍니다.

그림 18-2. 확장 가능한 리소스 풀을 가진 유효한 클러스터



유효한 클러스터는 다음과 같이 구성될 수 있습니다.

- 총 리소스가 16GHz인 클러스터가 있습니다.
- RP1과 RP3은 **확장 가능** 유형이며, RP2는 고정 유형입니다.
- 세 개의 리소스 풀이 결합하여 사용하는 총 예약은 16GHz(RP1에 6GHz, RP2에 5GHz, RP3에 5GHz)입니다. 최상위 수준에 있는 클러스터의 **예약된 용량**으로 16GHz가 표시됩니다.
- RP1은 4GHz의 예약으로 생성되었습니다. 각각이 2GHz인 세 가상 시스템의 전원이 켜졌습니다. 이들 가상 시스템 중 두 개(예: VM1 및 VM7)는 RP1의 예약을 사용할 수 있고 세 번째 가상 시스템(VM6)은 클러스터 리소스 풀의 예약을 사용할 수 있습니다. (이 리소스 풀 유형이 **고정**이면 추가 가상 시스템의 전원을 켤 수 없습니다.)
- RP2는 5GHz를 예약하여 생성되었습니다. 1GHz 및 2GHz인 두 가상 시스템의 전원이 켜졌습니다(**사용된 예약: 3GHz**). 2GHz가 예약되지 않은 채 남아 있습니다.

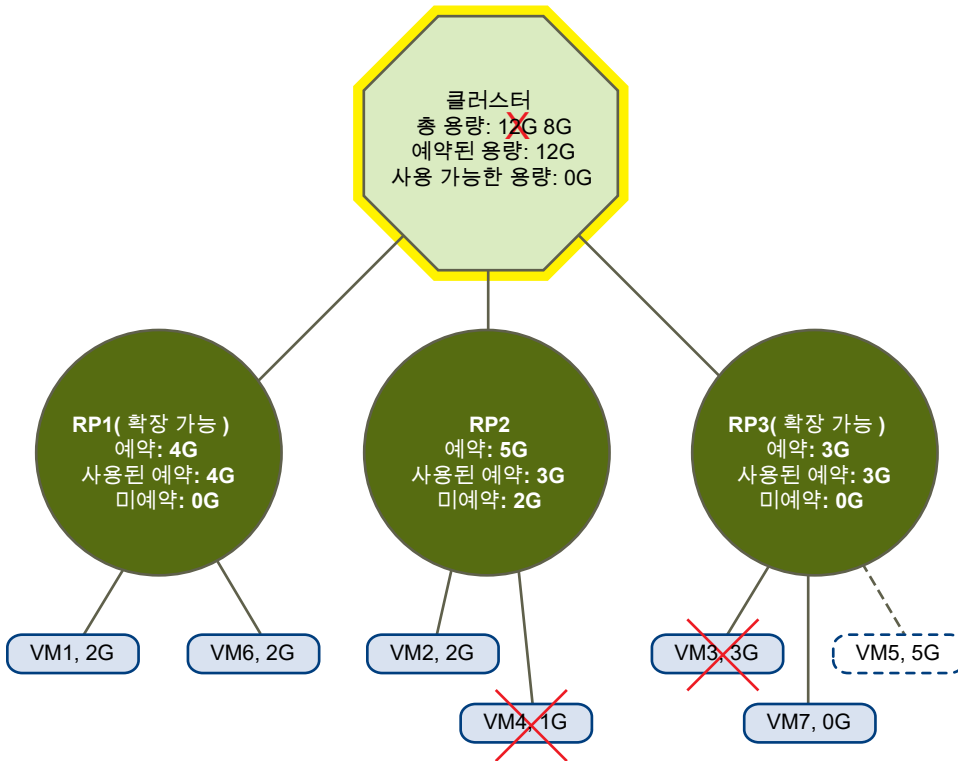
RP3은 5GHz의 예약으로 생성되었습니다. 3GHz 및 2GHz인 두 가상 시스템의 전원이 켜졌습니다. 이 리소스 풀 유형이 **확장 가능**이라도 상위에 있는 여분의 리소스가 이미 RP1에 의해 사용되었기 때문에 추가로 2GHz 가상 시스템의 전원을 켤 수 없습니다.

오버 커밋된 DRS 클러스터

리소스 풀 및 가상 시스템의 트리가 내부적으로는 일관성이 있지만 하위 리소스 풀에서 예약한 모든 리소스를 지원할 수 있는 클러스터 용량이 부족하면 클러스터가 오버 커밋됩니다(노란색).

호스트를 사용할 수 없게 되면 해당되는 모든 가상 시스템을 사용할 수 없게 되므로 실행 중인 모든 가상 시스템을 지원하기에 리소스는 항상 충분합니다. 클러스터 용량이 갑자기 줄어들 경우 예를 들어, 클러스터에 있는 호스트를 사용할 수 없게 되는 경우에는 일반적으로 클러스터가 노란색으로 바뀝니다. 클러스터가 노란색으로 바뀌지 않게 하려면 적절한 추가 클러스터 리소스를 남겨 두는 것이 좋습니다.

그림 18-3. 노란색 클러스터



이 예의 경우:

- 각 호스트마다 4GHz씩, 세 호스트에서 총 12GHz의 리소스를 얻을 수 있는 클러스터가 있습니다.
- 총 12GHz가 예약된 리소스 풀 3개가 있습니다.
- 세 개의 리소스 풀이 결합되어 사용한 총 예약은 12GHz(4+5+3GHz)입니다. 이것은 클러스터에 **예약된 용량**으로 표시됩니다.
- 4GHz 호스트 중 하나를 사용할 수 없게 되면 총 리소스는 8GHz로 줄어듭니다.
- 이와 동시에 해당 호스트에서 실행 중이던 VM4(1GHz)와 VM3(3GHz)은 더 이상 실행되지 않습니다.

- 클러스터는 이제 총 6GHz가 필요한 가상 시스템을 실행합니다. 클러스터에서는 8GHz를 사용할 수 있으므로 가상 시스템 요구 사항을 충족하기에 충분합니다.

리소스 풀 예약인 12GHz가 더 이상 충족될 수 없으므로 클러스터가 노란색으로 표시됩니다.

잘못된 DRS 클러스터

트리가 더 이상 내부적으로 일관되지 않은 경우, 즉 리소스 제약 조건이 지켜지지 않는 경우 DRS가 설정된 클러스터는 잘못된 상태(빨간색)로 표시됩니다.

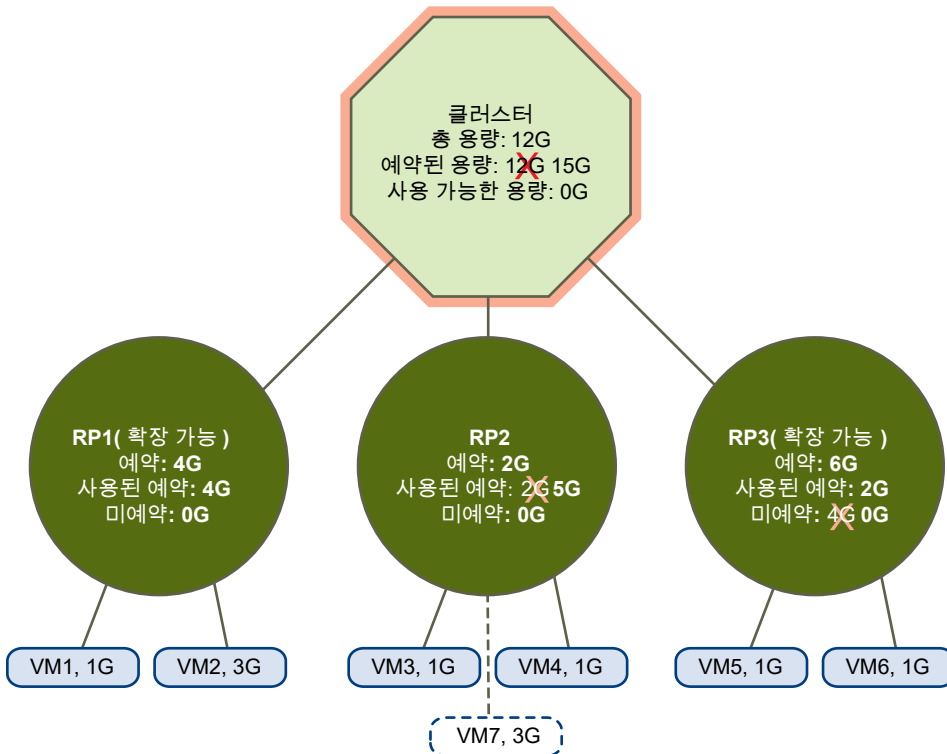
클러스터의 전체 리소스 양은 클러스터의 상태(빨간색)에 영향을 주지 않습니다. 하위 수준에 불일치가 있으면 루트 수준에 충분한 리소스가 있더라도 클러스터가 빨간색으로 표시될 수 있습니다.

하나 이상의 가상 시스템의 전원을 끄고 리소스가 충분한 트리 부분으로 가상 시스템을 이동하거나 빨간색 부분의 리소스 풀 설정을 편집하여 빨간색 DRS 클러스터 문제를 해결할 수 있습니다. 리소스를 추가하는 방법은 일반적으로 노란색 상태일 경우에만 도움이 됩니다.

가상 시스템 페일오버 도중에 리소스 풀을 재구성하는 경우에도 클러스터가 빨간색으로 바뀔 수 있습니다. 페일오버 중인 가상 시스템은 연결이 끊기고 상위 리소스 풀이 사용하는 예약으로 계수되지 않습니다. 페일오버가 완료되기 전에 상위 리소스 풀의 예약을 줄일 수 있습니다. 페일오버가 완료되면 가상 시스템 리소스가 다시 상위 리소스 풀로 반환됩니다. 풀의 사용량이 새 예약보다 커지면 클러스터가 빨간색으로 바뀝니다.

사용자가 지원되지 않는 방법을 통해 리소스 풀 2에서 3GHz의 예약으로 가상 시스템을 시작할 수 있다면 다음 그림에서처럼 클러스터가 빨간색으로 바뀝니다.

그림 18-4. 빨간색 클러스터



전원 리소스 관리

vSphere DPM(Distributed Power Management) 기능을 사용하면 클러스터 리소스 활용률을 기반으로 호스트의 전원을 켜고 꺼서 DRS 클러스터의 전력 소비가 줄어듭니다.

vSphere DPM은 클러스터에 있는 모든 가상 시스템의 메모리 및 CPU 리소스에 대한 누적 요구량을 모니터링하고 이를 클러스터에 있는 모든 호스트의 전체 가용 리소스 용량과 비교합니다. 남은 용량이 충분한 경우 vSphere DPM은 하나 이상의 호스트를 대기 모드로 전환하고 해당 가상 시스템을 다른 호스트로 마이그레이션한 후 이들 호스트의 전원을 끕니다. 이와 반대로 용량이 부족하면 DRS는 호스트의 대기 모드를 종료하고(전원을 켜고) vMotion을 사용하여 가상 시스템을 이들 호스트로 마이그레이션합니다. 이러한 계산을 할 때 vSphere DPM은 현재의 요구만 고려하는 것이 아니라 사용자가 지정한 가상 시스템 리소스 예약도 고려합니다.

DRS 클러스터를 생성할 때 **예측 메트릭**을 사용하도록 설정하면 선택하는 롤링 예측 창에 따라 DPM이 제안을 미리 발행합니다.

참고 ESXi 호스트는 vCenter Server에서 관리하는 클러스터에서 실행되지 않는 경우 대기 모드에서 자동으로 벗어날 수 없습니다.

vSphere DPM은 세 가지 전원 관리 프로토콜 중 하나를 사용하여 호스트의 대기 모드를 종료할 수 있습니다. 이 세 가지 프로토콜은 IPMI(Intelligent Platform Management Interface), Hewlett-Packard iLO(Integrated Lights-Out), WOL(Wake-On-LAN)입니다. 각 프로토콜에는 고유의 하드웨어 지원 및 구성이 필요합니다. 호스트가 이러한 프로토콜을 지원하지 않으면 vSphere DPM이 호스트를 대기 모드로 전환할 수 없습니다. 호스트가 여러 프로토콜을 지원하면 IPMI, iLO, WOL 순으로 사용됩니다.

참고 먼저 호스트의 전원을 켜지 않고 대기 모드의 호스트 연결을 끊거나 DRS 클러스터 외부로 호스트를 이동하면 안 됩니다. 이 경우 vCenter Server가 호스트의 전원을 다시 켜지 못합니다.

vSphere DPM의 IPMI 또는 iLO 설정 구성

IPMI는 하드웨어 수준의 규격이고 Hewlett-Packard iLO는 내장형 서버 관리 기술입니다. 각각은 컴퓨터를 원격으로 모니터링하고 제어하기 위한 인터페이스를 설명하고 제공합니다.

각 호스트에서 다음 절차를 수행해야 합니다.

사전 요구 사항

IPMI와 iLO 모두 하드웨어 BMC(베이스보드 관리 컨트롤러)를 사용하여 하드웨어 제어 기능에 액세스하는 게이트웨이를 제공하고 원격 시스템에서 직렬 또는 LAN 연결을 사용하여 해당 인터페이스에 액세스할 수 있도록 합니다. BMC는 호스트 자체의 전원이 꺼져 있어도 전원이 켜져 있습니다. 올바르게 설정할 경우 BMC는 원격 전원 켜기 명령에 응답할 수 있습니다.

IPMI 또는 iLO를 wake 프로토콜로 사용하려면 BMC를 반드시 구성해야 합니다. BMC를 구성하는 단계는 모델별로 다릅니다. 자세한 내용은 벤더 설명서를 참조하십시오. IPMI의 경우에는 BMC LAN 채널이 항상 사용 가능하고 운영자 권한의 명령을 허용하도록 구성되었는지 확인해야 합니다. 일부 IPMI 시스템에서는 "IPMI over LAN"을 사용하도록 설정하면 그 내용을 BIOS에도 구성하고 특정 IPMI 계정을 지정해야 합니다.

IPMI만 사용하는 vSphere DPM에서는 MD5 및 일반 텍스트 기반 인증을 지원하지만 MD2 기반 인증은 지원하지 않습니다. vCenter Server에서는 MD5가 지원되고 운영자 역할에 사용하도록 설정되었다고 호스트의 BMC가 보고하는 경우에 MD5를 사용합니다. 그렇지 않으면 일반 텍스트 기반 인증이 지원되고 사용 가능하다고 BMC가 보고하면 일반 텍스트 인증이 사용됩니다. MD5나 일반 텍스트 인증 둘 모두 사용하도록 설정되어 있지 않으면 호스트에서 IPMI를 사용할 수 없고 vCenter Server에서 Wake-on-LAN을 사용하려고 시도합니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 **구성** 탭을 클릭합니다.
- 3 **시스템**에서 **전원 관리**를 클릭합니다.
- 4 **편집**을 클릭합니다.
- 5 다음 정보를 입력합니다.
 - BMC 계정의 사용자 이름과 암호. 사용자 이름에는 원격으로 호스트 전원을 켤 수 있는 기능이 필요합니다.
 - BMC에 연결된 NIC의 IP 주소(호스트의 IP 주소와 달라야 함). IP 주소는 리스 기간에 제한이 없는 정적 주소나 DHCP 주소여야 합니다.
 - BMC에 연결된 NIC의 MAC 주소
- 6 **확인**을 클릭합니다.

vSphere DPM을 위한 Wake-on-LAN 테스트

VMware 지침에 따라 WOL(Wake-on-LAN)을 구성하고 성공적으로 테스트한 경우 vSphere DPM 기능에 WOL을 전면적으로 사용할 수 있습니다. 클러스터에 대해 처음으로 vSphere DPM을 활성화하거나 vSphere DPM을 사용하고 있는 클러스터에 호스트를 추가하기 전에 다음 단계를 수행해야 합니다.

사전 요구 사항

WOL을 테스트하기 전에 클러스터가 사전 요구 사항을 충족하는지 확인하십시오.

- 클러스터에 버전 ESX 3.5(또는 ESX 3i 버전 3.5) 이상의 호스트가 적어도 두 개는 있어야 합니다.
- 각 호스트의 vMotion 네트워킹 링크가 올바르게 작동하고 있어야 합니다. 또한 vMotion 네트워크는 라우터로 분리된 다중 서브넷이 아닌 단일 IP 서브넷이어야 합니다.
- 각 호스트의 vMotion NIC가 WOL을 지원해야 합니다. WOL 지원을 확인하려면 먼저 vSphere Client의 인벤토리 패널에서 호스트를 선택하고 **구성** 탭을 선택한 다음 **네트워킹**을 클릭하여 VMkernel 포트에 해당하는 물리적 네트워크 어댑터의 이름을 확인합니다. 이름을 확인했으면 **네트워크 어댑터**를 클릭하고 이 네트워크 어댑터에 해당하는 항목을 찾습니다. 관련 어댑터의 **WOL(Wake On Lan) 지원** 열에 "예"가 표시되어 있어야 합니다.
- 호스트에서 각 NIC의 WOL 호환성 상태를 표시하려면 vSphere Client의 인벤토리 패널에서 호스트를 선택하고 **구성** 탭을 선택한 다음 **네트워크 어댑터**를 클릭합니다. NIC의 **WOL(Wake On Lan) 지원** 열에 "예"가 표시되어 있어야 합니다.

- WOL을 지원하는 각 vMotion NIC가 연결되어 있는 스위치 포트는 고정 속도(예: 1000Mb/s)가 아니라, 링크 속도를 자동 협상하도록 설정되어 있어야 합니다. 대부분의 NIC는 호스트 전원이 꺼졌을 때 100Mb/s 이하로 전환될 수 있는 경우에만 WOL을 지원합니다.

이러한 사전 요구 사항을 확인했으면 WOL을 사용하여 vSphere DPM을 지원할 각 ESXi 호스트를 테스트합니다. 이러한 호스트를 테스트할 때 클러스터에 vSphere DPM 기능이 비활성화되었는지 확인하십시오.

경고 Wake 프로토콜로 WOL을 사용하는 vSphere DPM 클러스터에 추가하려는 모든 호스트를 테스트하고 테스트가 실패한 호스트는 전원 관리를 사용하지 않도록 비활성화해야 합니다. 그렇지 않으면 vSphere DPM이 호스트 전원을 꺼서 이후에 전원이 다시 켜지지 않게 됩니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 호스트를 마우스 오른쪽 버튼으로 클릭하고 **전원 > 대기 모드 시작**을 선택합니다.
이 작업은 호스트 전원을 끕니다.
- 3 호스트를 마우스 오른쪽 버튼으로 클릭하고 **전원 > 전원 켜기**를 선택하여 호스트의 대기 모드를 종료해 봅니다.
- 4 호스트 전원이 다시 켜지는지 여부를 확인합니다.
- 5 대기 모드가 종료되지 않는 호스트의 경우 다음 단계를 수행합니다.
 - a vSphere Client에서 호스트를 선택하고 **구성** 탭을 선택합니다.
 - b **하드웨어 > 전원 관리** 아래에서 **편집**을 클릭하여 전원 관리 정책을 조정합니다.
이렇게 하면 vSphere DPM이 해당 호스트를 전원 끄기 대상으로 간주하지 않습니다.

DRS 클러스터에 vSphere DPM 활성화

각 호스트에 사용하는 Wake 프로토콜에 필요한 구성 또는 테스트 단계를 수행한 후 vSphere DPM을 사용하도록 설정할 수 있습니다.

전원 관리 자동화 수준, 임계값 및 호스트 수준 재정의의 구성할 수 있습니다. 이러한 설정은 클러스터 설정 대화상자의 **전원 관리** 아래에서 구성합니다.

작업 스케줄링: 클러스터 전원 설정 변경 마법사를 사용하여 클러스터에 대해 DPM을 활성화 및 비활성화하는 스케줄링된 작업을 생성할 수도 있습니다.

참고 DRS 클러스터의 호스트에 USB 디바이스가 연결되어 있으면 해당 호스트에 대해 DPM을 비활성화하십시오. 그렇지 않으면 DPM에서 호스트 및 서버 전원을 꺼서 USB 디바이스와 해당 디바이스를 사용 중이던 가상 시스템 간의 연결을 끊을 수 있습니다.

자동화 수준

vSphere DPM에서 생성하는 호스트 전원 상태 및 마이그레이션 권장 사항이 자동으로 실행되는지 여부는 해당 기능에 대해 선택한 전원 관리 자동화 수준에 따라 결정됩니다.

자동화 수준은 클러스터의 설정 대화상자에 있는 **전원 관리**에서 구성됩니다.

참고 전원 관리 자동화 수준은 DRS 자동화 수준과는 다릅니다.

표 18-1. 전원 관리 자동화 수준

옵션	설명
꺼짐	기능이 비활성화되고 권장 사항이 제공되지 않습니다.
수동	호스트 전원 작업 및 관련 가상 시스템 마이그레이션 권장 사항이 생성되지만 자동으로 실행되지는 않습니다.
자동	호스트 전원 작업은 관련된 가상 시스템 마이그레이션을 모두 자동으로 실행할 수 있는 경우에 자동으로 실행됩니다.

vSphere DPM 임계값

vSphere DPM 기능으로 생성되는 전원 상태(호스트 전원 켜짐/꺼짐) 권장 사항에는 우선 순위 1 권장 사항부터 우선 순위 5 권장 사항까지의 우선 순위가 할당됩니다.

이 우선 순위 등급은 DRS 클러스터에서 발견된 초과 또는 미달 활용도 정도와 의도된 호스트 전원 상태 변경으로 기대되는 향상에 기반하여 지정됩니다. 우선 순위 1 권장 사항은 필수이며 우선 순위 5 권장 사항은 적용 시 약간의 향상만 가져옵니다.

해당 임계값은 클러스터의 [설정] 대화상자의 **전원 관리**에서 구성합니다. vSphere DPM 임계값 슬라이더를 한 수준 오른쪽으로 이동하면 자동으로 실행되거나 수동 실행 권장 사항으로 나타나는 권장 사항 집합에서 한 수준 더 낮은 우선 순위를 포함할 수 있습니다. 보수적인 설정에서 vSphere DPM은 우선 순위 1 권장 사항만 생성하고 오른쪽으로 한 수준 이동할 때 우선 순위 2 이상만 생성하는 식으로 진행되다 적극적 수준까지 이동하면 우선 순위 5 권장 사항 이상, 즉 모든 권장 사항을 생성합니다.

참고 DRS 임계값과 vSphere DPM 임계값은 기본적으로 독립적입니다. 두 임계값이 각각 제공하는 마이그레이션 및 호스트 전원 상태 권장 사항의 강도를 서로 다르게 지정할 수 있습니다.

호스트 수준 재정의

DRS 클러스터에서 vSphere DPM을 활성화하면 기본적으로 클러스터의 모든 호스트는 vSphere DPM 자동화 수준을 상속합니다.

클러스터 설정 대화상자의 [호스트 옵션] 페이지를 선택하고 해당 **전원 관리** 설정을 클릭하여 개별 호스트에 대해 이 기본값을 재정의할 수 있습니다. 이 설정을 다음 옵션으로 변경할 수 있습니다.

- 사용 안 함
- 수동
- 자동

참고 호스트의 전원 관리 설정이 대기 모드 종료 테스트 실패로 인해 [사용 안 함]으로 설정된 경우에는 변경하지 마십시오.

vSphere DPM을 활성화하고 실행한 후 클러스터 설정 대화상자의 [호스트 옵션] 페이지와 각 클러스터의 **호스트** 탭에 표시된 각 호스트의 **마지막 대기 종료 시간** 정보를 보고 제대로 작동하는지 확인할 수 있습니다. 이 필드에는 타임 스탬프와 vCenter Server가 호스트를 대기 모드에서 해제하려고 마지막으로 시도했을 때 성공 또는 실패했는지 여부를 표시합니다. 그러한 시도가 없었으면 필드에 [없음]이 표시됩니다.

참고 마지막 대기 종료 시간 텍스트 상자의 시간은 vCenter Server 이벤트 로그에서 파생됩니다. 이 로그가 지워지면 시간이 [없음]으로 재설정됩니다.

vSphere DPM 모니터링

vCenter Server의 이벤트 기반 경보를 사용하여 vSphere DPM을 모니터링할 수 있습니다.

vSphere DPM을 사용할 때 겪게 되는 가장 심각한 잠재적 오류는 DRS 클러스터에서 호스트 용량을 필요로 할 때 호스트의 대기 모드를 종료하지 못하는 경우입니다. vCenter Server의 미리 구성된 **대기 종료 오류** 경보를 사용하여 인스턴스에서 이 오류가 언제 발생하는지 모니터링할 수 있습니다. vSphere DPM이 호스트의 대기 모드를 종료할 수 없는 경우(vCenter Server 이벤트 `DrsExitStandbyModeFailedEvent`) 이 경보를 구성하여 관리자에게 경고 e-메일을 보내거나 SNMP 트랩을 사용하여 알림을 보낼 수 있습니다. 기본적으로 이 경보는 vCenter Server가 해당 호스트에 성공적으로 연결될 수 있게 되면 지워집니다.

vSphere DPM 작업을 모니터링하기 위해 다음 vCenter Server 이벤트에 대한 경보를 생성할 수도 있습니다.

표 18-2. vCenter Server 이벤트

이벤트 유형	이벤트 이름
대기 모드로 전환하고 있음(호스트의 전원을 끄려고 함)	<code>DrsEnteringStandbyModeEvent</code>
성공적으로 대기 모드로 전환됨(호스트 전원 끄기 성공)	<code>DrsEnteredStandbyModeEvent</code>
대기 모드를 종료하고 있음(호스트의 전원을 켜려고 함)	<code>DrsExitingStandbyModeEvent</code>
성공적으로 대기 모드를 종료함(전원 켜기 성공)	<code>DrsExitedStandbyModeEvent</code>

경보 생성 및 편집에 대한 자세한 내용은 "vSphere 모니터링 및 성능" 설명서를 참조하십시오.

vCenter Server가 아닌 다른 모니터링 소프트웨어를 사용하는 경우 물리적 호스트의 전원이 예상치 않게 꺼지면 소프트웨어가 경보를 트리거하게 되어 vSphere DPM이 호스트를 대기 모드로 만들 때 잘못된 경보가 생성되는 상황이 발생할 수 있습니다. 이러한 경보를 받지 않으려면 벤더와 상의하여 vCenter Server와 통합된 모니터링 소프트웨어 버전을 배포하도록 합니다. vCenter Server는 vSphere 4.x부터 vSphere DPM을 기본적으로 인식하고 이러한 잘못된 경보를 트리거하지 않기 때문에 vCenter Server 자체를 모니터링 솔루션으로 사용할 수도 있습니다.

VM 정적 전원

vSphere 8.0 U1에는 VM 정적 전력을 예측하는 새로운 전원 메트릭이 도입되었습니다.

이 전원 메트릭은 vCenter의 [vCenter 성능 모니터] 페이지의 **고급** 창에서 볼 수 있습니다. 이 전원 메트릭은 VM 구성의 기능, 특히 vCPU 수 및 메모리 크기와 실행 중인 호스트의 구성에 해당하는 반 고정 값입니다. VM 정적 전원은 ESXi 호스트와 유사한 하드웨어에서 실행되는 구성이 동일한 물리적 시스템의 요구 사항에 따라 필요한 최소 전력을 예측합니다. 한 호스트가 다른 호스트보다 전력 효율이 더 높은 경우 VM이 두 개의 서로 다른 호스트에서 서로 다른 정적 전원 값을 보고할 수 있습니다. 이는 동적 워크로드 배치 전략으로 전체 전력을 최적화하는 데 유용하며 워크로드의 가상화로 인한 에너지 절감을 예측하는 데 도움이 될 수 있습니다.

vSphere DRS와 선호도 규칙 사용

선호도 규칙을 사용하여 클러스터 내의 호스트에서 가상 시스템의 배치를 제어할 수 있습니다.

참고 이 항목에서는 DRS와 선호도 규칙을 사용하는 방법에 대해 설명합니다. DRS 없이 선호도 규칙을 사용할 수도 있습니다.

두 가지 유형의 규칙을 생성할 수 있습니다.

- 가상 시스템 그룹과 호스트 그룹 간에 선호도나 반선호도를 지정하기 위해 사용되었습니다. 선호도 규칙은 선택된 가상 시스템 DRS 그룹의 멤버가 특정 호스트 DRS 그룹 멤버에서 작동할 수 있거나 작동해야 한다고 지정합니다. 반선호도 규칙은 선택된 가상 시스템 DRS 그룹의 멤버가 특정 호스트 DRS 그룹 멤버에서 작동할 수 없다고 지정합니다.

이 규칙 유형의 생성 및 사용에 대한 자세한 내용은 [vSphere DRS의 VM-호스트 선호도 규칙](#) 항목을 참조하십시오.

- 개별 가상 시스템 간에 선호도나 반선호도를 지정하기 위해 사용되었습니다. 선호도를 지정하는 규칙은 예를 들어 성능 원인으로 인해 DRS가 동일한 호스트에 지정한 가상 시스템을 유지하도록 합니다. 반선호도 규칙을 사용하면 DRS가 지정한 가상 시스템을 따로 유지하기 때문에 한 호스트에서 문제가 발생하더라도 양쪽 가상 시스템이 모두 손실되지는 않습니다.

이 규칙 유형의 생성 및 사용에 대한 자세한 내용은 [vSphere DRS의 VM-VM 선호도 규칙](#) 항목을 참조하십시오.

선호도 규칙을 추가 또는 편집할 때 클러스터의 현재 상태가 규칙 위반 상태에 있으면 시스템은 계속 작동하며 위반 사항을 수정하려고 시도합니다. 수동 DRS 클러스터 및 부분적으로 자동화된 DRS 클러스터의 경우 규칙 준수 및 로드 밸런싱을 기반으로 한 마이그레이션 권장 사항이 승인을 위해 제시됩니다. 규칙을 따를 필요는 없지만 규칙이 준수될 때까지 해당 권장 사항이 남아 있습니다.

사용 중인 선호도 규칙을 위반했으며 DRS에서 수정할 수 없는 것인지 확인하려면 클러스터의 **DRS** 탭을 선택하고 **장애**를 클릭합니다. 현재 위반된 규칙이 있는 경우 이 페이지에 해당 장애가 표시됩니다. 장애 내용을 읽고 DRS가 특정 규칙을 충족할 수 없는 이유를 확인합니다. 규칙 위반 시에는 로그 이벤트도 생성됩니다.

참고 VM-VM 및 VM-호스트 선호도 규칙은 개별 호스트의 CPU 선호도 규칙과 다릅니다.

vSphere DRS의 VM-VM 선호도 규칙

DRS를 사용한 VM-VM 선호도 규칙은 선택한 개별 가상 시스템을 같은 호스트에서 실행할지 아니면 별도의 호스트에서 유지할지를 지정합니다. 이 규칙 유형은 선택한 개별 가상 시스템 간에 선호도 또는 반선호도를 생성하는데 사용됩니다.

참고 이 항목에서는 DRS와 선호도 규칙을 사용하는 방법에 대해 설명합니다. DRS 없이 선호도 규칙을 사용할 수도 있습니다.

선호도 규칙이 생성되면 DRS는 지정된 가상 시스템을 동일한 호스트에 함께 유지하려고 시도합니다. 예를 들어 성능상의 이유로 이 작업을 수행할 수 있습니다.

반선호도 규칙을 사용할 경우 DRS는 지정된 가상 시스템을 서로 다른 호스트에 유지하려고 시도합니다. 특정 가상 시스템이 항상 다른 물리적 호스트에 있도록 보장하려는 경우 이러한 규칙을 사용할 수 있습니다. 이 경우 하나의 호스트에서 문제가 발생하더라도 모든 가상 시스템이 위험에 처하게 되는 것은 아닙니다.

vSphere DRS의 VM-호스트 선호도 규칙

VM-호스트 선호도 규칙은 선택한 가상 시스템 DRS 그룹의 구성원이 특정 호스트 DRS 그룹의 구성원에서 실행될 수 있는지 여부를 지정합니다.

참고 이 항목에서는 DRS에 대한 선호도 규칙을 사용하는 방법에 대해 설명합니다. DRS 없이 선호도 규칙을 사용할 수도 있습니다.

개별 가상 시스템 간의 선호도(또는 반선호도)를 지정하는 VM-VM 선호도 규칙과 달리 VM-호스트 선호도 규칙은 가상 시스템 그룹과 호스트 그룹 간의 선호도 관계를 지정합니다. '필수' 규칙("필수"로 지정)과 '기본 설정' 규칙("해야 함"으로 지정)이 있습니다.

VM-호스트 선호도 규칙에는 다음 구성 요소가 포함됩니다.

- 가상 시스템 DRS 그룹 1개.
- 호스트 DRS 그룹 1개.
- 규칙이 요구 사항인지("필수") 또는 기본 설정인지("해야 함") 및 선호도("실행 중") 또는 반선호도("에서 실행되지 않음")인지를 지정합니다.

VM-호스트 선호도 규칙은 클러스터 기반이므로 규칙에 포함된 가상 시스템과 호스트는 모두 동일한 클러스터에 상주해야 합니다. 가상 시스템이 클러스터에서 제거되면 나중에 클러스터로 반환되더라도 DRS 그룹과의 연관성을 잃게 됩니다.

VM-호스트 그룹 생성

VM-호스트 선호도 규칙은 가상 시스템 그룹과 호스트 그룹 간에 선호도(또는 반선호도) 관계를 설정합니다. 연결 규칙을 만들기 전에 해당 그룹을 모두 만들어야 합니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.

- 2 구성 탭을 클릭합니다.
- 3 구성에서 **VM/호스트 그룹**을 선택하고 **추가**를 클릭합니다.
- 4 **VM/호스트 그룹 생성** 대화상자에서 그룹의 이름을 입력합니다.
- 5 **유형** 드롭다운 상자에서 **호스트 그룹**을 선택하고 **추가**를 클릭합니다.
- 6 추가할 호스트 옆의 확인란을 클릭합니다. 원하는 호스트가 모두 추가될 때까지 이 프로세스를 계속합니다.
- 7 **확인**을 클릭합니다.

다음에 수행할 작업

이 호스트 그룹을 사용하여 적절한 가상 시스템 그룹과 선호도(또는 반선호도) 관계를 설정하는 VM-호스트 선호도 규칙을 생성할 수 있습니다.

가상 시스템 그룹 생성

선호도 규칙은 그룹 간에 선호도(또는 반선호도) 관계를 설정합니다. 그룹을 연결하는 규칙을 생성하려면 먼저 그룹을 생성해야 합니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 구성 탭을 클릭합니다.
- 3 구성에서 **VM/호스트 그룹**을 선택하고 **추가**를 클릭합니다.
- 4 **VM/호스트 그룹 생성** 대화상자에서 그룹의 이름을 입력합니다.
- 5 **유형** 드롭다운 상자에서 **VM 그룹**을 선택하고 **추가**를 클릭합니다.
- 6 추가할 가상 시스템 옆의 확인란을 클릭합니다. 원하는 모든 가상 시스템이 추가될 때까지 이 프로세스를 계속합니다.
- 7 **확인**을 클릭합니다.

VM-VM 선호도 규칙 생성

VM-VM 선호도 규칙을 생성하여 선택된 개별 가상 시스템을 같은 호스트에서 실행할지 별도 호스트에서 유지할지 지정할 수 있습니다.

참고 vSphere HA 페일오버 호스트 지정 승인 제어 정책을 사용하고 여러 페일오버 호스트를 지정하는 경우 VM-VM 선호도 규칙이 지원되지 않습니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 구성 탭을 클릭합니다.

- 3 구성에서 **VM/호스트 규칙**을 클릭합니다.
- 4 **추가**를 클릭합니다.
- 5 **VM/호스트 규칙 생성** 대화상자에서 규칙 이름을 입력합니다.
- 6 **유형** 드롭다운 메뉴에서 **가상 시스템을 함께 보관** 또는 **가상 시스템을 별도로 보관** 중 하나를 선택합니다.
- 7 **추가**를 클릭합니다.
- 8 규칙을 적용할 가상 시스템을 두 개 이상 선택하고 **확인**을 클릭합니다.
- 9 **확인**을 클릭합니다.

VM-VM 선호도 규칙 충돌

여러 VM-VM 선호도 규칙을 생성하고 사용할 수 있지만 이로 인해 규칙이 서로 충돌하는 상황이 발생할 수 있습니다.

두 개의 VM-VM 선호도 규칙이 충돌하는 경우 둘 다 활성화할 수는 없습니다. 예를 들어 한 규칙은 두 가상 시스템을 함께 유지하고 다른 규칙은 동일한 두 가상 시스템을 따로 유지하는 경우 두 규칙을 모두 활성화할 수는 없습니다. 규칙 중 하나를 선택하여 적용 및 비활성화하거나 충돌하는 규칙을 제거하십시오.

두 개의 VM-VM 선호도 규칙이 충돌하면 더 오래된 규칙이 우선 적용되고 새 규칙은 비활성화됩니다. DRS는 활성화된 규칙만 충족하려고 하며 비활성화된 규칙은 무시됩니다. DRS는 선호도 규칙 위반보다 반선호도 규칙 위반을 방지하는 데 더 높은 우선 순위를 부여합니다.

VM 호스트 선호도 규칙 생성

VM-호스트 선호도 규칙을 생성하여 선택된 가상 시스템 그룹의 구성원이 특정 호스트 그룹의 구성원에서 실행될 수 있는지 여부를 지정할 수 있습니다.

사전 요구 사항

VM-호스트 선호도 규칙이 적용되는 가상 시스템 및 호스트의 그룹을 생성합니다.

절차

- 1 vSphere Client에서 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭합니다.
- 3 구성에서 **VM/호스트 규칙**을 클릭합니다.
- 4 **추가**를 클릭합니다.
- 5 **VM/호스트 규칙 생성** 대화상자에서 규칙 이름을 입력합니다.
- 6 **유형** 드롭다운 메뉴에서 **가상 시스템에서 호스트로**를 선택합니다.
- 7 규칙이 적용되는 가상 시스템 그룹 및 호스트 그룹을 선택합니다.

8 규칙의 규격을 선택합니다.

- **반드시 그룹의 호스트에서 실행되어야 합니다.** VM 그룹 1의 가상 시스템은 반드시 호스트 그룹 A의 호스트에서 실행되어야 합니다.
- **그룹의 호스트에서 실행되어야 합니다.** VM 그룹 1의 가상 시스템은 가급적이면 호스트 그룹 A의 호스트에서 실행되어야 합니다.
- **반드시 그룹의 호스트에서 실행되지 않아야 합니다.** VM 그룹 1의 가상 시스템은 호스트 그룹 A의 호스트에서 절대로 실행되지 않아야 합니다.
- **그룹의 호스트에서 실행되지 않아야 합니다.** VM 그룹 1의 가상 시스템은 가급적이면 호스트 그룹 A의 호스트에서 실행되지 않아야 합니다.

9 확인을 클릭합니다.

VM-호스트 선호도 규칙 사용

VM-호스트 선호도 규칙을 사용하여 가상 시스템 그룹과 호스트 그룹 간의 선호도 관계를 지정합니다. VM-호스트 선호도 규칙을 사용할 경우에는 해당 규칙이 가장 유용한 상황, 규칙 간의 충돌을 해결하는 방법 및 필수 선호도 규칙을 설정할 경우 주의해야 할 사항을 알고 있어야 합니다.

둘 이상의 VM-호스트 선호도 규칙을 생성할 경우에는 규칙의 순위가 정해지지 않고 동등하게 적용됩니다. 규칙이 상호 작용하는 방식에 이러한 방식이 미치는 영향에 유의해야 합니다. 예를 들어 서로 다른 필수 규칙에 속하는 두 그룹에 속하는 가상 시스템은 규칙에 표시된 두 호스트 그룹에 모두 속하는 호스트에서만 실행할 수 있습니다.

VM-호스트 선호도 규칙을 생성하면 다른 규칙과 관련되어 작동하는 기능에 대해 검사가 이루어지지 않습니다. 따라서 사용 중인 다른 규칙과 충돌하는 규칙이 생성될 수 있습니다. 두 개의 VM-호스트 선호도 규칙이 충돌하면 더 오래된 규칙이 우선 적용되고 새 규칙은 비활성화됩니다. DRS는 활성화된 규칙만 충족하려고 하며 비활성화된 규칙은 무시됩니다.

DRS, vSphere HA 및 vSphere DPM 필요한 선호도 규칙(가상 시스템 그룹이 호스트 그룹에서 '실행되어야' 또는 '실행되지 않아야 하는 규칙')을 위반하는 작업을 수행하지 않습니다. 따라서 이러한 유형의 규칙을 사용할 때는 잠재적으로 클러스터의 운영에 좋지 않은 영향을 줄 수 있기 때문에 주의해야 합니다. 잘못 사용될 경우 필수 VM-호스트 선호도 규칙은 클러스터를 조각화할 수 있고 DRS, vSphere HA 및 vSphere DPM이 제대로 작동하지 못하게 만들 수 있습니다.

클러스터 기능을 수행하는 것이 필수 선호도 규칙을 위반하게 되는 경우에는 해당 클러스터 기능이 수행되지 않습니다.

- DRS가 호스트를 유지 보수 모드로 전환할 때 가상 시스템을 제거하지 않습니다.
- DRS가 가상 시스템의 전원을 켜지 않거나 가상 시스템의 로드 밸런싱을 수행하지 않습니다.
- vSphere HA가 페일오버를 수행하지 않습니다.
- vSphere DPM이 호스트를 대기 모드로 전환하여 전원 관리를 최적화하지 않습니다.

이러한 상황을 방지하려면 두 개 이상의 필수 선호도 규칙을 생성할 때 주의를 기울이거나 기본 설정 전용인 VM-호스트 선호도 규칙(가상 시스템 그룹이 호스트 그룹에서 '실행되어야' 또는 '실행되지 않아야'하는 규칙)을 사용하는 것이 좋습니다. 각 가상 시스템이 선호하는 클러스터의 호스트 수가 호스트 하나를 손실하더라도 가상 시스템이 실행되기에 부족하지 않을 만큼 충분한지 확인하십시오. DRS, vSphere HA 및 vSphere DPM의 적절한 작동을 허용하기 위해 기본 설정 규칙을 위반할 수 있습니다.

참고 가상 시스템이 VM-호스트 선호도 규칙을 위반할 경우 트리거되는 이벤트에 기반하는 경보를 생성할 수 있습니다. 가상 시스템에 대한 새 경보를 추가하고 **VM에서 VM-호스트 선호도 규칙을 위반함**을 이벤트 트리거로 선택합니다. 경보 생성 및 편집에 대한 자세한 내용은 vSphere 모니터링 및 성능 설명서를 참조하십시오.

vSphere DRS 없이 선호도 규칙 사용

vSphere DRS를 사용하지 않고도 선호도 규칙을 사용할 수 있습니다.

VM 선호도 규칙을 사용하면 vSphere 클러스터 내의 VM 배치를 제어할 수 있습니다. VM-VM선호도는 VM이 클러스터 내의 동일한 호스트에서 실행되어야 한다는 것을 의미합니다. 워크로드는 서로 종속된 여러 VM으로 구성됩니다. **가상 시스템을 함께 보관** 규칙을 사용하면 선택한 모든 VM이 동일한 호스트에서 실행됩니다.

VM-VM 반선호도는 VM이 클러스터 내의 다른 호스트에서 실행되어야 함을 의미합니다. 리소스 사용이 많은 VM이 여러 개 있는 경우 **가상 시스템을 별도로 보관** 규칙을 사용하여 이러한 VM이 항상 클러스터의 다른 호스트에서 실행되어 리소스 경합을 방지할 수 있습니다.

VM-호스트 선호도 규칙은 VM-호스트 관계를 제어하며 4개의 하위 규칙이 있습니다. VM-호스트 규칙의 사용 사례에는 혼합 하드웨어 클러스터의 특정 호스트에서 워크로드가 실행되는지(또는 실행되지 않는지) 확인하는 것이 포함됩니다. 라이선싱 때문에 애플리케이션이 특정 호스트 또는 특정 호스트 하위 집합에서만 실행되도록 해야 할 수도 있습니다.

VM-호스트 규칙은 VM-호스트 그룹을 활용합니다. VM 그룹은 하나 이상의 VM으로 구성된 사용자 정의 그룹입니다. 호스트 그룹은 하나 이상의 호스트로 구성된 사용자 정의 그룹입니다.

규칙	설명	지속성
반드시 그룹의 호스트에서 실행되어야 합니다	VM 그룹은 정의된 호스트 그룹에서 실행되어야 합니다.	위반할 수 없는 하드 규칙입니다.
반드시 그룹의 호스트에서 실행되지 않아야 합니다	정의된 호스트 그룹에서 VM 그룹을 실행하면 안 됩니다.	위반할 수 없는 하드 규칙입니다.
그룹의 호스트에서 실행되어야 합니다	VM 그룹은 정의된 호스트 그룹에서 실행되어야 합니다.	위반될 수 있는 소프트 규칙입니다.
그룹의 호스트에서 실행되지 않아야 합니다	정의된 호스트 그룹에서 VM 그룹을 실행하면 안 됩니다.	위반될 수 있는 소프트 규칙입니다.

vSphere DRS가 활성화되지 않은 경우 VM-호스트 그룹 및 VM-호스트 규칙을 생성, 삭제 및 편집할 수 있습니다. VM-호스트 그룹 및 VM-호스트 규칙을 관리하는 인터페이스는 vSphere DRS 구성 외부에 있습니다.

VM의 vSphere HA 페일오버가 발생하는 경우 선호도 규칙이 준수됩니다. 이는 하드 VM-호스트 규칙 유형뿐 아니라 **가상 시스템을 함께 보관** 및 **가상 시스템을 별도로 보관** 규칙 유형에도 적용됩니다. 하드 규칙을 위반할 경우 vSphere HA는 VM을 페일오버할 수 없습니다. 소프트 VM-호스트 규칙을 사용하면 규칙을 위반하더라도 vSphere HA 작업이 성공합니다. 소프트 규칙은 정의 위반을 허용합니다.

예를 들어 ESX-01에 VM-1을 배치하고 ESX-02에 VM-2를 배치하도록 정의된 2노드 클러스터 및 **가상 시스템을 별도로 보관** 규칙이 있습니다. ESX-01에 오류가 발생하고 vSphere HA가 시작됩니다. vSphere HA는 ESX-02에서 VM-1의 전원을 켤 수 없습니다. 이는 각 가상 시스템을 별도로 유지하는 규칙을 위반하는 것이기 때문입니다.

VM 전원 켜기 작업은 선호도 규칙을 위반하는 것을 허용하지 않습니다. 이는 하드 VM-호스트 규칙뿐 아니라 **가상 시스템을 함께 보관** 및 **가상 시스템을 별도로 보관** 규칙 유형에 적용됩니다. 소프트 VM-호스트 규칙을 사용하는 경우 규칙을 위반하더라도 전원 켜기 작업이 성공합니다. 소프트 규칙은 정의 위반을 허용합니다.

예를 들어 VM-1과 VM-2에 **가상 시스템을 함께 보관**하도록 규칙이 정의되어 있고 VM-1의 전원이 이미 켜져 있는 경우 VM-2가 다른 호스트에 등록되어 있으면 전원을 켜려는 시도가 실패합니다. vSphere DRS가 활성화되지 않았기 때문에 VM의 전원을 켜려면 사용자가 수동으로 VM을 동일한 호스트로 마이그레이션해야 합니다.

vSphere DRS가 활성화되지 않아 VM이 자동으로 마이그레이션되지 않는 경우 선호도 규칙이 정의된 VM에서 vSphere vMotion을 사용하여 마이그레이션을 시작할 수 있습니다. 두 규칙 유형 **가상 시스템을 함께 보관** 및 **가상 시스템을 별도로 보관**은 vSphere vMotion을 사용하여 마이그레이션하면 위반될 수 있습니다. 예를 들어, VM-1과 VM-2에서 **가상 시스템을 별도로 보관**하는 규칙이 정의된 경우 VM을 동일한 호스트로 마이그레이션하려는 시도가 차단되지 않습니다. 이 작업은 규칙을 위반하더라도 성공합니다.

사용자가 시작한 마이그레이션에서는 하드 VM-호스트 규칙을 위반할 수 없습니다. 예를 들어 VM-1 및 VM-2가 호스트 ESX-01 및 ESX-02에서 실행되어야 한다는 규칙이 정의된 경우 VM을 호스트 그룹 외부의 호스트로 마이그레이션하려는 시도가 차단됩니다.

사용자가 시작한 마이그레이션으로 인해 소프트 VM-호스트 규칙이 위반될 수 있습니다. 소프트 규칙은 정의 위반을 허용합니다. 예를 들어 VM-1 및 VM-2가 호스트 ESX-03 및 ESX-04에서 실행되지 않아야 한다는 규칙이 정의된 경우 VM을 ESX-03 또는 ESX-04로 마이그레이션하려는 시도가 차단되지 않습니다.

vSphere DRS가 활성화되지 않으면 VM 배치에 대한 자동 업데이트 적용이 없습니다. 여기에는 수동 마이그레이션으로 인한 위반과 현재 VM 배치를 위반하는 새 규칙이 정의된 경우 발생하는 위반이 포함됩니다. vSphere DRS가 없으면 규칙을 준수하도록 VM 배치를 수정해야 합니다.

데이터스토어 클러스터는 공유 리소스 및 공유 관리 인터페이스를 사용하는 데이터스토어의 모음입니다. 데이터스토어 클러스터와 데이터스토어의 관계는 클러스터와 호스트의 관계와 같습니다. 데이터스토어 클러스터를 생성할 때 vSphere Storage DRS 사용하여 스토리지 리소스를 관리할 수 있습니다.

참고 데이터스토어 클러스터를 vSphere API 스토리지 포드라고 합니다.

데이터스토어 클러스터에 데이터스토어를 추가하면 데이터스토어의 리소스가 데이터스토어 클러스터 리소스의 일부가 됩니다. 호스트의 클러스터와 마찬가지로 데이터스토어 클러스터를 사용하여 스토리지 리소스를 집계하면 데이터스토어 클러스터 수준에서 리소스 할당 정책을 지원할 수 있습니다. 데이터스토어 클러스터별로 다음과 같은 리소스 관리 기능도 사용할 수 있습니다.

공간 사용률 로드 밸런싱

공간 사용에 대한 임계값을 설정할 수 있습니다. 데이터스토어의 공간 사용이 임계값을 초과하면 Storage DRS가 권장 사항을 생성하거나 Storage vMotion 마이그레이션을 수행하여 데이터스토어 클러스터 전체에서 공간 사용의 균형을 조정합니다.

I/O 지연 시간 로드 밸런싱

병목 현상 방지를 위해 I/O 지연 시간 임계값을 설정할 수 있습니다. 데이터스토어의 I/O 지연 시간이 임계값을 초과하면 Storage DRS가 권장 사항을 생성하거나 Storage vMotion 마이그레이션을 수행하여 높은 I/O 로드를 완화할 수 있습니다.

반선호도 규칙

가상 시스템 디스크에 대한 반선호도 규칙을 생성할 수 있습니다. 예를 들어 특정 가상 시스템의 가상 디스크는 서로 다른 데이터스토어에 보관해야 합니다. 기본적으로 가상 시스템의 모든 가상 디스크는 동일한 데이터스토어에 배치됩니다.

다음으로 아래 항목을 읽으십시오.

- 초기 배치 및 지속적인 균형 조정
- 스토리지 마이그레이션 권장 사항
- 데이터스토어 클러스터 생성
- Storage DRS 활성화 및 비활성화
- 데이터스토어 클러스터의 자동화 수준 설정

- Storage DRS의 강도 수준 설정
- 데이터스토어 클러스터 요구 사항
- 데이터스토어 클러스터에서 데이터스토어 추가 및 제거

초기 배치 및 지속적인 균형 조정

Storage DRS에서는 Storage DRS 사용 데이터스토어 클러스터의 데이터스토어에 대한 초기 배치 및 지속적인 균형 조정 권장 사항을 제공합니다.

초기 배치는 Storage DRS가 데이터스토어 클러스터 내에서 가상 시스템 디스크를 배치할 데이터스토어를 선택할 때 수행됩니다. 이는 가상 시스템이 만들어지거나 복제될 때, 가상 시스템 디스크가 다른 데이터스토어 클러스터로 마이그레이션될 때 또는 기존 가상 시스템에 디스크를 추가할 때 발생합니다.

초기 배치 권장 사항은 공간 제약 조건과 공간 및 I/O 로드 밸런싱의 목표에 따라 만들어집니다. 이러한 권장 사항은 한 데이터스토어가 과다 프로비저닝되거나, 스토리지 I/O 병목 현상이 발생하거나, 가상 시스템의 성능이 저하될 위험을 최소화하는 것을 목표로 합니다.

Storage DRS는 구성된 빈도(기본적으로 8시간마다)에 따라 호출되거나 데이터스토어 클러스터에 있는 하나 이상의 데이터스토어가 사용자가 구성할 수 있는 공간 사용률 임계값을 초과할 때 호출됩니다. Storage DRS가 호출되면 Storage DRS는 임계값을 기준으로 각 데이터스토어의 공간 사용률과 I/O 지연 시간 값을 확인합니다. I/O 지연 시간의 경우 Storage DRS는 하루 동안 측정된 90번째 백분위수 I/O 지연 시간을 사용하여 임계값과 비교합니다.

스토리지 마이그레이션 권장 사항

vCenter Server의 Storage DRS 권장 사항 페이지에는 자동화 모드가 수동인 데이터스토어 클러스터에 대한 마이그레이션 권장 사항이 표시됩니다.

시스템에서는 Storage DRS 규칙을 적용하고 데이터스토어 클러스터의 공간 및 I/O 리소스 균형을 조정하는 데 필요한 만큼의 권장 사항을 제공합니다. 각 권장 사항에는 가상 시스템 이름, 가상 디스크 이름, 데이터스토어 클러스터 이름, 소스 데이터스토어, 대상 데이터스토어 및 권장 사유가 포함됩니다.

- 데이터스토어 공간 사용 조정
- 데이터스토어 I/O 로드 조정

Storage DRS에서는 다음과 같은 경우 마이그레이션을 위한 필수 권장 사항을 제공합니다.

- 데이터스토어에 공간이 부족한 경우
- 반선호도 또는 선호도 규칙이 위반된 경우
- 데이터스토어가 유지 보수 모드로 전환 중이며 이를 제거해야 하는 경우

또한 데이터스토어의 공간이 거의 소모되거나 공간 및 I/O 로드 밸런싱을 위해 조정이 필요한 경우에도 선택적 권장 사항이 제공됩니다.

Storage DRS에서는 공간 균형을 조정하기 위해 전원이 꺼져 있거나 켜져 있는 가상 시스템을 이동할 것을 고려합니다. Storage DRS는 스냅샷을 사용하여 이러한 고려 사항에 전원이 꺼진 시스템을 포함합니다.

데이터스토어 클러스터 생성

Storage DRS를 사용하여 데이터스토어 클러스터 리소스를 관리할 수 있습니다.

절차

- 1 vSphere Client에서 데이터 센터를 찾습니다.
- 2 데이터 센터를 마우스 오른쪽 버튼으로 클릭하고 **스토리지, 새 데이터스토어 클러스터**를 선택합니다.
- 3 **새 데이터스토어 클러스터** 마법사를 완료하려면 프롬프트에 따릅니다.
- 4 **마침**을 클릭합니다.

Storage DRS 활성화 및 비활성화

Storage DRS를 사용하면 데이터스토어 클러스터의 집계된 리소스를 관리할 수 있습니다. Storage DRS가 활성화되면 데이터스토어 클러스터의 데이터스토어 간에 공간 및 I/O 리소스의 밸런싱을 조정할 수 있도록 가상 시스템 디스크 배치 및 마이그레이션에 대한 권장 사항이 제공됩니다.

Storage DRS를 활성화하면 다음 기능이 활성화됩니다.

- 데이터스토어 클러스터 내의 데이터스토어 간 공간 로드 밸런싱
- 데이터스토어 클러스터 내 데이터스토어 간 I/O 로드 밸런싱
- 공간 및 I/O 워크로드에 기반한 가상 디스크의 초기 배치

데이터스토어 클러스터 설정 대화상자의 Storage DRS 사용 확인란을 사용하여 이러한 모든 구성 요소를 한 번에 활성화하거나 비활성화할 수 있습니다. 필요한 경우 공간 밸런싱 기능과 별개로 Storage DRS의 I/O 관련 기능을 비활성화할 수 있습니다.

데이터스토어 클러스터에서 Storage DRS를 비활성화한 경우에도 Storage DRS 설정은 유지됩니다. Storage DRS를 활성화하면 데이터스토어 클러스터의 설정이 Storage DRS가 비활성화된 지점으로 복원됩니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **서비스**를 클릭합니다.
- 3 **Storage DRS**를 선택하고 **편집**을 클릭합니다.
- 4 **vSphere DRS 설정**을 선택하고 **확인**을 클릭합니다.

- 5 (선택 사항) Storage DRS의 I/O 관련 기능만 비활성화하고 공간 관련 제어는 활성화된 상태로 두려면 다음 단계를 수행합니다.
 - a Storage DRS에서 **편집**을 선택합니다.
 - b Storage DRS에 대해 **I/O 메트릭 사용** 확인란의 선택을 취소하고 **확인**을 클릭합니다.

데이터스토어 클러스터의 자동화 수준 설정

데이터스토어 클러스터의 자동화 수준은 Storage DRS의 배치 및 마이그레이션 권장 사항을 자동으로 적용할지 여부를 지정합니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **서비스**를 클릭합니다.
- 3 **DRS**를 선택하고 **편집**을 클릭합니다.
- 4 DRS 자동화를 확장하고 자동화 수준을 선택합니다.

수동이 기본 자동화 수준입니다.

옵션	설명
자동화 안 함(수동 모드)	배치 및 마이그레이션 권장 사항이 표시되지만 권장 사항을 수동으로 적용하기 전까지 실행하지 않습니다.
완전히 자동화됨	배치와 마이그레이션 권장 사항이 자동으로 실행됩니다.

- 5 **확인**을 클릭합니다.

Storage DRS의 강도 수준 설정

Storage DRS의 강도는 사용된 공간과 I/O 지연 시간에 대한 임계값을 지정하여 결정됩니다.

Storage DRS는 데이터스토어 클러스터의 데이터스토어에 대한 리소스 사용 정보를 수집합니다. vCenter Server에서는 이 정보를 사용하여 데이터스토어에 가상 디스크를 배치하기 위한 권장 사항을 생성합니다.

데이터스토어 클러스터의 강도 수준을 낮게 설정할 경우 Storage DRS에서는 예를 들어 I/O 로드, 공간 사용률 또는 이러한 항목의 불균형이 높은 경우와 같이 반드시 필요한 경우에만 Storage vMotion 마이그레이션을 권장합니다. 데이터스토어 클러스터의 강도 수준을 높게 설정할 경우 Storage DRS에서는 공간 또는 I/O 로드 밸런싱을 수행하는 것이 데이터스토어 클러스터에 유리할 때마다 마이그레이션을 권장합니다.

vSphere Client에서 다음 임계값을 사용하여 Storage DRS의 강도 수준을 설정할 수 있습니다.

공간 사용률

데이터스토어의 공간 사용률(%)이 vSphere Client에서 설정한 임계값보다 크면 Storage DRS에서는 권장 사항을 생성하거나 마이그레이션을 수행합니다.

I/O 지연 시간

데이터스토어에 대해 하루 동안 측정된 90번째 백분위수 I/O 지연 시간이 임계값보다 크면 Storage DRS에서 권장 사항을 생성하거나 마이그레이션을 수행합니다.

고급 옵션을 설정하여 Storage DRS의 강도 수준을 세부적으로 구성할 수도 있습니다.

공간 사용률 차이

이 임계값은 소스와 대상의 공간 사용률 간의 차이가 최소로 유지되도록 합니다. 예를 들어, 데이터스토어 A에 사용된 공간이 82%이고 데이터스토어 B에 사용된 공간이 79%인 경우 차이는 3입니다. 이때 임계값이 5이면 Storage DRS는 데이터스토어 A에서 데이터스토어 B로의 마이그레이션 권장 사항을 제공하지 않습니다.

I/O 로드 밸런싱 호출 간격

이 간격 이후 Storage DRS가 실행되어 I/O 로드 균형을 조정합니다.

I/O 불균형 임계값

이 값을 낮추면 I/O 로드 밸런싱의 강도가 낮아집니다. Storage DRS는 0과 1 사이의 I/O 공정성 메트릭을 계산합니다. 1이 가장 공정한 배포를 나타냅니다. I/O 로드 밸런싱은 계산된 메트릭이 $1 - (I/O \text{ 불균형 임계값} / 100)$ 보다 작은 경우에만 실행됩니다.

Storage DRS 런타임 규칙 설정

Storage DRS 트리거를 설정하고 데이터스토어 클러스터의 고급 옵션을 구성합니다.

절차

- 1 (선택 사항) **SDRS 권장 사항에 대해 I/O 메트릭 사용** 확인란을 선택하거나 선택 취소하여 I/O 메트릭 포함을 활성화 또는 비활성화합니다.

이 옵션을 비활성화하면 vCenter Server는 I/O 메트릭을 고려하지 않고 Storage DRS 권장 사항을 제공합니다. 이 옵션을 비활성화하면 다음과 같은 Storage DRS 요소가 비활성화됩니다.

- 데이터스토어 클러스터 내 데이터스토어 간 I/O 로드 밸런싱
- I/O 워크로드에 기반한 가상 디스크의 초기 배치. 공간만 기초로 하여 초기 배치가 이루어집니다.

- 2 (선택 사항) Storage DRS 임계값을 설정합니다.

사용된 공간과 I/O 지연 시간에 대한 임계값을 지정하여 Storage DRS의 강도 수준을 설정합니다.

- 사용된 공간 슬라이더를 사용하여 Storage DRS가 트리거되기 전에 허용되는 사용된 공간의 최대 백분율을 지정합니다. 데이터스토어의 공간 사용량이 임계값을 초과하면 Storage DRS가 권장 사항을 제공하고 마이그레이션을 수행합니다.

- I/O 지연 시간 슬라이더를 사용하여 Storage DRS가 트리거되기 전에 허용되는 최대 I/O 지연 시간을 지정합니다. Storage DRS는 지연 시간이 임계값보다 클 경우 권장 사항을 제공하고 마이그레이션을 수행합니다.

참고 데이터스토어 클러스터의 Storage DRS I/O 지연 시간 임계값은 Storage I/O Control 정체 임계값보다 작거나 같아야 합니다.

3 (선택 사항) 고급 옵션을 구성합니다.

- 소스와 대상 간의 사용률 차이가 다음과 같을 때까지 권장 사항 없음: 슬라이더를 사용하여 공간 사용률 차이 임계값을 지정합니다. 사용률은 사용량 * 100/용량입니다.

이 임계값은 소스와 대상의 공간 사용률 간의 차이가 최소로 유지되도록 합니다. 예를 들어, 데이터스토어 A에 사용된 공간이 82%이고 데이터스토어 B에 사용된 공간이 79%인 경우 차이는 3입니다. 이때 임계값이 5이면 Storage DRS는 데이터스토어 A에서 데이터스토어 B로의 마이그레이션 권장 사항을 제공하지 않습니다.

- 불균형 확인 간격: Storage DRS가 공간 및 I/O 로드 밸런싱을 평가하는 간격을 지정합니다.
- I/O 불균형 임계값: 슬라이더를 사용하여 I/O 로드 밸런싱의 강도를 표시합니다. 이 값을 낮추면 I/O 로드 밸런싱의 강도가 낮아집니다. Storage DRS는 0과 1 사이의 I/O 공정성 메트릭을 계산합니다. 1이 가장 공정한 배포를 나타냅니다. I/O 로드 밸런싱은 계산된 메트릭이 1 - (I/O 불균형 임계값/100)보다 작은 경우에만 실행됩니다.

4 확인을 클릭합니다.

데이터스토어 클러스터 요구 사항

데이터스토어 클러스터 기능을 제대로 사용하려면 데이터스토어 클러스터와 연결된 데이터스토어 및 호스트가 특정 요구 사항을 충족해야 합니다.

데이터스토어 클러스터를 만들 때는 다음 지침을 따릅니다.

- 데이터스토어 클러스터에는 유사하거나 교체 가능한 데이터스토어가 포함되어 있어야 합니다.
데이터스토어 클러스터는 크기와 I/O 용량이 서로 다른 데이터스토어의 조합을 포함할 수 있으며 데이터스토어의 어레이 및 벤더도 서로 다를 수 있습니다. 그러나 다음과 같은 유형의 데이터스토어는 같은 데이터스토어 클러스터에 함께 포함될 수 없습니다.
 - NFS 및 VMFS 데이터스토어는 동일한 데이터스토어 클러스터에 결합할 수 없습니다.
 - 복제된 데이터스토어와 복제되지 않은 데이터스토어를 동일한 Storage DRS 사용 데이터스토어 클러스터에서 결합할 수 없습니다.
- 데이터스토어 클러스터의 데이터스토어에 연결된 모든 호스트는 ESXi 5.0 이상이어야 합니다. 데이터스토어 클러스터의 데이터스토어가 ESX/ESXi 4.x 이하의 호스트에 연결되어 있으면 Storage DRS가 실행되지 않습니다.
- 여러 데이터 센터에서 공유되는 데이터스토어는 데이터스토어 클러스터에 포함할 수 없습니다.

- 하드웨어 가속을 사용하지 않는 데이터스토어와 하드웨어 가속을 사용하는 데이터스토어를 동일한 데이터스토어 클러스터에 포함하지 않는 것이 좋습니다. 하드웨어 가속 지원 동작을 보장하려면 데이터스토어 클러스터의 데이터스토어는 같은 유형이어야 합니다.

데이터스토어 클러스터에서 데이터스토어 추가 및 제거

기존 데이터스토어 클러스터에서 데이터스토어를 추가하거나 제거할 수 있습니다.

vSphere Client 인벤토리의 호스트에 마운트된 모든 데이터스토어를 데이터스토어 클러스터에 추가할 수 있지만 여기에는 다음과 같은 예외가 있습니다.

- 데이터스토어에 연결된 모든 호스트는 ESXi 5.0 이상이어야 합니다.
- 데이터스토어는 vSphere Client의 동일한 인스턴스의 하나 이상의 데이터 센터에 있을 수 없습니다.

데이터스토어 클러스터에서 데이터스토어를 제거해도 데이터스토어는 vSphere Client 인벤토리에 유지되며 호스트에서 마운트 해제되지 않습니다.

데이터스토어 클러스터를 사용하여 vSphere 스토리지 리소스 관리

20

데이터스토어 클러스터를 만든 후 이를 사용자 지정하여 스토리지 I/O 및 공간 사용 리소스를 관리하는 데 사용할 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- Storage DRS 유지 보수 모드 사용
- Storage DRS 권장 사항 적용
- 가상 시스템에 대해 Storage DRS 자동화 수준 변경
- Storage DRS에 대한 근무 외 시간 스케줄링 설정
- Storage DRS 반선호도 규칙
- Storage DRS 통계 지우기
- 데이터스토어 클러스터와의 Storage vMotion 호환성

Storage DRS 유지 보수 모드 사용

데이터스토어를 유지 보수하기 위해 사용을 중지해야 할 경우에는 데이터스토어를 유지 보수 모드로 전환합니다. 데이터스토어는 사용자 요청에 의해서만 유지 보수 모드로 전환하거나 유지 보수 모드를 마칩니다.

유지 보수 모드는 Storage DRS 지원 데이터스토어 클러스터 내의 데이터스토어에서 사용할 수 있습니다. 독립형 데이터스토어는 유지 보수 모드로 전환할 수 없습니다.

유지 보수 모드로 전환되는 데이터스토어에 있는 가상 디스크는 수동으로 또는 Storage DRS를 사용하여 다른 데이터스토어로 마이그레이션해야 합니다. 데이터스토어를 유지 보수 모드로 전환하려고 하면 **배치 권장 사항** 탭에 마이그레이션 권장 사항, 가상 디스크를 마이그레이션할 수 있는 동일한 데이터스토어 클러스터 내의 데이터스토어 목록이 표시됩니다. **장애** 탭에 vCenter Server가 마이그레이션할 수 없는 디스크 목록과 그 이유를 표시합니다. Storage DRS 선호도 또는 반선호도 규칙으로 인해 디스크가 마이그레이션되지 않는 경우 [유지 보수에 대해 선호도 규칙 무시] 옵션을 사용하도록 선택할 수 있습니다.

모든 가상 디스크가 마이그레이션될 때까지 데이터스토어는 [유지 보수 모드로 전환되고 있음] 상태에 있습니다.

유지 보수 모드로 데이터스토어 전환

데이터스토어 서비스가 필요하지 않은 경우에는 데이터스토어를 Storage DRS 유지 보수 모드로 전환할 수 있습니다.

사전 요구 사항

유지 보수 모드로 전환되는 데이터스토어를 포함하는 데이터스토어 클러스터에 Storage DRS가 사용되도록 설정되어 있어야 합니다.

데이터스토어에 CD-ROM 이미지 파일이 저장되어 있지 않아야 합니다.

데이터스토어 클러스터에 둘 이상의 데이터스토어가 있어야 합니다.

절차

- 1 vSphere Client에서 데이터스토어를 찾습니다.
- 2 데이터스토어를 마우스 오른쪽 버튼으로 클릭하고 **유지 보수 모드 > 유지 보수 모드 설정**을 선택합니다
데이터스토어 유지 보수 모드 마이그레이션과 관련된 권장 사항 목록이 나타납니다.
- 3 (선택 사항) 배치 권장 사항 탭에서 적용하지 않을 권장 사항의 선택을 취소합니다.

참고 데이터스토어를 유지 보수 모드로 전환하려면 모든 디스크를 비워야 합니다. 권장 사항의 선택을 취소할 경우에는 영향을 받는 가상 시스템을 수동으로 이동해야 합니다.

- 4 필요한 경우 **권장 사항 적용**을 클릭합니다.

vCenter Server가 스토리지 vMotion을 사용하여 가상 디스크를 소스 데이터스토어에서 대상 데이터스토어로 마이그레이션하고, 데이터스토어가 유지 보수 모드로 전환됩니다.

결과

업데이트된 데이터스토어의 현재 상태가 데이터스토어 아이콘에 곧바로 반영되지 않을 수 있습니다. **새로 고침**을 클릭하면 아이콘이 즉시 업데이트됩니다.

유지 보수 모드에 대해 Storage DRS 선호도 규칙 무시

Storage DRS 선호도 또는 반선호도 규칙은 데이터스토어가 유지 보수 모드로 전환되지 못하도록 할 수 있습니다. 데이터스토어를 유지 보수 모드로 전환하면 이러한 규칙을 무시할 수 있습니다.

데이터스토어 클러스터의 [유지 보수에 대해 선호도 규칙 무시] 옵션을 활성화하면 vCenter Server에서는 데이터스토어가 유지 보수 모드로 전환되지 않도록 하는 Storage DRS 선호도 및 반선호도 규칙을 무시합니다.

Storage DRS 규칙은 철수 권장 사항의 경우에만 무시됩니다. vCenter Server에서는 공간 및 로드 밸런싱 권장 사항 또는 초기 배치 권장 사항을 만들 때 이 규칙을 위반하지 않습니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **서비스**를 클릭합니다.
- 3 **DRS**를 선택하고 **편집**을 클릭합니다.
- 4 **고급 옵션**을 확장하고 **추가**를 클릭합니다.
- 5 옵션 열에 **IgnoreAffinityRulesForMaintenance**를 입력합니다.

6 값 열에 1을 입력하여 이 옵션을 활성화합니다.

이 옵션을 비활성화하려면 0을 입력합니다.

7 **확인**을 클릭합니다.

결과

유지 보수 모드에 대한 선호도 규칙 무시 옵션이 데이터스토어 클러스터에 적용됩니다.

Storage DRS 권장 사항 적용

Storage DRS는 데이터스토어 클러스터 내의 모든 데이터스토어에 대한 리소스 사용량 정보를 수집합니다. Storage DRS는 이 정보를 사용하여 데이터스토어 클러스터의 데이터스토어 내 가상 시스템 디스크 배치에 대한 권장 사항을 생성합니다.

Storage DRS 권장 사항은 vSphere Client 데이터스토어 보기의 **Storage DRS** 탭에 표시됩니다. 또한 데이터 스토어를 Storage DRS 유지 보수 모드로 전환하려고 할 때도 권장 사항이 표시됩니다. Storage DRS 권장 사항을 적용한 경우 vCenter Server가 Storage vMotion을 통해 가상 시스템 디스크를 데이터스토어 클러스터의 다른 데이터스토어로 마이그레이션하여 리소스 균형을 조정합니다.

일부 권장 사항을 적용하려면 제안된 DRS 권장 사항 재정의 확인란을 선택하고 적용할 각각의 권장 사항을 선택합니다.

표 20-1. Storage DRS 권장 사항

레이블	설명
우선 순위	권장 사항의 우선 순위 수준(1-5). (기본적으로 숨겨짐)
권장 사항	Storage DRS에서 권장한 작업
이유	작업이 필요한 이유
이전 공간 사용률(소스) 및 (대상)	마이그레이션 전 소스 및 대상 데이터스토어에 사용된 공간 사용률 (%)입니다.
이후 공간 사용률(소스) 및 (대상)	마이그레이션 후 소스 및 대상 데이터스토어에 사용된 공간 사용률 (%)입니다.
이전 I/O 지연 시간(소스)	마이그레이션 전 소스 데이터스토어의 I/O 지연 시간 값입니다.
이전 I/O 지연 시간(대상)	마이그레이션 전 대상 데이터스토어의 I/O 지연 시간 값입니다.

Storage DRS 권장 사항 새로 고침

Storage DRS 마이그레이션 권장 사항은 vSphere Client의 **Storage DRS** 탭에 표시됩니다. Storage DRS를 실행하여 이러한 권장 사항을 새로 고칠 수 있습니다.

사전 요구 사항

vSphere Client 인벤토리에 하나 이상의 데이터스토어 클러스터가 있어야 합니다.

데이터스토어 클러스터에 Storage DRS를 사용하도록 설정합니다. **Storage DRS** 탭은 Storage DRS를 사용하도록 설정된 경우에만 나타납니다.

절차

- 1 vSphere Client 데이터스토어 보기에서 데이터스토어 클러스터를 선택하고 **Storage DRS** 탭을 클릭합니다.
- 2 **권장 사항** 보기를 선택하고 오른쪽 위의 **Storage DRS 실행** 링크를 클릭합니다.

결과

권장 사항이 업데이트됩니다. 마지막으로 업데이트한 날짜 타임 스탬프에는 Storage DRS 권장 사항을 새로 고친 시간이 표시됩니다.

가상 시스템에 대해 Storage DRS 자동화 수준 변경

개별 가상 시스템에 대해 데이터스토어 클러스터 차원의 자동화 수준을 재정의할 수 있습니다. 또한 기본 가상 디스크 선호도 규칙을 재정의할 수도 있습니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **구성**을 클릭합니다.
- 3 **VM 재정의**에서 **추가**를 선택합니다.
- 4 가상 시스템을 선택합니다.
- 5 자동화 수준 드롭다운 메뉴를 클릭하고 가상 시스템의 자동화 수준을 선택합니다.

옵션	설명
기본값(수동)	배치 및 마이그레이션 권장 사항이 표시되지만 권장 사항을 수동으로 적용하기 전까지 실행하지 않습니다.
완전히 자동화됨	배치와 마이그레이션 권장 사항이 자동으로 실행됩니다.
사용 안 함	vCenter Server는 가상 시스템을 마이그레이션하지 않고 마이그레이션 권고사항을 제공하지 않습니다.

- 6 **VMDK 함께 보관** 드롭다운 메뉴를 클릭하여 기본 VMDK 선호도를 재정의합니다.
[VMDK 선호도 규칙 재정의](#)의 내용을 참조하십시오.
- 7 **확인**을 클릭합니다.

Storage DRS에 대한 근무 외 시간 스케줄링 설정

완전히 자동화된 데이터스토어 클러스터의 마이그레이션이 작업량이 많지 않은 시간대에 수행될 가능성이 높아지도록 데이터스토어 클러스터의 Storage DRS 설정을 변경하는 스케줄링된 작업을 생성할 수 있습니다.

스케줄링된 작업을 생성하여 데이터스토어 클러스터에 대한 자동화 수준 및 강도 수준을 변경할 수 있습니다. 예를 들면, 성능을 우선하는 경우 스토리지 마이그레이션 발생을 최소화하기 위해 최대 사용 시간 동안에는 Storage DRS가 소극적으로 실행되도록 구성할 수 있습니다. 한가한 시간에는 Storage DRS가 급진 모드로 실행되고 보다 자주 호출될 수 있습니다.

사전 요구 사항

Storage DRS 기능을 설정합니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **서비스**를 클릭합니다.
- 3 **vSphere DRS**에서 **DRS 스케줄링** 버튼을 클릭합니다.
- 4 데이터스토어 클러스터 편집 대화상자에서 **SDRS 스케줄링**을 클릭합니다.
- 5 **DRS 자동화**를 확장합니다.
 - a 자동화 수준을 선택합니다.
 - b 마이그레이션 임계값을 설정합니다.

마이그레이션 슬라이더를 사용하여 클러스터의 로드 밸런싱을 조정하는 vCenter Server 권장 사항의 우선 순위 수준을 선택합니다.
 - c 가상 시스템 자동화 사용 여부를 선택합니다.

개별 가상 시스템에 대한 재정의는 [VM 재정의] 페이지에서 설정할 수 있습니다.
- 6 **전원 관리**를 확장합니다.
 - a 자동화 수준을 선택합니다.
 - b DPM 임계값을 설정합니다.

DPM 슬라이더를 사용하여 vCenter Server가 적용할 전원 권장 사항을 선택합니다.
- 7 작업 이름을 입력합니다.
- 8 생성한 작업에 대한 설명을 입력합니다.
- 9 구성된 스케줄러에서 **변경**을 클릭하고 작업을 실행할 시간을 선택한 후 **확인**을 클릭합니다.
- 10 작업 완료 시 알림 e-메일을 보낼 e-메일 주소를 입력합니다.
- 11 **확인**을 클릭합니다.

결과

스케줄링된 작업이 지정된 시간에 실행됩니다.

Storage DRS 반선호도 규칙

Storage DRS 반선호도 규칙을 생성하여 데이터스토어 클러스터 내에서 같은 데이터스토어에 배치하면 안 되는 가상 디스크를 제어할 수 있습니다. 기본적으로 가상 시스템의 가상 디스크는 같은 데이터스토어에 배치됩니다.

반선호도 규칙을 생성하면 데이터스토어 클러스터의 관련 가상 디스크에 해당 규칙이 적용됩니다. 반선호도 규칙은 초기 배치할 때 및 Storage DRS에서 권장하는 마이그레이션을 수행할 때 적용되지만 사용자가 시작하는 마이그레이션 작업에는 적용되지 않습니다.

참고 또한 반선호도 규칙은 데이터스토어 클러스터의 데이터스토어에 저장된 CD-ROM ISO 이미지 파일이나 사용자 정의 위치에 저장된 스왑 파일에는 적용되지 않습니다.

VM 반선호도 규칙

동일한 데이터스토어에 유지해서는 안 되는 가상 시스템을 지정합니다. **VM 반선호도 규칙 생성**의 내용을 참조하십시오.

VMDK 반선호도 규칙

특정 가상 시스템에 연결된 가상 디스크 중 서로 다른 데이터스토어에 두어야 할 가상 디스크를 지정합니다. **VMDK 반선호도 규칙 생성**의 내용을 참조하십시오.

가상 디스크를 데이터스토어 클러스터 외부로 이동하면 선호도 또는 반선호도 규칙이 해당 디스크에 더 이상 적용되지 않습니다.

반대로 선호도 및 반선호도 규칙이 이미 있는 데이터스토어 클러스터로 가상 디스크 파일을 이동하면 다음과 같은 동작이 적용됩니다.

- 데이터스토어 클러스터 B에 VM 내부 선호도 규칙이 설정되어 있는 경우, 가상 디스크를 데이터스토어 클러스터 A에서 데이터스토어 클러스터 B로 이동하면 데이터스토어 클러스터 A에서 해당 가상 시스템의 가상 디스크에 적용되었던 모든 규칙이 더 이상 적용되지 않습니다. 이제 가상 디스크에는 데이터스토어 클러스터 B의 VM 내부 선호도 규칙이 적용됩니다.
- 데이터스토어 클러스터 B에 VM 반선호도 규칙이 설정되어 있는 경우, 가상 디스크를 데이터스토어 클러스터 A에서 데이터스토어 클러스터 B로 이동하면 데이터스토어 클러스터 A에서 해당 가상 시스템의 가상 디스크에 적용되었던 모든 규칙이 더 이상 적용되지 않습니다. 이제 가상 디스크에는 데이터스토어 클러스터 B의 VM 반선호도 규칙이 적용됩니다.
- 데이터스토어 클러스터 B에 VMDK 반선호도 규칙이 설정되어 있는 경우, 가상 디스크를 데이터스토어 클러스터 A에서 데이터스토어 클러스터 B로 이동하면 규칙은 데이터스토어 클러스터 B에 있는 지정된 가상 디스크로만 제한되기 때문에 VMDK 반선호도 규칙은 해당 가상 시스템의 가상 디스크에 적용되지 않습니다.

참고 Storage DRS 규칙은 데이터스토어가 유지 보수 모드로 전환되지 못하도록 할 수 있습니다. 유지 보수에 대해 선호도 규칙 무시 옵션을 사용하도록 설정하여 유지 보수 모드에 대해 Storage DRS 규칙을 무시하도록 선택할 수 있습니다.

VM 반선택도 규칙 생성

반선택도 규칙을 생성하여 특정 가상 시스템의 모든 가상 디스크가 서로 다른 데이터스토어에 보관되어야 함을 나타낼 수 있습니다. 규칙은 개별 데이터스토어 클러스터에 적용됩니다.

데이터스토어 클러스터의 VM 반선택도 규칙에 참여하는 가상 시스템은 데이터스토어 클러스터의 VM 내 선택도 규칙과 연결되어야 합니다. 또한 가상 시스템은 VM 내 선택도 규칙을 준수해야 합니다.

가상 시스템이 VM 반선택도 규칙에 따르는 경우 다음의 동작이 적용됩니다.

- Storage DRS는 규칙에 따라 가상 시스템의 가상 디스크를 배치합니다.
- 데이터스토어를 유지 보수 모드로 설정하는 등 반드시 마이그레이션해야 하는 경우라도 Storage DRS는 규칙에 따라 vMotion을 사용하여 가상 디스크를 마이그레이션합니다.
- 가상 시스템의 가상 디스크가 규칙을 위반하면 Storage DRS는 오류를 해결하기 위한 마이그레이션 권장 사항을 제공하거나 오류 해결을 위한 권장 사항을 제공할 수 없으면 위반을 장애로 보고합니다.

VM 반선택도 규칙은 기본으로 정의되지 않습니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **구성**을 클릭합니다.
- 3 **VM/호스트 규칙**을 선택합니다.
- 4 **추가**를 클릭합니다.
- 5 규칙의 이름을 입력합니다.
- 6 유형 메뉴에서 **VM 반선택도**를 선택합니다.
- 7 **추가**를 클릭합니다.
- 8 **가상 시스템 선택**을 클릭합니다.
- 9 최소한 두 개의 가상 시스템을 선택하고 **확인**을 클릭합니다.
- 10 **확인**을 클릭하여 규칙을 저장합니다.

VMDK 반선택도 규칙 생성

가상 시스템에 대해 VMDK 반선택도 규칙을 생성하여 서로 다른 데이터스토어에 보관해야 할 가상 디스크를 나타낼 수 있습니다.

VMDK 반선택도 규칙은 모든 가상 시스템에 적용되는 것이 아니라 해당 규칙이 정의된 가상 시스템에 적용됩니다. 이 규칙은 서로 분리된 가상 디스크의 목록으로 나타냅니다.

가상 시스템에 VMDK 반선택도 규칙과 VM 내 선택도 규칙의 설정을 시도하면 vCenter Server는 가장 최근에 정의한 규칙을 거부합니다.

가상 시스템이 VMDK 반선호도 규칙에 따르는 경우 다음의 동작이 적용됩니다.

- Storage DRS는 규칙에 따라 가상 시스템의 가상 디스크를 배치합니다.
- 데이터스토어를 유지 보수 모드로 설정하는 등 반드시 마이그레이션해야 하는 경우라도 Storage DRS는 규칙에 따라 vMotion을 사용하여 가상 디스크를 마이그레이션합니다.
- 가상 시스템의 가상 디스크가 규칙을 위반하면 Storage DRS는 오류를 해결하기 위한 마이그레이션 권장 사항을 제공하거나 오류 해결을 위한 권장 사항을 제공할 수 없으면 위반을 장애로 보고합니다.

VMDK 반선호도 규칙은 기본으로 정의되지 않습니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **구성**을 클릭합니다.
- 3 **VM/호스트 규칙**을 선택합니다.
- 4 **추가**를 클릭합니다.
- 5 규칙의 이름을 입력합니다.
- 6 유형 메뉴에서 **VMDK 반선호도**를 선택합니다.
- 7 **추가**를 클릭합니다.
- 8 **가상 시스템 선택**을 클릭합니다.
- 9 가상 시스템을 선택하고 **확인**을 클릭합니다.
- 10 규칙이 적용될 두 개 이상의 가상 디스크를 선택하고 **확인**을 클릭합니다.
- 11 **확인**을 클릭하여 규칙을 저장합니다.

VMDK 선호도 규칙 재정의

VMDK 선호도 규칙은 특정 가상 시스템과 연결된 데이터스토어 클러스터의 모든 가상 디스크가 데이터스토어 클러스터에서 같은 데이터스토어에 있음을 나타냅니다. 규칙은 개별 데이터스토어 클러스터에 적용됩니다.

VMDK 선호도 규칙은 데이터스토어 클러스터의 모든 가상 시스템에 대해 기본적으로 활성화됩니다. 데이터스토어 클러스터 또는 개별 가상 시스템에 대해 기본 설정을 재정의할 수 있습니다.

VMDK 선호도 규칙에 따르는 가상 시스템은 다음의 동작을 수행합니다.

- Storage DRS는 규칙에 따라 가상 시스템의 가상 디스크를 배치합니다.
- 데이터스토어를 유지 보수 모드로 설정하는 등 반드시 마이그레이션해야 하는 경우라도 Storage DRS는 규칙에 따라 vMotion을 사용하여 가상 디스크를 마이그레이션합니다.
- 가상 시스템의 가상 디스크가 규칙을 위반하면 Storage DRS는 오류를 해결하기 위한 마이그레이션 권장 사항을 제공하거나 오류 해결을 위한 권장 사항을 제공할 수 없으면 위반을 장애로 보고합니다.

데이터스토어를 Storage DRS에 대해 활성화된 데이터스토어 클러스터에 추가하면 가상 디스크가 해당 데이터스토어뿐만 아니라 다른 데이터스토어에도 있는 모든 가상 시스템에 대해 VMDK 선호도 규칙이 비활성화됩니다.

절차

- 1 vSphere Client에서 데이터스토어 클러스터를 찾습니다.
- 2 **구성** 탭을 클릭하고 **구성**을 클릭합니다.
- 3 **VM 재정의**를 선택합니다.
- 4 **추가**를 클릭합니다.
- 5 **+** 버튼을 사용하여 가상 시스템을 선택합니다.
- 6 **VMDK 함께 보관** 드롭다운 메뉴를 클릭하고 **아니요**를 선택합니다.
- 7 **확인**을 클릭합니다.

Storage DRS 통계 지우기

Storage DRS 관련 문제를 진단하기 위해 Storage DRS를 수동으로 실행하기 전에 Storage DRS 통계를 지울 수 있습니다.

중요 Storage DRS 통계를 지우는 옵션을 활성화하면 이 옵션을 비활성화할 때까지 Storage DRS를 실행할 때마다 통계가 지워집니다. 따라서 Storage DRS 문제를 진단한 후에는 항상 이 옵션을 비활성화해야 합니다.

사전 요구 사항

데이터스토어 클러스터에 대해 Storage DRS를 활성화합니다.

절차

- 1 **ClearIoStatsOnSdrsRun** 옵션을 활성화합니다.
 - a vSphere Client에서 데이터스토어 클러스터를 찾습니다.
 - b **구성** 탭을 클릭하고 **서비스**를 클릭합니다.
 - c **vSphere DRS**를 선택하고 **편집**을 클릭합니다.
 - d **고급 옵션**을 확장하고 **추가**를 클릭합니다.
 - e 옵션 열에 **ClearIoStatsOnSdrsRun**을 입력합니다.
 - f 해당하는 값 텍스트 상자에 **1**을 입력합니다.
 - g **확인**을 클릭합니다.
- 2 데이터스토어 클러스터에서 Storage DRS를 실행합니다.

vSphere Client 인벤토리에 있는 모든 데이터스토어 클러스터의 모든 데이터스토어 및 가상 디스크에 대한 현재 Storage DRS 통계가 지워지고 새 통계가 수집되지 않습니다.
- 3 **ClearIoStatsOnSdrsRun** 플러그 값을 **0**으로 변경하여 이 옵션을 비활성화합니다.

4 Storage DRS를 다시 실행합니다.

Storage DRS가 정상적으로 실행됩니다. 새 설정이 적용되는 데는 몇 시간이 걸릴 수 있습니다.

데이터스토어 클러스터와의 Storage vMotion 호환성

데이터스토어 클러스터에는 vSphere Storage vMotion[®] 과 관련된 몇 가지 요구 사항이 적용됩니다.

- 호스트에서 Storage vMotion을 지원하는 버전의 ESXi를 실행하고 있어야 합니다.
- 호스트에 소스 데이터스토어와 대상 데이터스토어 모두에 대한 쓰기 액세스 권한이 있어야 합니다.
- 호스트의 사용 가능한 메모리 리소스가 Storage vMotion을 수용할 수 있을 만큼 충분히 있어야 합니다.
- 대상 데이터스토어에 충분한 디스크 공간이 있어야 합니다.
- 대상 데이터스토어가 현재 유지 보수 모드에 있거나 유지 보수 모드로 전환 중이어서는 안 됩니다.

ESXi는 NUMA(통일되지 않은 메모리 액세스)를 지원하는 서버 아키텍처에서 Intel 및 AMD Opteron 프로세서에 대한 메모리 액세스 최적화를 지원합니다.

ESXi NUMA 스케줄링이 수행되는 방식과 VMware NUMA 알고리즘의 작동 방식을 이해한 후에는 NUMA 컨트롤을 지정하여 가상 시스템의 성능을 최적화할 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- NUMA란?
- 운영 체제 문제점
- ESXi NUMA 스케줄링 작동 방식
- VMware NUMA 최적화 알고리즘 및 설정
- 홈 노드 및 초기 배치
- 동적 로드 밸런싱 및 페이지 마이그레이션
- NUMA용으로 최적화된 투명 페이지 공유
- NUMA 아키텍처의 리소스 관리
- 가상 NUMA 사용
- ESXi 8.0의 가상 토폴로지
- 가상 NUMA 제어
- NUMA 제어 지정
- 특정 프로세서에 가상 시스템 연결
- 메모리 선호도를 사용하여 특정 NUMA 노드와 메모리 할당 연결
- 가상 시스템을 지정된 NUMA 노드에 연결

NUMA란?

NUMA 시스템은 둘 이상의 시스템 버스가 있는 고급 서버 플랫폼입니다. 가격 대비 성능이 우수한 단일 시스템 이미지에서 많은 수의 프로세서를 활용할 수 있습니다.

처리 능력을 효과적으로 사용하려면 멀티 기가헤르츠 CPU에 대량의 메모리 대역폭을 제공해야 합니다. 과학적 컴퓨팅 애플리케이션과 같이 메모리 사용량이 많은 워크로드를 실행하는 단일 CPU라도 메모리 대역폭의 제약을 받을 수 있습니다.

이런 문제는 다수의 프로세서가 동일한 시스템 버스에서 대역폭을 놓고 경쟁해야 하는 SMP(Symmetric Multiprocessing) 시스템에서 증폭됩니다. 일부 고급 시스템은 고속 데이터 버스를 구축하여 이 문제를 해결하려고 시도하는 경우가 많습니다. 그러나 이러한 솔루션은 비용이 많이 들고 확장성이 제한됩니다.

NUMA는 고성능 연결을 사용하여 비용 효율적인 소형 노드 여러 개를 연결하는 대안적 접근 방식입니다. 각 노드에는 소형 SMP 시스템과 마찬가지로 프로세서와 메모리가 포함됩니다. 그러나 고급 메모리 컨트롤러를 사용하면 노드가 다른 모든 노드의 메모리를 사용하여 단일 시스템 이미지를 생성할 수 있습니다. 프로세서가 자체 노드(원격 메모리) 내에 있지 않은 메모리에 액세스하는 경우 로컬 메모리에 액세스하는 것보다 느린 NUMA 연결을 통해 데이터를 전송해야 합니다. 메모리 액세스 시간은 일정하지 않으며 기술 이름에서 알 수 있듯이 메모리의 위치 및 메모리 액세스에 사용되는 노드에 따라 달라집니다.

운영 체제 문제점

NUMA 아키텍처는 단일 시스템 이미지를 제공하기 때문에 별도의 최적화 없이 운영 체제를 실행할 수 있습니다.

원격 메모리 액세스의 지연 시간이 길어지면 로컬 노드로 데이터가 전송될 때까지 지속적으로 대기하여 프로세서가 미달 사용될 수 있으며, 많은 양의 메모리를 사용하는 대역폭을 요청하는 애플리케이션으로 인해 NUMA 연결에서 병목 현상이 나타날 수 있습니다.

뿐만 아니라 이러한 시스템에서는 성능이 매우 불안정할 수 있습니다. 예를 들어, 애플리케이션을 벤치마킹 실행할 때 한 번은 로컬 메모리를 사용하고 다음 번에는 모든 메모리가 원격 노드에 있는 경우 성능 차이가 심할 수 있습니다. 이와 같은 현상 때문에 용량을 계획하기 어려울 수 있습니다.

몇몇 고급 UNIX 시스템은 컴파일러와 프로그래밍 라이브러리에 NUMA 최적화 기능을 지원합니다. 이 지원 기능을 사용하려면 소프트웨어 개발자가 최적의 성능을 위해 프로그램을 조정하고 다시 컴파일해야 합니다. 단일 시스템에 대한 최적화 설정은 동일한 시스템의 다음 세대에서도 제 기능을 하지 못할 수 있습니다. 애플리케이션을 실행할 노드를 관리자가 명시적으로 결정할 수 있도록 허용하는 시스템도 있습니다. 이러한 시스템은 전체 메모리가 로컬 메모리여야 하는 특정 애플리케이션에는 적합하지만 워크로드가 달라질 경우 관리가 어려워지고 노드 간의 불균형을 초래할 수 있습니다.

가장 이상적인 방법으로 시스템 소프트웨어는 애플리케이션에서 별도의 수정 없이 이 소프트웨어의 장점을 바로 활용할 수 있도록 투명한 NUMA 지원 기능을 제공합니다. 시스템에서는 관리자의 지속적인 개입 없이 로컬 메모리의 사용을 극대화하고 지능적으로 프로그램을 스케줄링해야 합니다. 마지막으로, 시스템에서는 공정성이나 성능에 영향을 주지 않으면서 변화하는 상황에 적절하게 대응해야 합니다.

ESXi NUMA 스케줄링 작동 방식

ESXi에서는 정교한 NUMA 스케줄러를 사용하여 프로세서 로드와 메모리 인접성의 균형을 동적으로 조정하거나 동적 프로세서 로드 밸런싱을 수행합니다.

- 1 NUMA 스케줄러에서 관리하는 각 가상 시스템에는 홈 노드가 할당됩니다. 홈 노드는 SRAT(시스템 리소스 할당 테이블)에 표시된 것과 같이 프로세서 및 로컬 메모리를 포함하는 시스템의 NUMA 노드 중 하나입니다.

- 2 가상 시스템에 메모리를 할당할 때 ESXi 호스트는 홈 노드의 메모리를 우선적으로 할당합니다. 메모리 인접성을 최대화하기 위해 가상 시스템의 가상 CPU는 홈 노드에서만 실행할 수 있습니다.
- 3 NUMA 스케줄러는 시스템 로드의 변화에 따라 가상 시스템의 홈 노드를 동적으로 변경할 수 있습니다. 프로세서 로드 불균형을 줄이기 위해 가상 시스템을 새 홈 노드로 마이그레이션할 수도 있습니다. 이 경우 원격 메모리가 많아지므로 스케줄러에서는 가상 시스템의 메모리를 새 홈 노드로 동적으로 마이그레이션하여 메모리 인접성을 높일 수 있습니다. NUMA 스케줄러에서는 전체 메모리 인접성을 높이는 데 도움이 될 경우 가상 시스템을 노드 간에 스왑할 수도 있습니다.

일부 가상 시스템은 ESXi NUMA 스케줄러에서 관리되지 않습니다. 예를 들어 가상 시스템의 프로세서 또는 메모리 선호도를 수동으로 설정한 경우에는 NUMA 스케줄러에서 이 가상 시스템을 관리할 수 없습니다. NUMA 스케줄러에서 관리되지 않는 가상 시스템도 올바르게 실행됩니다. 그러나 ESXi NUMA 최적화의 이점을 누릴 수는 없습니다.

ESXi의 NUMA 스케줄링 및 메모리 배치 정책은 모든 가상 시스템을 투명하게 관리할 수 있으므로 관리자가 노드 간에 가상 시스템의 균형을 조정하는 복잡한 작업을 명시적으로 처리할 필요가 없습니다.

최적화는 게스트 운영 체제의 종류에 관계없이 원활하게 작동합니다. ESXi는 NUMA 하드웨어를 지원하지 않는 가상 시스템(예: Windows NT 4.0)에도 NUMA 지원을 제공합니다. 따라서 레거시 운영 체제에서도 새 하드웨어를 활용할 수 있습니다.

단일 하드웨어 노드에서 사용할 수 있는 물리적 프로세서 코어보다 많은 수의 가상 프로세서가 있는 가상 시스템은 자동으로 관리할 수 있습니다. NUMA 스케줄러에서는 이러한 가상 시스템을 NUMA 노드에 분산하여 수용합니다. 즉, 가상 시스템을 여러 개의 NUMA 클라이언트로 분할하여 각각 하나의 노드에 할당하며, 분할된 각 클라이언트를 분할되지 않은 정상적인 클라이언트와 같이 관리합니다. 이렇게 하면 메모리 소모가 많은 일부 워크로드의 성능을 향상하면서 인접성을 높일 수 있습니다. 이 기능의 동작을 구성하는 방법은 [고급 가상 시스템 특성 항목을 참조하십시오](#).

ESXi에는 게스트 운영 체제에 가상 NUMA 토폴로지를 노출하기 위한 지원이 포함됩니다. 가상 NUMA 제어에 대한 자세한 내용은 [가상 NUMA 사용](#)의 내용을 참조하십시오.

VMware NUMA 최적화 알고리즘 및 설정

이 섹션에서는 ESXi에서 리소스 보장을 유지하면서 애플리케이션 성능을 최대화하는 데 사용하는 알고리즘 및 설정에 대해 설명합니다.

홈 노드 및 초기 배치

가상 시스템의 전원이 켜지면 ESXi는 가상 시스템에 홈 노드를 할당합니다. 가상 시스템은 홈 노드 내의 프로세서에서만 실행되며 새로 할당된 메모리도 홈 노드에서 제공됩니다.

가상 시스템의 홈 노드가 변경되지 않는 한 로컬 메모리만 사용하므로 다른 NUMA 노드에 대한 원격 메모리 액세스와 관련된 성능 저하를 방지할 수 있습니다.

가상 시스템의 전원이 켜지면 NUMA 노드 간의 전체 CPU 및 메모리 로드 균형이 유지하도록 가상 시스템에 초기 홈 노드가 할당됩니다. 큰 NUMA 시스템의 노드 간 지연 시간은 매우 다양할 수 있으므로 ESXi는 부팅 시 이러한 노드 간 지연 시간을 결정하고 단일 NUMA 노드보다 넓은 가상 시스템을 처음 배치할 때 이 정보를 사용합니다. 이러한 넓은 가상 시스템은 메모리 액세스 지연 시간을 최소화하기 위해 서로 가까운 NUMA 노드에 배치됩니다.

일반적으로 초기 배치 전용 접근 방식은 시스템이 실행되는 한 변경되지 않는 벤치마크 구성과 같이 단일 워크로드만 실행하는 시스템에는 충분합니다. 하지만 이 접근 방식은 변화하는 워크로드를 지원하는 데이터 센터급 시스템에 대해서는 양호한 성능과 공정성을 보장할 수 없습니다. 따라서 ESXi는 초기 배치 외에도 CPU 균형을 개선하고 메모리 인접성을 높이기 위해 NUMA 노드 간에 가상 CPU와 메모리를 동적으로 마이그레이션합니다.

동적 로드 밸런싱 및 페이지 마이그레이션

ESXi는 기존의 초기 배치 방법을 동적 재조정 알고리즘과 결합합니다. 시스템에서 주기적으로(기본적으로 2초 간격) 다양한 노드의 로드를 검사하고 노드 간에 가상 시스템을 이동하여 로드를 재조정해야 하는지 여부를 결정합니다.

이 계산에서는 공정성 또는 리소스 사용 권한을 위반하지 않고 성능을 향상시킬 수 있도록 가상 시스템 및 리소스 풀에 대한 리소스 설정을 고려합니다.

리밸런서는 적절한 가상 시스템을 선택하고 해당 시스템의 홈 노드를 가장 적게 로드된 노드로 변경합니다. 가능한 경우 리밸런서는 대상 노드에 이미 일부 메모리가 있는 가상 시스템을 이동합니다. 가상 시스템을 다시 이동하지 않으면 가상 시스템은 해당 시점부터 새 홈 노드에 메모리를 할당하고 새 홈 노드 내의 프로세서에서만 실행됩니다.

재조정은 공정성을 유지하고 모든 노드가 완전히 사용되도록 하는 효과적인 솔루션입니다. 리밸런서가 메모리를 거의 할당하지 않거나 전혀 할당하지 않은 노드로 가상 시스템을 이동해야 할 수도 있습니다. 이 경우에는 가상 시스템으로 인해 많은 수의 원격 메모리 액세스와 관련된 성능 저하가 발생합니다. ESXi는 가상 시스템의 원래 노드에서 새 홈 노드로 메모리를 투명하게 마이그레이션하여 이러한 문제가 발생하지 않게 할 수 있습니다.

- 1 이 시스템은 원래 노드에서 특정 페이지(4KB의 연속 메모리)를 선택한 다음 해당 페이지의 데이터를 대상 노드의 페이지로 복사합니다.
- 2 이 시스템은 가상 시스템 모니터 계층 및 프로세서의 메모리 관리 하드웨어를 사용하여 가상 시스템의 메모리 보기를 원활하게 다시 매핑합니다. 그러면 모든 추가 참조에 대상 노드의 페이지를 사용할 수 있으므로 원격 메모리 액세스로 인한 문제가 발생하지 않습니다.

가상 시스템이 새 노드로 이동되는 즉시 ESXi 호스트에서 이러한 방식으로 메모리 마이그레이션을 시작합니다. 가상 시스템은 특히 가상 시스템에 남아 있는 원격 메모리가 거의 없거나 대상 노드에 사용 가능한 메모리가 거의 없는 경우 시스템에 부담이 되지 않도록 속도를 관리합니다. 또한 메모리 마이그레이션 알고리즘을 통해 짧은 기간 동안만 가상 시스템을 새 노드로 이동하는 경우 ESXi 호스트가 불필요하게 메모리를 이동하지 않도록 할 수 있습니다.

초기 배치, 동적 재조정 및 지능형 메모리 마이그레이션이 결합하여 작동하면 워크로드가 변경되는 경우에도 NUMA 시스템에서 우수한 메모리 성능을 유지할 수 있습니다. 예를 들어 새 가상 시스템이 시작되는 경우와 같이 주요 워크로드 변경이 발생하는 경우 시스템에서 가상 시스템 및 메모리를 새 위치로 마이그레이션하는 재조정 작업에 시간이 걸립니다. 시스템은 일반적으로 몇 초 또는 몇 분의 짧은 기간에 걸쳐 재조정 작업을 완료하고 안정적인 상태에 도달합니다.

NUMA용으로 최적화된 투명 페이지 공유

많은 ESXi 작업 부하는 가상 시스템 간에 메모리를 공유할 기회를 제공합니다.

동일한 게스트 운영 체제의 인스턴스를 실행하는 여러 가상 시스템이 있거나 동일한 애플리케이션 또는 구성 요소가 로드되거나, 공통의 데이터를 포함하고 있을 수 있습니다. 이 경우 ESXi 시스템에서는 독점적 투명 페이지 공유 기술을 사용하여 메모리 페이지의 중복 복사본을 제거합니다. 메모리 공유를 사용할 경우, 가상 시스템에서 실행 중인 워크로드는 일반적으로 물리적 시스템에서 실행 중인 워크로드보다 소비하는 메모리 양이 적습니다. 결과적으로 높은 수준의 오버 커밋을 효율적으로 지원할 수 있습니다.

ESXi 시스템의 투명 페이지 공유는 NUMA 시스템에도 사용할 수 있도록 최적화되어 있습니다. NUMA 시스템에서 페이지는 노드 단위로 공유되므로 각 NUMA 노드에는 매우 자주 공유되는 페이지의 자체 로컬 복사본이 있습니다. 가상 시스템이 공유 페이지를 사용하는 경우 원격 메모리에 액세스할 필요가 없습니다.

참고 이 기본 동작은 모든 이전 버전의 ESX 및 ESXi에서 동일합니다.

NUMA 아키텍처의 리소스 관리

다양한 유형의 NUMA 아키텍처를 사용하여 리소스 관리를 수행할 수 있습니다.

고성능 다중 코어 시스템이 급증하면서 NUMA 아키텍처는 메모리 소모가 많은 워크로드의 성능 확장에 뛰어나다는 점 때문에 점점 더 대중화되고 있습니다. 모든 최신 Intel 및 AMD 시스템의 프로세서에는 NUMA 지원 기능이 내장되어 있습니다. 또한 특수 칩셋 지원을 통한 NUMA 동작으로 Intel 및 AMD 프로세서를 확장하는 IBM Enterprise X-Architecture와 같은 기존 NUMA 시스템이 있습니다.

일반적으로 BIOS 설정을 사용하여 NUMA 동작을 활성화하고 비활성화할 수 있습니다. 예를 들어 AMD Opteron 기반 HP Proliant 서버에서는 BIOS에서 노드 인터리빙을 활성화하여 NUMA를 비활성화할 수 있습니다. NUMA가 활성화되면 BIOS는 SRAT(시스템 리소스 할당 테이블)를 구축하며 이 테이블은 ESXi에서 최적화에 사용되는 NUMA 정보를 생성하는 데 사용됩니다. 스케줄링 공정성을 위해 NUMA 노드당 코어 수가 너무 적거나 전체적으로 코어 수가 너무 적은 시스템에 대해서는 NUMA 최적화가 활성화되지 않습니다.

`numa.rebalancecorestotal` 및 `numa.rebalancecoresnode` 옵션을 수정하여 이 동작을 변경할 수 있습니다.

가상 NUMA 사용

vSphere에는 게스트 운영 체제에 가상 NUMA 토폴로지를 노출하기 위한 지원이 포함되어 게스트 운영 체제와 애플리케이션의 NUMA 최적화가 용이해졌습니다. 따라서 성능이 향상될 수 있습니다.

가상 NUMA 토폴로지는 가상 시스템에서 사용할 수 있으며 가상 CPU 수가 8개를 초과하는 경우에 기본적으로 활성화됩니다. 고급 구성 옵션을 사용하여 가상 NUMA 토폴로지를 수동으로 조정할 수도 있습니다.

가상 NUMA가 활성화된 가상 시스템의 전원을 처음으로 켜면 가상 NUMA 토폴로지는 기본 물리적 호스트의 NUMA 토폴로지를 기반으로 합니다. 가상 시스템의 가상 NUMA 토폴로지가 한 번 초기화된 후에는 가상 시스템 내의 vCPU 수가 변경되지 않는 한 변경되지 않습니다.

가상 NUMA 토폴로지에는 가상 시스템에 구성된 메모리가 고려되지 않습니다. 가상 NUMA 토폴로지는 가상 시스템의 가상 소켓 수 및 소켓당 코어 수에 영향을 받지 않습니다.

가상 NUMA 토폴로지를 재정의해야 하는 경우에는 [가상 NUMA 제어](#)의 내용을 참조하십시오.

참고 CPU HotAdd를 사용하도록 설정하면 가상 NUMA가 비활성화됩니다. <https://kb.vmware.com/kb/2040375>의 내용을 참조하십시오.

ESXi 8.0의 가상 토폴로지

ESXi 8.0에는 향상된 가상 토폴로지 기능이 포함되어 있습니다.

VM의 가상 토폴로지를 사용하면 GOS 내에서 배치 및 로드 밸런싱을 최적화할 수 있습니다. VM이 실행 중인 호스트의 기본 물리적 토폴로지와 일치하는 정확한 가상 토폴로지를 선택하는 것은 애플리케이션 성능에 매우 중요합니다.

ESXi 8.0에서는 VM에 대한 최적의 `coresPerSocket`과 최적의 가상 L3 크기가 자동으로 선택됩니다. 또한 CPU 핫 플러그를 사용하도록 설정한 경우 가상 디바이스의 NUMA 및 vNUMA 토폴로지를 노출하는 새로운 가상 마더보드 레이아웃도 포함됩니다.

참고 향상된 가상 토폴로지는 ESXi 8.0에서만 사용할 수 있습니다. 이 기능을 사용하려면 VM에 하드웨어 버전 20 이상이 있어야 합니다.

절차

- 1 VM 토폴로지를 수동으로 구성하려면 먼저 VM을 찾습니다.
- 2 **VM 옵션**을 선택합니다. **CPU 토폴로지**에서 **소켓당 코어 수** 및 **NUMA 노드**를 조정할 수 있습니다.

무중단 추가된 CPU를 새 NUMA 노드에 사용하도록 설정하려면 고급 구성 옵션 아래에 `numa.allowHotadd`를 추가합니다. 그런 다음 NUMA 구성을 수동으로 추가할 수 있습니다.

참고 기본적으로 CPU 핫 플러그가 사용되도록 설정된 VM은 단일 NUMA 노드 토폴로지를 적용합니다. 무중단 추가된 CPU는 단일 NUMA 노드로 이동합니다.

- 3 **디바이스 할당**에서 디바이스를 가상 NUMA 노드에 할당하거나 할당되지 않은 상태로 둘 수도 있습니다.

결과

새로 구성된 이 토폴로지는 기존 VM 토폴로지 섹션에 **수동**으로 표시됩니다. 수동 구성이 없으면 이 탭은 **전원을 켤 때 할당됨**으로 표시됩니다.

가상 NUMA 제어

메모리 소모가 과도하게 많은 가상 시스템의 경우 고급 옵션을 사용하여 기본 가상 CPU 설정을 재정의할 수 있습니다.

이러한 고급 옵션을 가상 시스템 구성 파일에 추가할 수 있습니다.

표 21-1. 가상 NUMA 제어를 위한 고급 옵션

옵션	설명	기본값
<code>cpuid.coresPerSocket</code>	가상 CPU 소켓당 가상 코어의 수를 결정합니다. <code>numa.vcpu.followcorespersocket</code> 이 구성되어 있는 경우가 아니면 이 옵션은 가상 NUMA 토폴로지에 영향을 미치지 않습니다. 참고 8.0에서는 VM에 대한 최적의 <code>coresPerSocket</code> 이 자동으로 선택되며 ESXi 기본값은 0으로 표시됩니다.	1
<code>numa.vcpu.maxPerVirtualNode</code>	이 값을 약수로 사용하여 총 vCPU 수를 고르게 나눠서 가상 NUMA 노드의 수를 결정합니다.	8
<code>numa.autosize.once</code>	이 설정으로 가상 시스템 템플릿을 생성하는 경우 이후에 기본값 TRUE를 사용하여 가상 시스템 전원을 켤 때마다 해당 설정이 동일하게 유지됩니다. 값이 FALSE로 설정되어 있으면 전원이 켜질 때마다 가상 NUMA 토폴로지가 업데이트됩니다. 가상 시스템에 구성된 가상 CPU의 수가 언제든지 수정되면 가상 NUMA 토폴로지가 재평가됩니다.	FALSE
<code>numa.vcpu.min</code>	가상 NUMA 토폴로지를 생성하기 위해 가상 시스템에 포함되어야 하는 가상 CPU의 최소 개수입니다. 크기가 <code>numa.vcpu.min</code> 보다 작으면 가상 시스템은 항상 UMA입니다.	9
<code>numa.vcpu.followcorespersocket</code>	1로 설정되면, <code>cpuid.coresPerSocket</code> 과 연관된 가상 NUMA 노드 크기 조정의 이전 동작으로 복구됩니다.	0
<code>numa.allowHotadd</code>	새 NUMA 노드에 CPU를 무중단 추가하는 기능을 활성화하려면 고급 구성 옵션 아래에 <code>numa.allowHotadd</code> 를 추가합니다. 그런 다음 CPU 무중단 추가가 활성화되어 있는 동안 NUMA 구성을 수동으로 추가할 수 있습니다.	FALSE
<code>numa.vcpu.coresPerNode</code>	가상 NUMA 노드의 크기를 구성하는 VMX 매개 변수이며, UI 재구성에서 해석됩니다. 이 매개 변수는 HWv20에만 유효합니다. 기본값은 0으로, 이것은 ESXi에서 vNUMA 크기가 자동으로 선택됨을 나타냅니다. 참고 이 옵션이 <code>numa.vcpu.maxPerVirtualNode</code> 와 충돌하면 VM의 전원을 켤 수 없습니다.	0
<code>vcpu.hotadd</code>	이 옵션이 TRUE이면 가상 NUMA가 비활성화됩니다. VM이 CPU 핫 플러그를 사용할 수 있는 경우 VM에는 항상 하나의 가상 NUMA 노드가 표시됩니다.	
<code>llc.multiLLCPerSocket</code>	이 옵션이 TRUE인 경우 VM은 AMD Epyc에서 진정한 vLLC를 노출합니다. 노출된 vLLC 크기는 가상 소켓 크기와 다를 수 있습니다.	FALSE

표 21-1. 가상 NUMA 제어를 위한 고급 옵션 (계속)

옵션	설명	기본값
<code>llc.size.vcpu</code>	AMD Epyc에서 vLLC에 대해 수동으로 구성된 VCPU 수입니다. VM의 다른 설정과 호환되지 않는 값은 무시됩니다.	
<code>chipset.motherboardLayout</code>	이 VM에서 사용하는 가상 마더보드의 유형입니다. 다음 두 값 중 하나만 가질 수 있습니다. acpi: HWv 20의 새 마더보드 레이아웃. i440bx: 레거시 마더보드 레이아웃	
<code>cpuid.coresPerSocket.cookie</code>	자동으로 생성된 <code>coresPerSocket</code> 값을 저장하기 위해 ESXi에서 생성된 <code>vmx</code> 항목입니다. vMotion 일관성을 보장하기 위해 존 재합니다. 수동으로 변경하거나 제거하지 마십시오.	

NUMA 제어 지정

많은 메모리를 사용하는 애플리케이션이나 소수의 가상 시스템이 있는 경우 가상 시스템 CPU 및 메모리 배치를 명시적으로 지정하여 성능을 최적화할 수 있습니다.

가상 시스템에서 대량의 데이터 집합을 사용하는 과학적 계산 애플리케이션이나 메모리 내 데이터베이스 같이 메모리 소모가 많은 워크로드를 실행하는 경우 제어 기능을 지정하면 유용합니다. 시스템 워크로드가 단순하고 변화가 없는 것으로 알려진 경우 NUMA 배치를 수동으로 최적화할 수도 있습니다. 예를 들어 워크로드가 비슷한 8개의 가상 시스템을 실행하는 8중 프로세서 시스템은 명시적으로 최적화하기가 쉽습니다.

참고 대부분의 경우 ESXi 호스트의 자동 NUMA 최적화를 수행하면 성능이 향상됩니다.

ESXi에서는 NUMA 배치를 위한 세 가지 제어 기능 집합이 제공되므로 관리자가 가상 시스템의 메모리 및 프로세서 배치를 제어할 수 있습니다.

다음 옵션을 지정할 수 있습니다.

NUMA 노드 선호도

이 옵션을 설정하면 NUMA가 선호도에 지정된 노드에서만 가상 시스템을 스케줄링할 수 있습니다.

CPU 선호도

이 옵션을 설정하면 가상 시스템이 선호도에 지정된 프로세서만 사용합니다.

메모리 선호도

이 옵션을 설정하면 서버가 지정된 노드에만 메모리를 할당합니다.

NUMA 노드 선호도를 지정해도 가상 시스템은 여전히 NUMA를 통해 관리되지만 가상 시스템의 가상 CPU는 NUMA 노드 선호도에 지정된 노드에서만 스케줄링할 수 있습니다. 마찬가지로 NUMA 노드 선호도에 지정된 노드에서만 메모리를 가져올 수 있습니다. CPU 또는 메모리 선호도를 지정하면 가상 시스템이 더 이상 NUMA를 통해 관리되지 않습니다. CPU 및 메모리 선호도 제약 조건을 제거하면 이러한 가상 시스템의 NUMA 관리 기능을 사용할 수 있습니다.

수동 NUMA 배치를 수행할 경우 시스템 간에 프로세서 리소스를 고르게 분산하는 ESXi 리소스 관리 알고리즘에 방해가 될 수 있습니다. 예를 들어 워크로드의 프로세서 소모가 많은 가상 시스템 10개를 하나의 노드에 수동으로 배치하고 다른 노드에 2개의 가상 시스템을 수동으로 배치할 경우 시스템에서 12개의 가상 시스템 모두에 시스템 리소스를 균등하게 배분할 수 없게 됩니다.

특정 프로세서에 가상 시스템 연결

가상 CPU를 고정 프로세서에 고정하여 가상 시스템에서 실행되는 애플리케이션의 성능을 향상할 수 있습니다. 이렇게 하면 가상 CPU가 NUMA 노드 간에 마이그레이션되지 않도록 할 수 있습니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
 - a 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - b **VM** 탭을 클릭합니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭한 후 **설정 편집**을 클릭합니다.
- 3 **가상 하드웨어** 탭을 선택하고 **CPU**를 확장합니다.
- 4 스케줄링 선호도에서 CPU 선호도를 원하는 프로세서로 설정합니다.

참고 NUMA 노드의 모든 프로세서를 수동으로 선택해야 합니다. CPU 선호도는 기본적으로 노드 단위가 아니라 프로세서 단위로 지정됩니다.

메모리 선호도를 사용하여 특정 NUMA 노드와 메모리 할당 연결

가상 시스템의 향후 모든 메모리 할당이 특정 NUMA 노드와 연결된 페이지를 사용하도록 지정할 수 있습니다. 이를 수동 메모리 선호도라고도 합니다.

참고 CPU 선호도를 지정한 경우에 한해 향후 메모리 할당용으로 사용할 노드를 지정합니다. 메모리 선호도 설정에만 수동 변경을 시행하면 NUMA의 자동 재조정은 올바르게 작동하지 않습니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
- 2 **구성** 탭을 클릭합니다.
- 3 **설정**을 클릭하고 **VM 하드웨어**를 클릭합니다.
- 4 **편집**을 클릭합니다.
- 5 **가상 하드웨어** 탭을 선택하고 **메모리**를 확장합니다.
- 6 NUMA 메모리 선호도에서 메모리 선호도를 설정합니다.

예제: 단일 NUMA 노드에 가상 시스템 바인딩

다음 예제에서는 8방향 서버의 양방향 가상 시스템에 대한 단일 NUMA 노드에 마지막 네 개의 물리적 CPU를 수동으로 바인딩하는 방법을 보여 줍니다.

CPU(예: 4, 5, 6 및 7)는 물리적 CPU 번호입니다.

- 1 vSphere Client에서 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
- 2 **고급 매개 변수** 탭을 클릭합니다.
- 3 vSphere Client에서 프로세서 4, 5, 6 및 7에 대한 CPU 선호도를 설정합니다.

그런 다음 이 가상 시스템을 노드 1에서만 실행하려고 합니다.

- 1 vSphere Client 인벤토리 패널에서 가상 시스템을 선택하고 **설정 편집**을 선택합니다.
- 2 **고급 매개 변수** 탭을 클릭합니다.
- 3 vSphere Client에서 NUMA 노드에 대한 메모리 선호도를 1로 설정합니다.

이 두 작업을 완료하면 가상 시스템이 NUMA 노드 1에서만 실행되고 가능한 경우 동일한 노드에서 메모리가 할당됩니다.

가상 시스템을 지정된 NUMA 노드에 연결

NUMA 노드를 가상 시스템에 연결하여 NUMA 노드 선호도를 지정하면 ESXi에서 가상 시스템의 가상 CPU 및 메모리를 스케줄링할 수 있는 NUMA 노드 집합이 제한됩니다.

참고 NUMA 노드 선호도를 제한하면 ESXi NUMA 스케줄러가 공정성을 위해 가상 시스템을 여러 NUMA 노드에서 재조정하는 기능이 방해될 수 있습니다. NUMA 노드 선호도는 재조정 문제를 고려한 후에만 지정하십시오.

절차

- 1 vSphere Client에서 VM을 찾습니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
- 3 **고급 매개 변수** 탭을 클릭합니다.
- 4 (선택 사항) 매개 변수를 추가하려면 매개 변수의 특성 이름과 값을 입력하고 **추가**를 클릭합니다.
- 5 (선택 사항) 매개 변수를 변경하려면 해당 매개 변수의 **값** 텍스트 상자에 새 값을 입력하고 확인 표시를 클릭하여 확인합니다.
- 6
 - 가상 시스템에 NUMA 노드를 지정하려면 이름 옆에 `numa.nodeAffinity`를 입력합니다.
 - 가상 시스템의 특정 가상 NUMA 노드에 대해 NUMA 노드를 지정하려면 이름 옆에 `sched.nodeX.affinity`를 입력합니다. 여기서 X는 가상 NUMA 노드 번호입니다. 예를 들어 `sched.node0.affinity`는 가상 시스템에 가상 NUMA 노드 0을 지정합니다.

- 7 값 열에서 가상 시스템 또는 가상 NUMA 노드를 스케줄링할 수 있는 NUMA 노드를 입력합니다.
노드를 여러 개 입력하는 경우 쉼표로 구분된 목록을 사용하십시오. 예를 들어 가상 시스템 리소스 스케줄링을 NUMA 노드 0 및 1로 제한하려면 0,1을 입력합니다.
- 8 **확인**을 클릭합니다.

리소스 관리를 사용자 지정하는 데 도움이 되도록 호스트 또는 개별 가상 시스템에 대한 고급 특성을 설정할 수 있습니다.

대부분의 경우 기본 리소스 할당 설정(예약, 제한, 공유)을 조정하거나 기본 설정을 수락하면 적절한 리소스 할당이 이루어집니다. 하지만 고급 특성을 사용하여 호스트 또는 특정 가상 시스템에 대한 리소스 관리를 사용자 지정할 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- 고급 호스트 특성 설정
- 고급 가상 시스템 특성 설정
- 지연 시간 감도
- VM에 대한 가상 하이퍼스레딩 지원
- vHT 전체 CPU 예약
- VM에 대해 vHT 활성화
- 신뢰할 수 있는 메모리
- 게스트 vRAM에 1GB 페이지 사용

고급 호스트 특성 설정

호스트에 대한 고급 특성을 설정할 수 있습니다.

경고 고급 옵션 변경은 지원되지 않습니다. 일반적으로 기본 설정으로 최적의 결과를 얻을 수 있습니다. 고급 옵션은 VMware 기술 지원 또는 기술 자료 문서의 특정 지침이 있을 때에만 변경하십시오.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 구성 탭을 클릭합니다.
- 3 시스템에서 **고급 시스템 설정**을 클릭합니다.
- 4 편집 버튼을 클릭합니다.

5 적절한 항목을 찾고 값을 변경합니다.

6 확인을 클릭합니다.

고급 메모리 특성

고급 메모리 특성을 사용하여 메모리 리소스 사용을 사용자 지정할 수 있습니다.

표 22-1. 고급 메모리 특성

특성	설명	기본값
Mem.ShareForceSalting	Mem.ShareForceSalting 0: 가상 시스템 간 TPS(투명 페이지 공유) 동작이 계속 유지됩니다. VMX 옵션 <code>sched.mem.pshare.salt</code> 의 값은 있어도 무시됩니다. Mem.ShareForceSalting 1: 기본적으로 솔팅 값은 <code>sched.mem.pshare.salt</code> 에서 가져옵니다. 값을 지정하지 않으면 가상 시스템의 솔팅 값이 0으로 간주되어 이전 TPS(VM 간) 동작으로 돌아갑니다. Mem.ShareForceSalting 2: 기본적으로 솔팅 값은 <code>sched.mem.pshare.salt</code> (있는 경우) 또는 <code>vc.uuid</code> 에서 가져옵니다. 값이 없으면 페이지 공유 알고리즘이 가상 시스템별로 임의의 고유한 솔팅 값을 생성하며, 이 값은 사용자가 구성할 수 없습니다.	2
Mem.SamplePeriod	가상 시스템의 실행 시간 중 작업 집합 크기를 예측하기 위해 메모리 작업을 모니터링할 시간 간격(초)을 지정합니다.	60
Mem.BalancePeriod	자동 메모리 재할당의 시간 간격(초)을 지정합니다. 사용 가능한 메모리 양이 크게 변경된 경우에도 재할당이 트리거됩니다.	15
Mem.IdleTax	유휴 메모리 세율을 백분율로 지정합니다. 이 세율은 실제로 가상 시스템에서 현재 사용 중인 메모리보다 유휴 메모리에 더 높게 적용됩니다. 세율 0%는 작업 집합을 무시하고 공유에 따라 메모리를 엄격하게 할당하는 할당 정책을 정의합니다. 세율이 높으면 유휴 메모리를 비생산적으로 비축하고 있는 가상 시스템에서 해당 메모리를 회수하여 재할당할 수 있는 할당 정책이 만들어집니다.	75
Mem.ShareScanGHZ	사용 가능한 호스트 CPU 리소스의 각 GHz에 대해 페이지 공유 가능성을 검사할 초당 최대 메모리 페이지 크기를 지정합니다. 예를 들어 기본값은 1GHz마다 초당 4MB입니다.	4
Mem.ShareScanTime	전체 가상 시스템에서 페이지 공유 가능성을 검사할 시간(분)을 지정합니다. 기본값은 60분입니다.	60
Mem.CtlMaxPercent	구성된 메모리 크기의 백분율에 기반해 메모리 벌룬 드라이버(<code>vmmemctl</code>)를 사용하여 가상 시스템에서 회수되는 최대 메모리 양을 제한합니다. 모든 가상 시스템에 대해 회수를 비활성화하려면 0을 지정합니다.	65
Mem.AllocGuestLargePage	호스트의 큰 페이지로 게스트의 큰 페이지 지원을 활성화합니다. 이렇게 하면 게스트의 큰 페이지를 사용하는 서버 워크로드에서 TLB 누락을 줄이고 성능을 향상할 수 있습니다. 0 = 비활성화.	1
Mem.AllocUsePSharePool 및 Mem.AllocUseGuestPool	호스트의 큰 페이지로 게스트의 큰 페이지를 지원할 수 있는 확률을 높여 메모리 조각화를 줄입니다. 호스트 메모리가 조각화되면 호스트의 큰 페이지는 가용성이 줄어듭니다. 0 = 비활성화.	15
Mem.MemZipEnable	호스트에 대한 메모리 압축을 활성화합니다. 0 = 비활성화.	1

표 22-1. 고급 메모리 특성 (계속)

특성	설명	기본값
Mem.MemZipMaxPct	압축된 메모리로 저장할 수 있는 각 가상 시스템 메모리의 최대 백분율을 기준으로 압축 캐시의 최대 크기를 지정합니다.	10
LPage.LPageDefragEnable	큰 페이지 조각 모음을 활성화합니다. 0 = 비활성화.	1
LPage.LPageDefragRateVM	각 가상 시스템에서 큰 페이지 조각 모음을 시도할 초당 최대 횟수입니다. 허용되는 값 범위는 1~1024입니다.	32
LPage.LPageDefragRateTotal	큰 페이지 조각 모음을 시도할 초당 최대 횟수입니다. 허용되는 값 범위는 1~10240입니다.	256
LPage.LPageAlwaysTryForNPT	중첩된 페이지 테이블(AMD의 경우 'RVI', Intel의 경우 'EPT')에 큰 페이지를 할당하려고 시도합니다. 이 옵션을 활성화하면 중첩된 페이지 테이블을 사용하는 시스템(예: AMD Barcelona)에서 모든 게스트 메모리가 큰 페이지를 사용하여 지원됩니다. NPT를 사용할 수 없는 경우에는 게스트 메모리의 일부만 큰 페이지로 지원됩니다. 0 = 비활성화.	1

고급 NUMA 특성

고급 NUMA 특성을 사용하여 NUMA 사용을 사용자 지정할 수 있습니다.

표 22-2. 고급 NUMA 특성

특성	설명	기본값
Numa.RebalancePeriod	재조정 기간의 빈도를 밀리초 단위로 제어합니다. 재조정 빈도가 높으면 특히 많은 수의 가상 시스템을 실행하는 시스템에서 CPU 오버헤드가 증가할 수 있습니다. 또한 재조정 빈도가 높으면 공정성을 향상시킬 수 있습니다.	2000
Numa.MigImbalanceThreshold	NUMA 리밸런서는 각 가상 시스템의 허용되는 CPU 시간과 실제 소비량의 차이를 고려하여 노드 간의 CPU 불균형을 계산합니다. 이 옵션은 가상 시스템 마이그레이션을 트리거하는 데 필요한 노드 간 최소 로드 불균형을 백분율 단위로 제어합니다.	10
Numa.RebalanceEnable	NUMA 재조정 및 스케줄링을 활성화합니다. 가상 시스템의 NUMA 재조정 및 초기 배치를 모두 비활성화하여 사실상 NUMA 스케줄링 시스템을 비활성화하려면 이 옵션을 0으로 설정합니다.	1
Numa.RebalanceCoresTotal	호스트에서 NUMA 리밸런서를 활성화하는 데 필요한 총 프로세서 코어의 최소 개수를 지정합니다.	4
Numa.RebalanceCoresNode	NUMA 리밸런서를 활성화하는 데 필요한 노드당 프로세서 코어의 최소 개수를 지정합니다. 이 옵션과 Numa.RebalanceCoresTotal은 총 프로세서 또는 노드당 프로세서 수가 적어서 NUMA 재조정을 활성화할 경우 스케줄링 공정성이 저하될 수 있는 양방향 Opteron 호스트 같은 소규모 NUMA 구성에서 NUMA 재조정을 비활성화할 때 유용합니다.	2

표 22-2. 고급 NUMA 특성 (계속)

특성	설명	기본값
Numa.AutoMemAffinity	CPU 선호도가 설정된 가상 시스템의 메모리 선호도를 자동으로 설정합니다.	1
Numa.PageMigEnable	NUMA 노드 간에 페이지를 자동으로 마이그레이션하여 메모리 인접성을 향상시킵니다. 수동으로 설정한 페이지 마이그레이션 속도는 여전히 적용됩니다.	1

고급 가상 시스템 특성 설정

가상 시스템에 대한 고급 특성을 설정할 수 있습니다.

사전 요구 사항

고급 특성을 설정하기 전에 VM의 전원을 끕니다.

절차

- vSphere Client에서 가상 시스템을 찾습니다.
 - 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - VM 탭을 클릭합니다.
- 가상 시스템을 마우스 오른쪽 버튼으로 클릭하고 **설정 편집**을 선택합니다.
- 고급 매개 변수** 탭을 클릭합니다.
- (선택 사항) 매개 변수를 추가하려면 매개 변수의 특성 이름과 값을 입력하고 **추가**를 클릭합니다.
- (선택 사항) 매개 변수를 변경하려면 해당 매개 변수의 **값** 텍스트 상자에 새 값을 입력하고 확인 표시를 클릭하여 확인합니다.
- 확인**을 클릭합니다.

고급 가상 시스템 특성

고급 가상 시스템 특성을 사용하여 가상 시스템 구성을 사용자 지정할 수 있습니다.

표 22-3. 고급 가상 시스템 특성

특성	설명	기본값
sched.mem.maxmemctl	벌루닝을 통해 선택된 가상 시스템에서 회수되는 최대 메모리 양(MB)입니다. ESXi 호스트에서 추가 메모리를 회수해야 하는 경우에는 강제로 스와핑이 적용됩니다. 스와핑보다는 벌루닝을 사용하는 것이 좋습니다.	-1(제한 없음)
sched.mem.pshare.enable	선택된 가상 시스템에 대해 메모리 공유를 사용하도록 설정합니다. 이 부울 값은 기본적으로 True로 설정됩니다. 가상 시스템에 대해 이 값을 False로 설정하면 메모리 공유가 해제됩니다.	참

표 22-3. 고급 가상 시스템 특성 (계속)

특성	설명	기본값
sched.mem.pshare.salt	솔트 값은 각 가상 시스템에 대해 구성 가능한 VMX 옵션입니다. 가상 시스템의 VMX 파일에 이 옵션이 없으면 <code>vc.uuid vmx</code> 옵션의 값이 기본값으로 사용됩니다. <code>vc.uuid</code> 는 각 가상 시스템마다 고유하기 때문에 기본적으로 투명 페이지 공유는 특정 가상 시스템(VM 내)에 속해 있는 페이지 사이에서만 적용됩니다. 가상 시스템 그룹을 신뢰할 수 있는 경우에는 해당하는 모든 가상 시스템(VM 간)에 대해 공통적인 솔트 값을 설정하여 VM끼리 페이지를 공유할 수 있습니다.	사용자 구성 가능
sched.swap.persist	가상 시스템의 전원이 꺼질 때 가상 시스템의 스왑 파일을 유지할지 삭제할지를 지정합니다. 기본적으로 시스템에서는 가상 시스템의 전원이 켜질 때 가상 시스템에 대한 스왑 파일을 만들고 가상 시스템의 전원이 꺼질 때 스왑 파일을 삭제합니다.	False
sched.swap.dir	가상 시스템의 스왑 파일이 저장되는 디렉토리 위치입니다. 기본적으로 가상 시스템의 작업 디렉토리, 즉 구성 파일이 들어 있는 디렉토리로 설정됩니다. 이 디렉토리는 가상 시스템에 액세스할 수 있는 호스트에 있어야 합니다. 가상 시스템 또는 가상 시스템에서 만들어진 복제본을 이동하는 경우에는 이 특성을 다시 설정해야 할 수 있습니다.	<code>workingDir</code> 와 같음

고급 가상 NUMA 특성

고급 가상 NUMA 특성을 사용하여 가상 NUMA 사용량을 사용자 지정할 수 있습니다.

표 22-4. 고급 NUMA 특성

특성	설명	기본값
<code>cpuid.coresPerSocket</code>	가상 CPU 소켓당 가상 코어의 수를 결정합니다. 값이 1보다 크면 가상 시스템에 가상 NUMA 토폴로지가 있는 경우 가상 NUMA 노드의 크기도 결정합니다. 각 물리적 호스트에 대한 정확한 가상 NUMA 토폴로지를 아는 경우 이 옵션을 설정할 수 있습니다.	1
<code>numa.autosize</code>	이 옵션을 설정하면 가상 NUMA 토폴로지서 가상 노드당 가상 CPU의 수는 각 물리적 노드의 코어 수와 같습니다.	FALSE
<code>numa.autosize.once</code>	이 설정으로 가상 시스템 템플릿을 만들면 이후에 가상 시스템 전원을 켤 때마다 설정이 항상 동일하게 유지됩니다. 가상 시스템의 구성된 가상 CPU의 수가 수정되면 가상 NUMA 토폴로지가 재평가됩니다.	TRUE
<code>numa.vcpu.maxPerVirtualNode</code>	<code>cpuid.coresPerSocket</code> 이 2의 배수로 너무 제한된 경우에는 <code>numa.vcpu.maxPerVirtualNode</code> 를 직접 설정할 수 있습니다. 이 경우에는 <code>cpuid.coresPerSocket</code> 을 설정하지 마십시오.	8
<code>numa.vcpu.min</code>	가상 NUMA 토폴로지를 생성하기 위해 필요한 가상 시스템의 최소 가상 CPU 수입니다.	9

표 22-4. 고급 NUMA 특성 (계속)

특성	설명	기본값
numa.vcpu.maxPerMachineNode	NUMA 노드에서 동시에 스케줄링할 수 있는 동일한 가상 시스템에 속하는 최대 가상 CPU 수입니다. 이 특성을 사용하면 NUMA 노드마다 서로 다른 NUMA 클라이언트를 적용하여 최대 대역폭을 사용할 수 있습니다.	가상 시스템이 실행되고 있는 물리적 호스트에서 노드당 코어의 수입니다.
numa.vcpu.maxPerClient	NUMA 클라이언트에서 최대 가상 CPU 수입니다. 클라이언트는 NUMA에서 하나의 엔터티로 관리되는 가상 CPU 그룹입니다. 기본적으로 각 가상 NUMA 노드가 NUMA 클라이언트이지만 가상 NUMA 노드가 물리적 NUMA 노드보다 큰 경우에는 하나의 가상 NUMA 노드를 여러 NUMA 클라이언트에서 지원할 수 있습니다.	같음: numa.vcpu.maxPerMachineNode
numa.nodeAffinity	가상 시스템의 가상 CPU 및 메모리를 스케줄링할 수 있는 NUMA 노드 집합을 제한합니다. 참고 NUMA 노드 선호도를 제한하면 NUMA 스케줄러가 공정성을 위해 가상 시스템을 여러 NUMA 노드에서 재조정하는 기능이 방해될 수 있습니다. NUMA 노드 선호도는 재조정 문제를 고려한 후에만 지정하십시오.	
numa.mem.interleave	노드를 구성하는 NUMA 클라이언트가 실행 중이고 가상 NUMA 토폴로지가 노출되지 않은 모든 NUMA 노드에서 가상 시스템에 할당된 메모리가 정적으로 인터리브되는지 여부를 지정합니다.	True

지연 시간 감도

가상 시스템의 지연 시간 감도를 조정하여 지연 시간에 민감한 애플리케이션의 스케줄링 지연을 최적화할 수 있습니다.

ESXi는 높은 처리량을 제공하도록 최적화되어 있습니다. 지연 시간에 민감한 애플리케이션의 낮은 지연 시간 요구 사항을 충족하도록 가상 시스템을 최적화할 수 있습니다. 지연 시간에 민감한 애플리케이션의 예로는 VOIP 또는 미디어 플레이어 애플리케이션, 마우스 또는 키보드 디바이스에 자주 액세스해야 하는 애플리케이션 등이 있습니다.

지연 시간 감도 조정

가상 시스템의 지연 시간 감도를 조정할 수 있습니다.

사전 요구 사항

지연 시간 감도가 **높음**으로 설정된 경우 ESXi에서 하드웨어 버전 14를 사용하는 VM의 전원을 켜려면 전체 CPU 예약이 필요합니다.

절차

- 1 vSphere Client에서 가상 시스템을 찾습니다.
 - a 가상 시스템을 찾으려면 데이터 센터, 폴더, 클러스터, 리소스 풀 또는 호스트를 선택합니다.
 - b **VM** 탭을 클릭합니다.
- 2 가상 시스템을 마우스 오른쪽 버튼으로 클릭한 후 **설정 편집**을 클릭합니다.
- 3 **VM 옵션**을 클릭하고 **고급**을 클릭합니다.
- 4 **지연 시간 감도** 드롭다운 메뉴에서 설정을 선택합니다.
- 5 **확인**을 클릭합니다.

VM에 대한 가상 하이퍼스레딩 지원

vHT(가상 하이퍼스레딩)는 VM에 대해 지원됩니다.

vSphere 8.0에서는 VM에 대해 vHT가 지원됩니다. vHT는 기본적으로 비활성화되어 있으며 각 VM에 대한 지연 시간 감도 설정에서 활성화할 수 있습니다. vHT는 최대 HT 크기 2를 지원합니다.

vHT는 지연 시간 감도가 높은 기능의 확장입니다. 하이퍼스레딩 인식의 이점을 활용하는 애플리케이션은 높은 지연 시간 감도와 vHT 활성화를 통해 성능 향상을 볼 수 있습니다. 충분한 리소스 예약과 VM에 전용 물리적 CPU가 있기 때문에 성능이 향상될 수 있습니다.

ESXi에 vHT가 활성화되어 있지 않으면 각 vCPU(가상 CPU)는 게스트 운영 체제에서 사용할 수 있는 하이퍼스레딩되지 않은 단일 코어와 동일합니다. vHT가 활성화되면 각 게스트 vCPU는 vCore(가상 코어)의 단일 하이퍼스레드로 처리됩니다.

동일한 vCore의 가상 하이퍼스레드는 동일한 물리적 코어를 차지합니다. 따라서 VM의 vCPU는 vHT가 비활성화된 지연 시간 감도가 높은 VM에서 여러 코어를 사용하는 것과는 대조적으로 동일한 코어를 공유할 수 있습니다.

이전 하드웨어 버전을 실행하는 ESXi 호스트 및 VM은 이 기능을 사용할 수 없습니다.

vHT 전체 CPU 예약

공식을 사용하여 vHT에 대한 전체 CPU 예약을 계산할 수 있습니다.

vHT가 없는 낮은 지연 시간 VM의 경우 VM의 각 vCPU에는 물리적 코어의 스레드에 대한 배타적인 선호도가 있습니다. 하이퍼스레딩이 활성화된 호스트의 경우 파트너 하이퍼스레드에는 유휴 상태의 월드에 대한 배타적인 선호도가 있습니다. 낮은 지연 시간 VM의 각 vCPU에는 전용 물리적 코어가 할당됩니다.

낮은 지연 시간 VM에 대한 CPU 예약은 다음과 같이 계산됩니다.

낮은 지연 시간 VM(vHT 없음) CPU 최소 예약 = numVcpus * cpuFrequency

그러나 VM에 대해 vHT가 활성화되면 물리적 코어의 각 하이퍼트윈은 각 하이퍼트윈이 VM의 vCPU에 대한 배타적인 선호도를 갖는 VM의 여러 vCPU 간에 공유됩니다. 즉 numSMT 물리적 하이퍼스레드가 있는 코어는 numSMT 가상 스레드에서 공유됩니다. 이 경우 CPU 예약 요구 사항은 다음과 같이 계산됩니다.

$$\text{낮은 지연 시간 VM (vHT 포함) CPU 최소 예약} = (\text{numVcpus} / \text{numSMT}) * \text{cpuFrequency}$$

표 22-5. CPU 주파수가 2GHz인 호스트에서 vCPU가 20개인 낮은 지연 시간 VM을 부팅하는 예

	numSMT = 1(vHT 없음)	numSMT = 2(vHT 포함)
numVcpus	20	20
물리적 코어 수	20	10(2개의 vCPU에서 공유되는 각 코어)
필요한 최소 CPU 예약	20 * 2.0GHz = 40GHz	(20/2) * 2.0GHz = 20GHz

VM에 대해 vHT 활성화

vHT는 ESXi 8.0에서 지원되지만 기본적으로 비활성화되어 있습니다. 각 VM에 대한 지연 시간 감도 설정에서 vHT를 활성화할 수 있습니다.

사전 요구 사항

vHT가 활성화되면 CPU 및 메모리를 전체 예약으로 설정해야 합니다. 예약을 더 낮게 설정하면 주의가 표시됩니다.

절차

- 1 vSphere에서 VM을 선택합니다.
- 2 **작업**을 선택하고 **설정 편집**을 클릭합니다.
- 3 [지연 시간 감도]에서 드롭다운 메뉴를 클릭하고 **높음(하이퍼스레딩 사용)**을 선택합니다.
- 4 **확인**을 클릭합니다.

결과

vHT가 활성화되었습니다.

신뢰할 수 있는 메모리

ESXi는 신뢰할 수 있는 메모리를 지원합니다.

신뢰할 수 있는 메모리는 시스템의 메모리 중 다른 부분보다 하드웨어 메모리 오류가 발생할 가능성이 낮은 메모리 부분으로, 일부 시스템에는 이러한 메모리가 있습니다. 하드웨어가 다른 안정성 수준에 대한 정보를 제공할 경우 ESXi는 보다 높은 시스템 안정성을 확보할 수 있습니다.

신뢰할 수 있는 메모리 보기

라이센스가 신뢰할 수 있는 메모리를 허용하는지 여부를 확인할 수 있습니다.

절차

- 1 vSphere Client에서 호스트를 찾습니다.
- 2 구성 탭을 클릭하고 시스템 탭을 클릭합니다.
- 3 라이선싱을 선택합니다.
- 4 라이선스가 부여된 기능에서 신뢰할 수 있는 메모리가 표시되는지 확인합니다.

다음에 수행할 작업

ESXCLI `hardware memory get` 명령을 사용하여 신뢰할 수 있는 것으로 간주되는 메모리의 양을 확인할 수 있습니다.

게스트 vRAM에 1GB 페이지 사용

vSphere ESXi는 게스트 vRAM에 1GB 페이지를 사용하는 것을 제한적으로 지원합니다.

게스트 메모리에 1GB 페이지를 사용할 수 있도록 하려면 VM에

`sched.mem.lpage.enable1GPage = "TRUE"` 옵션을 적용해야 합니다. 이는 **설정 편집**을 선택한 후 [고급] 옵션에서 설정할 수 있습니다. 전원이 꺼져 있는 VM에만 1GB 페이지를 사용하도록 설정할 수 있습니다.

1GB 페이지가 사용되도록 설정된 VM에는 전체 메모리가 예약되어 있어야 합니다. 그렇지 않은 경우 VM의 전원을 켜지 못할 수 있습니다. 1GB 페이지가 사용되도록 설정된 VM은 전원을 켤 때 모든 vRAM이 사전 할당됩니다. 이러한 VM은 전체 메모리가 예약되어 있으므로 메모리 회수의 영향을 받지 않으며 VM의 전체 수명 동안 메모리 사용량이 최대 수준으로 유지됩니다.

1GB 페이지 vRAM은 경우에 따라 지원되며 가능한 최선의 범위에서 1GB 페이지가 할당됩니다. 호스트 CPU에 1GB 기능이 없는 경우 등을 예로 들 수 있습니다. 게스트 vRAM에 1GB 페이지가 지원될 가능성을 극대화하려면 새로 부팅된 호스트에서 1GB 페이지를 필요로 하는 VM을 시작하는 것이 좋습니다. 시간이 지남에 따라 호스트 RAM이 조각화되기 때문입니다.

1GB 페이지가 사용되도록 설정된 VM은 다른 호스트로 마이그레이션할 수 있습니다. 그러나 대상 호스트에서는 소스 호스트와 동일한 양으로 1GB 페이지 크기가 할당되지 않을 수 있습니다. 또한 소스 호스트에서 1GB 페이지가 지원된 vRAM 부분이 대상 호스트에서는 1GB 페이지가 더 이상 지원되지 않는 경우도 있습니다.

이 같은 경우에 따른 1GB 페이지 지원은 HA 및 DRS를 비롯한 vSphere 서비스에도 적용되어 1GB 페이지 vRAM 지원이 유지되지 않을 수 있습니다. 이러한 서비스는 대상 호스트의 1GB 기능을 인식하지 않으며 배치를 결정할 때 1GB 메모리 지원을 고려하지 않습니다.

DRS 장애는 DRS 작업(또는 수동 모드에서 이 작업의 권장 사항) 생성이 차단되는 원인을 표시합니다.

이 섹션 내에서 DRS 장애가 정의되어 있습니다.

참고 이 장에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- 가상 시스템이 고정되어 있음
- 가상 시스템이 다른 호스트와 호환되지 않음
- 다른 호스트로 이동할 때 VM/VM DRS 규칙 위반
- 호스트가 가상 시스템과 호환되지 않음
- 호스트의 가상 시스템에서 VM/VM DRS 규칙을 위반함
- 호스트에 가상 시스템을 위한 용량이 충분하지 않음
- 잘못된 상태의 호스트
- 호스트에 가상 시스템을 위한 물리적 CPU 수가 충분하지 않음
- 호스트에 각 가상 시스템 CPU를 위한 용량이 충분하지 않음
- 가상 시스템이 vMotion에 있음
- 클러스터에 활성화된 호스트 없음
- 불충분한 리소스
- HA용으로 구성된 페일오버 수준을 만족하기 위한 리소스가 충분하지 않음
- 호환되는 하드 선호도 호스트가 없음
- 호환되는 소프트 선호도 호스트가 없음
- 소프트 규칙 위반 수정 허용되지 않음
- 소프트 규칙 위반 수정 영향

가상 시스템이 고정되어 있음

가상 시스템에서 DRS가 비활성화되어 있기 때문에 DRS가 가상 시스템을 이동할 수 없는 경우 이런 장애가 발생합니다. 즉, 가상 시스템이 등록된 호스트에서 "고정"되어 있습니다.

가상 시스템이 다른 호스트와 호환되지 않음

DRS가 가상 시스템을 실행할 수 있는 호스트를 찾지 못하면 이 장애가 발생합니다.

호스트가 가상 시스템의 CPU나 메모리 리소스 요구를 충족할 수 없거나 현재 호스트가 해당 가상 시스템으로 요구한 스토리지 액세스 또는 호스트를 가지지 않는 경우에 이 문제가 발생할 수도 있습니다.

이 문제를 해결하려면 가상 시스템의 요구사항을 충족할 수 있는 호스트를 제공합니다.

다른 호스트로 이동할 때 VM/VM DRS 규칙 위반

이 장애는 동일한 호스트에서 실행되고 서로 선호도 규칙을 공유하는 둘 이상의 가상 시스템이 다른 호스트로 이동할 수 없는 경우에 발생합니다.

이 문제는 일부 가상 시스템은 현재 호스트에서 vMotion을 수행할 수 없기 때문에 발생할 수 있습니다. 예를 들어 그룹에 있는 가상 시스템 중 하나가 DRS 비활성화 상태입니다.

이를 방지하려면 그룹의 일부 가상 시스템이 vMotion을 수행할 수 없는 이유를 확인하십시오.

호스트가 가상 시스템과 호환되지 않음

DRS가 가상 시스템을 호스트로 마이그레이션하는 것으로 간주하지만 호스트가 기존 가상 시스템과 호환되지 않음을 발견했을 때 이 장애가 발생합니다.

가상 시스템에 필요한 네트워크 또는 스토리지 연결이 대상 호스트에 없는 경우, 이 문제가 발생할 수 있습니다. 또 다른 이유로는 대상 호스트가 현재 호스트와 완전히 다르고 따라서 호스트 사이의 vMotion이 지원되지 않은 상태로 사용하면 이 장애가 발생합니다.

이러한 문제를 방지하기 위해서 모든 호스트가 일정하게 구성되어 있고 vMotion이 호스트 간에 호환되도록 클러스터를 만듭니다.

또 다른 이유로는 가상 시스템 및 호스트간에 필요한 VM/호스트 DRS 규칙으로 인해 DRS가 특정 가상 시스템을 특정 호스트에 배치할 수 없게 되었다는 점입니다.

호스트의 가상 시스템에서 VM/VM DRS 규칙을 위반함

가상 시스템의 전원을 켜거나 vMotion을 시작하여 가상 시스템을 이동할 때 가상 시스템이 VM/VM DRS 규칙을 위반하면 이 장애가 발생합니다.

가상 시스템의 전원을 수동으로 켜거나 vMotion을 사용하여 가상 시스템을 이동할 수 있지만 vCenter Server는 자동으로 전원을 켜거나 이동할 수 없습니다.

호스트에 가상 시스템을 위한 용량이 충분하지 않음

호스트가 가상 시스템을 실행할 수 있는 충분한 CPU나 메모리 용량을 가지고 있지 않으면 이 장애가 발생합니다.

잘못된 상태의 호스트

DRS 작업이 필요한 경우, 호스트가 유지 보수 상태나 대기 상태로 전환하면 이 장애가 발생합니다.

이 장애를 해결하려면 대기 모드나 유지 보수 모드로 전환하려는 호스트 요청을 취소합니다.

호스트에 가상 시스템을 위한 물리적 CPU 수가 충분하지 않음

가상 시스템 내의 가상 CPU 수를 충분하게 지원할 수 있는 수량의 CPU(하이퍼스레드)가 하드웨어에 없는 경우 이 장애가 발생합니다.

호스트에 각 가상 시스템 CPU를 위한 용량이 충분하지 않음

호스트가 가상 시스템을 실행할 수 있는 충분한 CPU 용량을 가지고 있지 않으면 이 장애가 발생합니다.

가상 시스템이 vMotion에 있음

가상 시스템이 vMotion에 있기 때문에 DRS가 가상 시스템을 이동하지 못할 때 이 장애가 발생합니다.

클러스터에 활성화된 호스트 없음

가상 시스템이 이동하고 있는 상태에서 클러스터가 비유지 보수 상태에 있고 연결된 호스트를 포함하지 않으면 이 장애가 발생합니다.

예를 들어서 모든 호스트가 연결이 끊겼거나 유지 보수 모드에 있으면 이 문제가 발생할 수 있습니다.

불충분한 리소스

시도한 작업이 리소스 구성 정책과 충돌할 때 이 장애가 발생합니다.

예를 들어 리소스 풀에 할당된 것 보다 더 많은 양의 메모리가 전원 켜기 작업에 예약된 경우 이 장애가 발생할 수 있습니다.

리소스를 조정하여 메모리 양을 늘린 후 이 작업을 재시도합니다.

HA용으로 구성된 페일오버 수준을 만족하기 위한 리소스가 충분하지 않음

페일오버를 위해 예약한 메모리 리소스 또는 CPU의 HA 구성이 위반되거나 DRS 작업을 수행하기에 부족할 때 이 장애가 발생합니다.

이 장애는 다음과 같은 경우에 보고됩니다.

- 유지 보수나 대기 모드로 호스트를 전환하도록 요청받은 경우.
- 가상 시스템이 전원 켜기를 시도할 때 페일오버를 위반하는 경우.

호환되는 하드 선호도 호스트가 없음

필수적인 VM/호스트 DRS 선호도 또는 반선호도 규칙을 충족하는 가상 시스템에 사용될 수 있는 호스트가 없습니다.

호환되는 소프트 선호도 호스트가 없음

지정된 VM/호스트 DRS 선호도 또는 반선호도 규칙을 충족하는 가상 시스템에 사용할 수 있는 호스트가 없습니다.

소프트 규칙 위반 수정 허용되지 않음

DRS 마이그레이션 임계값은 필수적일 때에만 설정됩니다.

이를 통해, 비 필수 VM/호스트 DRS 선호도 규칙을 수정하는 DRS 작업을 생성할 수 없습니다.

소프트 규칙 위반 수정 영향

필수가 아닌 VM/호스트 DRS 선호도 규칙은 성능에 영향을 미치지 때문에 수정이 일어나지 않습니다.

이 정보는 특정 범주(클러스터, 호스트 및 가상 시스템 문제)의 vSphere® DRS(Distributed Resource Scheduler) 문제를 설명합니다.

참고 이 장에서 "메모리"는 물리적 RAM 또는 영구 메모리를 나타낼 수 있습니다.

다음으로 아래 항목을 읽으십시오.

- [클러스터 문제](#)
- [호스트 문제](#)
- [가상 시스템 문제](#)

클러스터 문제

클러스터 문제가 발생하면 DRS에서 성능이 최적화되지 않거나 장애를 보고하지 못하게 될 수 있습니다.

클러스터의 로드 불균형

클러스터에 리소스 로드 불균형 문제가 있습니다.

문제

클러스터는 가상 시스템의 고르지 않은 리소스 요구 및 동일하지 않은 호스트 용량으로 인해 불균형이 발생할 수 있습니다.

원인

다음과 같은 원인으로 인해 클러스터에서 로드 불균형이 발생할 수 있습니다.

- 마이그레이션 임계값이 너무 높습니다.
임계값이 높으면 클러스터에서 로드 불균형이 나타날 가능성이 높습니다.
- VM/VM 또는 VM/호스트 DRS 규칙이 가상 시스템의 이동을 제한합니다.
- DRS 기능이 하나 이상의 가상 시스템에서 비활성화되어 있습니다.
- 디바이스가 하나 이상의 가상 시스템에 마운트되어 있어 DRS가 가상 시스템을 이동하여 로드 균형을 조정할 수 없게 됩니다.

- 가상 시스템이 DRS가 해당 가상 시스템을 이동한 호스트와 호환되지 않습니다. 즉 클러스터에 있는 호스트 중 최소한 하나는 마이그레이션을 수행할 가상 시스템과 호환되지 않습니다. 예를 들어, 호스트 A의 CPU가 호스트 B의 CPU와 vMotion이 호환되지 않으면 호스트 A는 호스트 B에서 실행 중이며 전원이 켜져 있는 가상 시스템과 호환되지 않습니다.
- 현재 위치에서 가상 시스템을 실행하는 것보다 가상 시스템을 이동하는 경우에 가상 시스템 성능이 더 낮아집니다. 로드가 불안정하거나 가상 시스템을 이동함에 따라 얻는 이익과 비교하여 마이그레이션 비용이 높으면 이 문제가 발생할 수 있습니다.
- vMotion이 클러스터의 호스트에 대해 활성화되거나 설정되지 않았습니다.

해결책

로드 불균형을 초래하는 문제를 해결합니다.

클러스터가 노란색으로 표시됨

리소스 부족으로 인해 클러스터가 노란색으로 표시됩니다.

문제

클러스터가 모든 리소스 풀과 가상 시스템의 예약을 충족할 만큼의 충분한 리소스를 가지고 있지 않지만 모든 실행되는 가상 시스템의 예약을 충족할 수 있는 충분한 리소스를 가진 경우, DRS가 계속해서 실행되고 클러스터가 노란색으로 표시됩니다.

원인

클러스터에서 호스트 리소스가 제거된 경우(예: 호스트 장애의 경우) 클러스터가 노란색으로 표시될 수 있습니다.

해결책

호스트 리소스를 클러스터에 추가하거나 리소스 풀 예약을 줄입니다.

일치하지 않는 리소스 풀로 인해 클러스터가 빨간색으로 표시됨

DRS 클러스터가 잘못된 경우 빨간색으로 됩니다. 리소스 풀 트리가 내부적으로 일정하지 않을 경우 클러스터가 빨간색으로 표시될 수 있습니다.

문제

클러스터 리소스 풀 트리가 내부적으로 일정하지 않으면(예를 들어 하위 예약의 합계가 상위 풀의 비 확장 예약보다 클 때), 클러스터에 모든 실행되고 있는 가상 시스템의 예약을 만족시킬 수 있는 리소스가 불충분하게 되고, 따라서 클러스터가 빨간색으로 표시됩니다.

원인

이러한 문제는 vCenter Server를 사용할 수 없거나 가상 시스템이 페일오버 상태인 동안 리소스 풀 설정이 변경된 경우 발생할 수 있습니다.

해결책

관련 변경 사항을 되돌리거나 리소스 풀 설정을 수정합니다.

페일오버 용량 위반으로 인해 클러스터가 빨간색으로 표시됨

DRS 클러스터가 잘못된 경우 빨간색으로 됩니다. 페일오버 용량이 위반될 경우 클러스터가 빨간색으로 표시될 수 있습니다.

문제

클러스터는 호스트 실패의 경우 가상 시스템의 페일오버를 시도하지만, 페일오버 요구 사항이 적용되는 모든 가상 시스템을 페일오버하기에 충분한 리소스를 가지도록 보장되지 않습니다.

원인

페일오버 요구 사항을 더 이상 충족시킬 수 없는 너무나 많은 리소스를 HA 작동의 클러스터가 상실한 경우에는 메시지가 나타나고 클러스터 상태가 빨간색으로 바뀝니다.

해결책

클러스터 요약 페이지 상단에 있는 노란색 상자에서 구성 문제 목록을 검토하고 이 문제를 초래하는 문제를 해결합니다.

전체 클러스터 로드가 낮을 때는 호스트의 전원이 꺼지지 않음

전체 클러스터 로드가 낮을 때 호스트 전원이 꺼지지 않습니다.

문제

HA 페일오버 예약에 예비 용량이 필요하기 때문에 전체 클러스터 로드가 낮을 때 호스트의 전원이 꺼지지 않습니다.

원인

호스트는 다음의 이유 때문에 전원이 꺼지지 않을 수도 있습니다.

- `MinPoweredOn{Cpu|Memory}Capacity` 고급 옵션 설정이 충족되어야 합니다.
- 리소스 예약, VM/호스트 DRS 규칙, VM/VM DRS 규칙, DRS가 활성화되지 않았거나 사용 가능한 용량이 있는 호스트와 호환되지 않는 등의 이유로 가상 시스템을 더 적은 수의 호스트로 통합할 수 없습니다.
- 로드가 불안정합니다.
- DRS 마이그레이션 임계값이 가장 높게 설정되어 있고 필수적인 이동만 허용합니다.
- vMotion이 구성되어 있지 않기 때문에 실행할 수 없습니다.
- 전원이 꺼져 있을 수 있는 호스트에서 DPM이 비활성화되었습니다.
- 호스트가 다른 호스트로 이동하는 가상 시스템과 호환되지 않습니다.

- 호스트에 Wake On LAN, IPMI 또는 iLO 기술이 적용되어 있지 않습니다. 두 기술 중 하나가 적용되어 있어야 DPM이 호스트를 대기 모드로 전환할 수 있습니다.

해결책

전체 클러스터 로드가 낮을 때 호스트 전원이 꺼지지 않는 문제를 해결합니다.

전체 클러스터 로드가 높을 때 호스트의 전원이 꺼짐

전체 클러스터 로드가 높을 때 호스트의 전원이 꺼집니다.

문제

호스트나 가상 시스템 성능의 저하 없이 보다 적은 호스트로 가상 시스템을 실행할 수 있음을 DRS가 판단했습니다. 또한 DRS는 활용률이 높은 호스트로부터 전원이 꺼지도록 예약되어 있는 호스트로 가상 시스템을 이동하는 것이 제한되어 있습니다.

원인

전체 클러스터 로드가 너무 높습니다.

해결책

클러스터 로드를 줄이십시오.

DRS가 거의 또는 전혀 vMotion 마이그레이션을 수행하지 않음

DRS가 vMotion 마이그레이션을 거의 수행하지 않거나 전혀 수행하지 않습니다.

문제

DRS가 vMotion 마이그레이션을 수행하지 않습니다.

원인

DRS는 클러스터에 다음 문제 중 하나 이상이 있을 경우 vMotion 마이그레이션을 수행하지 않습니다.

- 클러스터에서 DRS가 비활성화되었습니다.
- 호스트가 공유 스토리지를 가지고 있지 않습니다.
- 클러스터의 호스트가 vMotion 네트워크를 포함하고 있지 않습니다.
- DRS가 수동이며 마이그레이션을 승인한 사람이 없습니다.

DRS는 클러스터에 다음 문제 중 하나 이상이 있을 경우 vMotion 마이그레이션을 거의 수행하지 않습니다.

- 로드가 불안정하거나 vMotion에 시간이 오래 걸리거나 또는 둘 다의 이유입니다. 이동이 적절하지 않습니다.
- DRS가 가상 시스템을 거의 또는 결코 마이그레이션하지 않습니다.
- DRS 마이그레이션 임계값이 너무 높게 설정되어 있습니다.

다음의 이유로 DRS가 가상 시스템을 이동합니다.

- 사용자가 호스트 제거를 요청하여 유지 보수 모드 또는 대기 모드로 전환되었습니다.
- VM/호스트 DRS 규칙 또는 VM/VM DRS 규칙
- 예약 위반
- 로드 불균형
- 전원 관리

해결책

DRS가 vMotion 마이그레이션을 수행하지 못하게 하는 문제를 해결합니다.

호스트 문제

호스트 문제는 DRS를 예상한 대로 실행되지 않게 할 수 있습니다.

DRS는 전체 클러스터 로드가 낮을 때에는 용량을 증가시키기 위해 호스트의 전원이 켜져 있도록 권장함

호스트는 클러스터에 더 많은 용량을 제공할 수 있도록 또는 오버로드된 호스트를 위해 전원이 켜져 있어야 합니다.

문제

DRS에서는 전체 클러스터 로드가 낮을 경우 용량 증가를 위해 호스트의 전원을 켜진 상태로 둘 것을 권장합니다.

원인

다음 사항을 권장합니다.

- 클러스터가 DRS-HA 클러스터입니다. 보다 많은 페일오버 용량을 제공하기 위해서 추가로 전원이 켜진 호스트가 필요합니다.
- 일부 호스트는 오버로드되어 있고 현재 전원이 켜진 호스트의 가상 시스템은 로드 균형을 위해 대기 모드에 있는 호스트를 이동 시킬 수 있습니다.
- `MinPoweredOn{Cpu|Memory}Capacity` 고급 옵션을 충족할 수 있는 용량이 필요합니다.

해결책

호스트 전원을 켭니다.

전체 클러스터 로드가 높음

전체 클러스터 로드가 높습니다.

문제

전체 클러스터 로드가 높을 때 DRS는 호스트의 전원을 켜지 않습니다.

원인

다음과 같은 원인으로 인해 DRS가 호스트의 전원을 켜지 못할 수 있습니다.

- VM/VM DRS 규칙 또는 VM/호스트 DRS 규칙은 가상 시스템이 이 호스트로 이동하는 것을 제한합니다.
- 가상 시스템이 현재 호스트에 고정되어 있기 때문에 DRS가 로드 밸런싱을 위해 이러한 가상 시스템을 대기 모드에 있는 호스트로 이동할 수 없습니다.
- DRS 또는 DPM은 수동 모드에 있고 권장 사항이 적용되지 않았습니다.
- 사용 빈도가 높은 호스트의 가상 시스템이 해당 호스트로 이동하지 않습니다.
- 사용자 설정으로 인해 또는 이전 작업에서 호스트가 대기 모드를 종료하지 못했기 때문에 해당 호스트에서 DPM이 비활성화되었습니다.

해결책

DRS가 호스트 전원을 켜지 못하게 하는 문제를 해결합니다.

전체 클러스터 로드가 낮음

전체 클러스터 로드가 낮습니다.

문제

전체 클러스터 로드가 낮을 때 DRS는 호스트 전원을 끄지 않습니다.

원인

다음과 같은 원인으로 인해 호스트에서 DRS의 전원을 끄지 못할 수 있습니다.

- DPM(분배 전원 관리)이 전원을 끌 더 나은 후보자를 탐색하였습니다.
- vSphere HA에 페일오버를 위한 추가 용량이 필요합니다.
- 로드가 낮아 호스트 전원 끄기를 트리거할 수 없습니다.
- DPM 프로젝트의 로드가 증가합니다.
- DPM은 호스트에 설정되지 않았습니다.
- DPM 임계값이 너무 높게 설정되었습니다.
- DPM이 호스트에 설정된 동안 적절한 전원 켜기 메커니즘이 호스트에 존재하는 것은 아닙니다.
- DRS가 호스트를 제거할 수 없습니다.
- DRS 마이그레이션 임계값이 가장 높은 설정에 있고 필수 이동만을 실행합니다.

해결책

DRS가 호스트 전원을 끄지 못하게 하는 문제를 해결합니다.

DRS가 유지 보수 모드나 대기 모드로 전환하도록 요청한 호스트를 제거하지 않음

DRS가 유지 보수 모드나 대기 모드로 전환하도록 요청한 호스트를 제거하지 않습니다.

문제

호스트를 유지 보수 모드나 대기 모드로 전환하려고 할 때 DRS가 호스트를 제거하지 않습니다.

원인

vSphere HA가 활성화되어 있는 경우 이 호스트를 제거하면 HA 페일오버 용량을 위반할 수 있습니다.

해결책

해결 방법이 없습니다. 해당하는 경우 호스트를 유지 보수 모드나 대기 모드로 전환하기 전에 vSphere HA를 비활성화합니다.

DRS가 가상 시스템을 호스트로 이동하지 않음

DRS가 가상 시스템을 호스트로 이동하지 않습니다.

문제

DRS에서는 가상 시스템을 DRS가 활성화된 클러스터에 추가된 호스트로 마이그레이션하는 것을 권장하지 않습니다.

원인

호스트가 DRS가 활성화된 클러스터에 추가된 후에는 호스트에 배포된 가상 시스템이 클러스터의 일부가 됩니다. DRS는 새로 클러스터에 추가된 호스트에 일부 가상 시스템을 마이그레이션하도록 하는 권장 사항을 제시할 수 있습니다. 이러한 권장 사항이 제시되지 않는 경우에는 vMotion, 호스트 호환성 또는 선호도 규칙에 문제가 있을 수 있습니다. 다음 사항은 예상 원인입니다.

- vMotion이 이 호스트에 구성되거나 활성화되지 않았습니다.
- 다른 호스트의 가상 시스템이 이 호스트와 호환되지 않습니다.
- 호스트가 가상 시스템에 충분한 리소스를 가지고 있지 않습니다.
- 이 호스트로 가상 시스템을 이동하면 VM/VM DRS 규칙 또는 VM/호스트 DRS 규칙을 위반하게 됩니다.
- 이 호스트는 HA 페일오버 용량으로 예약됩니다.
- 디바이스가 가상 시스템에 마운트되었습니다.
- vMotion 임계값이 너무 높습니다.
- 가상 시스템에 DRS가 비활성화되어 있으므로 가상 시스템을 대상 호스트로 이동할 수 없습니다.

해결책

DRS가 가상 시스템을 호스트로 이동하지 못하게 하는 문제를 해결합니다.

DRS가 가상 시스템을 호스트에서 이동하지 않음

DRS가 가상 시스템을 호스트에서 이동하지 않습니다.

문제

가상 시스템이 이 호스트에서 이동되지 않습니다.

원인

이는 vMotion, DRS 또는 호스트 호환성 관련 문제로 인한 것일 수 있습니다. 다음 사항은 예상 원인입니다.

- vMotion이 이 호스트에 구성되거나 활성화되지 않았습니다.
- DRS가 이 호스트의 가상 시스템에서 비활성화되어 있습니다.
- 이 호스트의 가상 시스템이 다른 호스트와 호환되지 않습니다.
- 이 호스트의 가상 시스템에 충분한 리소스를 가지고 있는 다른 호스트가 없습니다.
- 이 호스트에서 가상 시스템을 이동하면 VM/VM DRS 규칙 또는 VM/호스트 DRS 규칙을 위반하게 됩니다.
- DRS가 호스트에 있는 하나 이상의 가상 시스템에 대해 비활성화되어 있습니다.
- 디바이스가 가상 시스템에 마운트되었습니다.

해결책

DRS가 호스트에서 가상 시스템을 이동하지 못하게 하는 문제를 해결합니다.

VM과 해당 호스트 간의 비호환성으로 인해 DRS에서 시작된 VM 마이그레이션

DRS 활성화 클러스터에서는 클러스터의 일부 호스트만 대상으로 DRS에서 시작된 대량의 VM 마이그레이션을 관찰할 수 있습니다.

DRS는 로드 밸런싱 워크플로의 일부로 가능한 권장 사항을 찾기 위해 1분마다 클러스터의 호스트를 검색합니다. 결과는 클러스터의 호스트에 대해 전원이 켜진 VM의 호환성 검사 결과에 따라 달라집니다. VM의 호환되는 호스트 집합을 클러스터의 특정 호스트로만 제한하는 호환성 제약 조건이 있는 경우 DRS는 해당 VM을 호환되는 호스트 중 하나로 마이그레이션하여 이러한 제약 조건을 충족하려고 시도합니다.

제약 조건은 일반적으로 클러스터의 정적 사용자 구성 및 런타임 상태 변경이라는 두 가지 소스에서 발생합니다. 이 문서에서는 사용자에게 예상치 못한 런타임 상태 변경에 중점을 둡니다.

전원이 켜진 VM과 호스트 간의 호환성에 영향을 미칠 수 있는 런타임 상태 변경은 다음 영역 중 하나에 있을 수 있습니다.

vSphere High Availability

HA 활성화 클러스터에서는 모든 호스트가 정상 HA 상태여야 합니다. 호스트의 HA 상태가 어느 시점에 정상이 아닌 경우 VM 호환성 확인 중에 호환성 오류가 발생합니다. 이러한 상황의 몇 가지 예는 HA 에이전트에 연결할 수 없음, HA 에이전트가 격리됨, HA 에이전트가 분할됨입니다. 자세한 내용은 "[vSphere HA호스트 상태 문제 해결](#)"을 참조하십시오.

이러한 상태 변경은 일반적으로 vCenter Server에서 다음 이벤트를 수반합니다.

"호스트의 vSphere HA 에이전트에서 오류 발생"

스토리지 접근성

VM이 현재 호스트에서 구성 파일(VMX 파일), 가상 디스크(VMDK) 또는 스왑 파일에 액세스할 수 없는 경우 호환성 검사에 실패하여 현재 호스트가 호환되지 않습니다. 클러스터의 다른 호스트가 이러한 파일에 계속 액세스할 수 있는 경우 DRS VM을 해당 호스트로 마이그레이션하려고 시도할 수 있습니다. 이러한 마이그레이션의 결과는 현재 호스트에서 VM의 VMX 파일에 대한 액세스 가능성에 따라 달라집니다. VM이 VMX 파일을 제외하고 VMDK에 대한 액세스 권한만 상실한 경우 마이그레이션이 성공할 수 있습니다. VM이 VMX 파일에 액세스할 수 없게 되면 마이그레이션이 실패할 수 있습니다.

네트워크 접근성

NSX-T를 사용하는 환경의 경우 NSX 구성 요소 상태가 일부 호스트에서 작동 중지되거나 모든 호스트에서 서로 다른 시간에 작동 중지될 수 있습니다. 7.0u2 이전 버전의 vCenter Server에서는 이로 인해 VM 및 영향을 받는 호스트에 대한 호환성 검사가 실패할 수 있습니다.

해결책

- 1 사용자가 위에서 설명한 런타임 상태 변경으로 이어질 수 있는 작업을 수행하려는 경우 일시적으로 DRS를 수동 모드로 설정하면 원치 않는 마이그레이션을 방지할 수 있습니다. 또는 사용자가 DRS 고급 옵션인 VmsPerLBIteration을 일시적으로 0으로 설정할 수도 있습니다. 로드 밸런싱 워크플로 중에 VM을 검색하지 않도록 DRS에 요청하므로 마이그레이션 권장 사항이 없습니다.
- 2 vCenter 7.0 업데이트 1부터 DRS는 미리 정의된 기간인 CompatCheckTransientFailureTimeSeconds 동안 전원이 켜진 VM과 현재 호스트의 비호환성을 허용하는 고급 옵션을 도입했습니다. 사용자는 일시적인 비호환성으로 인한 원치 않는 마이그레이션을 방지하도록 이 옵션을 구성할 수 있습니다.

참고 이 옵션의 기본값은 600(10분)입니다. 이는 비호환성이 10분 미만 동안 지속되는 경우 DRS가 현재 호스트와의 비호환성 때문에 VM을 외부로 이동하지 않음을 의미합니다. 이 옵션의 최대값은 3600(60분)입니다.

vCenter 7.0 업데이트 3 및 8.0 업데이트 1부터 이 옵션을 -1로 설정할 수도 있습니다. 이렇게 하면 DRS가 현재 호스트와 호환되지 않기 때문에 VM을 다른 곳으로 이동할 수 없습니다.

vCenter 8.0 업데이트 3부터 이 옵션의 기본값은 -1로 설정됩니다.

DRS 고급 옵션을 설정하려면 vSphere Client에서 다음 단계를 수행합니다.

- 1 DRS 클러스터를 마우스 오른쪽 버튼으로 클릭하고 Settings > vSphere DRS > Edit > Advanced Options > Add를 클릭합니다.
- 2 [옵션] 열에 옵션 이름을 입력합니다.
- 3 [값] 열을 클릭하여 원하는 값을 입력한 다음 [확인]을 클릭하여 이 설정을 적용합니다.

가상 시스템 문제

가상 시스템 문제는 DRS를 예상한 대로 실행되지 않게 할 수 있습니다.

CPU 또는 메모리 리소스 부족

가상 시스템이 충분한 CPU 또는 메모리 리소스를 수신하지 않습니다.

문제

가상 시스템의 요구가 리소스 이용 자격보다 큰 경우일 수 있습니다. 이 경우 가상 시스템은 충분한 CPU 또는 메모리 리소스를 수신하지 않습니다.

원인

다음 섹션은 가상 시스템의 자격 부여에 영향을 미치는 요소를 설명합니다.

클러스터가 노란색 또는 빨간색으로 표시됨

클러스터가 노란색 또는 빨간색으로 표시되면 해당 용량은 클러스터의 리소스 풀과 가상 시스템에 구성된 리소스 예약을 충족하기에 충분하지 않습니다. 특정 가상 시스템은 해당 예약을 수신하지 않을 수도 있습니다. 클러스터(빨간색 또는 노란색)의 상태를 확인하고 이 문제를 해결합니다.

너무 엄격한 리소스 제한

가상 시스템, 상위 리소스 풀 또는 해당 리소스 풀의 상위 항목에 대해 리소스 제한이 너무 엄격하게 구성되었을 수 있습니다. 요구 수준이 구성된 제한 값과 같거나 이보다 큰지 확인하십시오.

클러스터 오버로드

가상 시스템이 실행되고 있는 클러스터의 리소스가 불충분할 수 있습니다. 또한 가상 시스템의 공유 값은 다른 가상 시스템이 비례적으로 더 많은 리소스를 보장한 값입니다. 요구 수준이 용량보다 큰지 확인하려면 클러스터 통계를 검토하십시오.

호스트 오버로드

호스트 리소스가 초과 가입되었는지를 확인하려면 호스트 통계를 검토하십시오. 초과 가입되었으면 DRS가 호스트에서 현재 실행되는 가상 시스템을 다른 호스트로 이동하지 않은 이유를 고려해 보십시오. 다음과 같은 원인이 있을 수 있습니다.

- VM/VM DRS 규칙과 VM/호스트 DRS 규칙이 현재의 가상 시스템과 호스트와의 매핑을 요구합니다. 그와 같은 규칙이 클러스터에 구성되어 있으면 그 중 하나 이상을 비활성화하는 것이 좋습니다. 그 후에 DRS를 실행하고 문제가 해결되었는지 확인하십시오.
- DRS가 용량을 확보하기 위해 이 가상 시스템이나 다른 가상 시스템을 다른 호스트로 이동할 수 없습니다. DRS는 다음과 같은 이유 때문에 가상 시스템을 이동하지 않습니다.
 - DRS 기능이 가상 시스템에서 비활성화되어 있습니다.
 - 호스트 디바이스가 가상 시스템에 마운트되었습니다.
 - 하나의 리소스 예약이 너무 커서 가상 시스템이 클러스터의 다른 호스트에서 실행될 수 없습니다.

- 가상 시스템은 클러스터의 다른 호스트와 호환되지 않습니다.

이 조건 중 어떤 것이 가상 시스템에 존재하는지 확인합니다. 해당되는 것이 없으면 클러스터의 다른 가상 시스템에 이 조건이 존재할 수도 있습니다. 이러한 경우에 DRS는 가상 시스템의 요구를 충족할 수 있는 수준으로 클러스터에 균형을 맞출 수 없습니다.

- DRS 마이그레이션 임계값 설정을 감소시키고 이 상황이 해결되었는지 여부를 확인합니다.
- 가상 시스템의 예약을 늘립니다.

해결책

가상 시스템이 충분한 CPU 또는 메모리 리소스를 수신하지 못하게 하는 문제를 해결합니다.

VM/VM DRS 규칙 또는 VM/호스트 DRS 규칙 위반

DRS 규칙은 가상 시스템이 있어야 하거나 있으면 안 되는 호스트 또는 동일한 호스트에 있어야 하거나 있으면 안 되는 가상 시스템을 지정합니다.

문제

VM/VM DRS 규칙 또는 VM/호스트 DRS 규칙을 위반했습니다.

원인

선택한 가상 시스템이 동일한 호스트(선호도)에서 배치되어야 하고 또는 다른 호스트(반선호도)에 이 가상 시스템이 배치되어야 함을 VM/VM DRS 규칙이 지정합니다. VM/호스트 DRS 규칙은 선택한 가상 시스템이 지정된 호스트(선호도)에 배치되어야 하고 또는 선택한 가상 시스템이 지정한 호스트(반선호도)에 배치되지 않아야 함을 지정합니다.

VM/VM DRS 규칙 또는 VM/호스트 DRS 규칙을 위반하는 경우 해당 규칙에서 DRS가 일부 가상 시스템 또는 모든 가상 시스템을 이동할 수 없음으로 인해 발생할 수 있습니다. 선호도 규칙에서 해당 가상 시스템이나 다른 가상 시스템의 예약 또는 상위 리소스 풀은 DRS가 동일한 호스트에 있는 모든 가상 시스템을 찾는 데 방해가 될 수 있습니다.

해결책

- 선호도 규칙과 연결된 [DRS 장애] 패널을 확인합니다.
- 선호도 규칙의 모든 가상 시스템 예약 합계를 계산합니다. 이 값이 호스트에서 사용할 수 있는 용량보다 큰 경우에는 이 규칙이 충족될 수 없습니다.
- 상위 리소스 풀의 예약 합계를 계산합니다. 이 값이 호스트의 사용 가능한 용량보다 큰 경우에는 리소스가 단일 호스트에서 얻어진다면 이 규칙이 충족될 수 없습니다.

가상 시스템의 전원 켜기 작업 실패

가상 시스템의 전원을 켜지 못했다는 오류 메시지가 나타납니다.

문제

가상 시스템의 전원을 켜지 못했습니다.

원인

리소스가 부족하거나 가상 시스템에 호환되는 호스트가 없기 때문에 가상 시스템의 전원을 켜지 못할 수 있습니다.

해결책

클러스터에 리소스가 부족하여 단일 가상 시스템 또는 그룹 전원 켜기 시도에 포함된 모든 가상 시스템의 전원을 켜지 못하는 경우 클러스터 또는 상위 리소스 풀에 사용 가능한 리소스를 기준으로 가상 시스템에 필요한 리소스를 확인합니다. 필요한 경우 전원을 켜 가상 시스템의 예약을 줄이거나, 형제 가상 시스템의 예약을 줄이거나, 클러스터 또는 상위 리소스 풀에 사용 가능한 리소스를 늘립니다.

DRS가 가상 시스템을 이동하지 않음

DRS가 초기에 호스트의 리소스가 부족할 때 전원이 켜진 경우 가상 시스템을 이동하지 않습니다.

문제

가상 시스템의 전원을 켜 때 가상 시스템이 등록된 호스트에 리소스가 부족하면 DRS가 해당 가상 시스템을 마이그레이션하지 않습니다.

원인

DRS가 가상 시스템을 이동하지 않는 원인은 다음과 같습니다.

- DRS가 가상 시스템에서 비활성화되어 있습니다.
- 가상 시스템에 디바이스가 마운트되었습니다.
- 가상 시스템이 다른 호스트와 호환되지 않습니다.
- 다른 호스트에 가상 시스템에 충분한 개수의 물리적 CPU가 없거나 각 CPU를 위한 용량이 충분하지 않습니다.
- 다른 호스트가 본 가상 시스템이 요청한 메모리와 예약을 충족할 수 있는 충분한 CPU나 메모리 리소스를 가지고 있지 않습니다.
- 가상 시스템을 이동하면 선호도 또는 반선호도 규칙을 위반하게 됩니다.
- 가상 시스템의 DRS 자동화 수준이 수동으로 지정되어 있으며 사용자가 마이그레이션 권장 사항을 승인하지 않았습니다.
- DRS는 Fault Tolerance가 활성화된 가상 시스템을 이동하지 않습니다.
- 자동 마이그레이션을 방해할 수 있는 VM 재정의 정의가 VM에 있는지 확인합니다.

해결책

DRS가 가상 시스템을 이동하지 못하게 하는 문제를 해결합니다.