**RAID는 왜? 사용되나**

서버의 사용자가 서버 운영에 있어 가장 당혹스러울 때는 하드디스크의 장애로 인한 DATA 손실일 것입니다. 하드디시크는 사실상 소모품으로 분류되며 I/O가 많은 서버에는 고장이 잦은 것이 당연하다고 볼 수 있습니다. 하지만 서버에 저장되는 데이터의 경우 손실 또는 유출 되었을 때 치명적인 것이 대부분일 것으로 대표적으로 은행과 같은 금융, 군사적 목적의 데이터가 있을 것입니다. 이로 인해 백업이 절대적으로 필요한 경우가 있고 또한, 여분의 디스크가 있어 용량을 증설하려고 할 때 데이터 손실 없이 증설이 필요한 경우가 있습니다. 그래서 많은 서버 관리자는 RAID 구성을 통해 하드디스크의 가용성을 높이거나 서버 데이터의 안정성을 확보해야 합니다.

**RAID란?**

RAID는 Redundant Array of Inexpensive Disks의 약자입니다. 여러 개의 디스크를 배열하여 속도의 증대, 안정성의 증대, 효율성, 가용성의 증대를 하는데 쓰이는 기술입니다.

**RAID의 장점**

1) 운용 가용성, 데이터 안정성 증대

2) 디스크 용량 증설의 용이성

3) 디스크 I/O 성능 향상

**그 전에 패리티(Parity)란 무엇이란?**

1) 정수의 홀수 또는 짝수 특성

2) 정보 블록과 연결된 중복 정보, 패리티는 오류 후 데이터를 재구축하는데 사용되는 계산된 값입니다.

**RAID의 종류와 구성방식**

**RAID 0**: RAID 0에는 Concatenate 방식과 Stripe 방식 두 가지 방식이 있습니다.

**1) Concatenate 방식 (두개 이상의 디스크에 데이터를 순차적으로 쓰는 방법)**

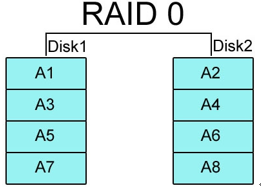


장점: 디스크 기본 공간이 부족할 때 데이터는 보존하며 여분의 디스크를 볼륨에 포함하여 용량 증설이 가능하게 됩니다.

단점: RAID 0의 특성상 디스크 중 하나의 디스크라도 장애가 발생하면 복구가 어렵고, 패리티(오류검출기능)를 지원하지 않습니다.

용량: 모든 디스크의 용량을 합친 용량 (300GB disk \* 2ea = 600GB)

**2) Stripe 방식 (흔히 RAID 0라고 하면 Stripe 방식을 말함)**



두개 이상의 디스크에 데이터를 랜덤하게 쓰는 방법입니다.

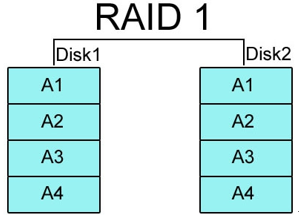
장점: 데이터를 사용할 때 I/O를 디스크 수만큼 분할하여 쓰기 때문에 I/O 속도가 향상되고 I/O Controller나 I/O board 등 I/O를 담당하는 장치가 별도로 장착된 경우 더 큰 I/O 속도 향상 효과를 볼 수 있씁니다.

단점: Stripe를 구성할 시 기존 데이터는 모두 삭제 되어야 합니다. 그외의 단점은 위의 Concat 방식과 같습니다.

용량: 위의 Concat 방식과 같습니다.

**RAID 1(Mirroring)**

: Mirror 볼륨 내의 패리티를 사용하지 않고 디스크에 같은 데이터를 중복 기록하여 데이터를 보존하게 되며, 적어도 동일한 용량의 디스크 두 개가 필요합니다.



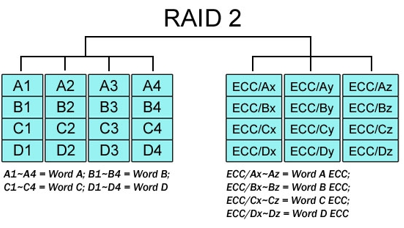
장점: 볼륨 내 디스크 중 하나의 디스크만 정상이어도 데이터는 보존되어 운영이 가능하기 때문에 가용성이 높고, 복원이 비교적 매우 간단합니다.

단점: 용량이 절반으로 줄고, 쓰기 속도가 조금 느려집니다.

용량: 모든 디스크의 절반의 용량 (300GB \*2ea = 300GB)

**RAID 2**

: RAID 2는 RAID 0처럼 striping 방식이지만 에러 체크와 수정을 할 수 있도록 Hamming code를 사용하고 있는 것이 특징입니다. 하드 디스크에서 ECC(Error Correction Code)를 지원하지 않기 때문에 ECC를 별도의 드라이브에 저장하는 방식으로 처리됩니다.



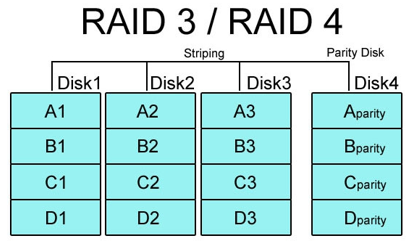
하지만 ECC를 위한 드라이브가 손상될 경우는 문제가 발생할 수 있으며 패리티 정보를 하나의 하드 드라이브에 저장하는 RAID 4가 나오면서 거의 사용되지 않는 방식 입니다.

**RAID 3, RAID 4**

: RAID 3, RAID 4는 RAID 0, RAID 1의 문제점을 보완하기 위한 방식으로 3, 4로 나뉘긴 하지만 RAID 구성 방식은 거의 같습니다. RAID 3, 4는 기본적으로 RAID 0과 같은 striping 구성을 하고 있어 성능을 보완하고 디스크 용량을 온전히 사용할 수 있게 해주는데 여기에 추가로 에러 체크 및 수정을 위해서 패리티 정보를 별도의 디스크에 따로 저장하게 됩니다.

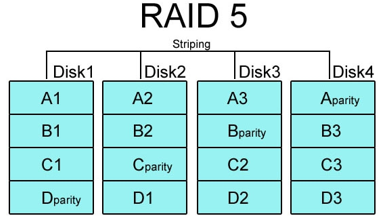
RAID 3은 데이터를 바이트 단위로 나누어 디스크에 동등하게 분산 기록하며 RAID 4는 데이터를 블록 단위로 나눠 기록하므로 완벽하게 동일하진 않다는 차이가 있습니다.

RAID 3은 드라이브 동기화가 필수적이라 많이 사용되지 않고 RAID 4를 더 많이 쓴다고 보시면 됩니다.



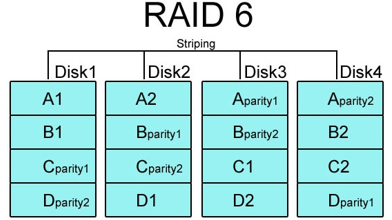
**RAID 5**

: RAID 5는 RAID 3,4에서 별도의 패리티 정보 디스크를 사용함으로써 발생하는 문제점을 보완하는 방식으로 패리티 정보를 stripe로 구성된 디스크 내에서 처리하게 만들었습니다. 만약 1개의 하드가 고장나더라도 남은 하드들을 통해 데이터를 복구할 수 있다는 장점이 있습니다.

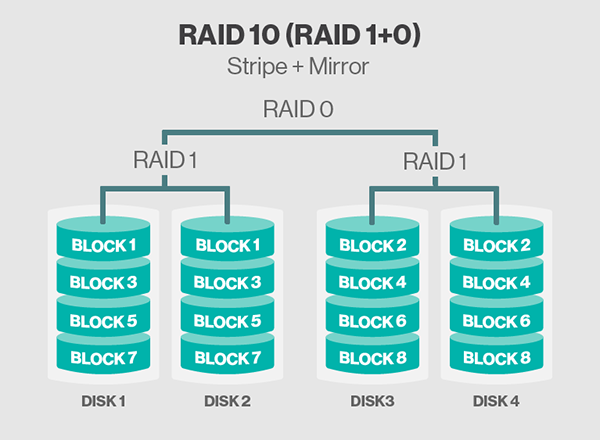


**RAID 6**

: RAID 6은 RAID 5와 같은 개념이지만 다른 드라이브들 간에 분포되어 있는 2차 패리티 정보를 넣어 2개의 하드에 문제가 생겨도 복구할 수 있게 설계되었으므로 RAID 5보다 더욱 데이터의 안전성을 고려하는 시스템에서 사용됩니다.



RAID 10 (RAID 1+0)



RAID 1과 RAID 0을 결합한 이 구성은 RAID 1보다 높은 성능을 제공하지만 추가 비용이 든다. RAID 1+0에서는 데이터가 미러링되고 미러가 스트라이핑 되기 때문이다.

최소 디스크 개수 : 4

이 외에도

RAID 01 (RAID 0 + 1)

RAID 03 (RAID 0 + 3), RAID 53 (RAID 5 + 3)

RAID 50 (RAID 5 + 0)

RAID 7 (비표준이며 RAID 3, 4를 기반으로 하지만 캐싱 추가)

등이 있는데 이는 다 기존의 RAID 구성을 혼합하면 다 만들어낼 수 있다.

정리하면 다음과 같다.

장점 : 성능, 탄력성 및 비용은 RAID의 주요 장점이라고 볼 수 있다. 여러 개의 하드 드라이브를 함께 설치하면 RAID를 사용하여 단일 하드 드라이브의 작업을 향상시킬 수 있으며 구성 방법에 따라 충돌 후 컴퓨터 속도와 안정성이 향상 될 수 있기 때문이다.

단점 : 중첩 된 RAID 레벨은 많은 수의 디스크가 필요하기 때문에 기존의 RAID 레벨보다 구현 비용이 비싸다. 대다수의 디스크가 중복성을 위해 사용되기 때문에 중첩 된 RAID의 경우 스토리지 비용도 높습니다. 하지만 중첩 된 RAID는 표준 RAID 레벨과 관련된 일부 안정성 문제를 극복하는 데 도움이되므로 비용에도 불구하고 널리 보급되어 사용되었다.

출처: <https://12bme.tistory.com/286>