

Computer Hardware & Industry Perspective

C.H.I.P 2주차 선각 Report
크래프톤 정글 8-10 백지원

Computational Storage란 무엇인가?

1. 서론

현대 사회의 디지털화로 인해 생성·저장되는 데이터의 양은 기하급수적으로 증가하고 있다. 이에 따라 기존 컴퓨팅 아키텍처는 데이터를 저장소에서 중앙처리장치(CPU)로 옮겨 처리하는 과정에서 심각한 병목 현상에 직면하고 있다. 특히 지연 시간, 대역폭 한계, 전력 소비 등은 성능 저하의 핵심 원인으로 지적된다. 이러한 한계를 해결하기 위한 새로운 접근법으로 Computational Storage 기술이 부상하고 있다.

2. 본론

2.1 개념 및 정의

Computational Storage는 데이터를 저장하는 장치 내부 또는 그 인접 영역에서 연산을 수행하는 기술을 말한다. 즉, 데이터를 굳이 CPU나 메인 메모리로 옮기지 않고, 스토리지 장치 자체에서 처리함으로써 데이터 이동을 최소화하고 전체 시스템 효율을 극대화하는 것이 핵심이다. SNIA는 이를 "호스트의 연산 부담을 줄이기 위해 스토리지와 결합된 연산 기능(CSF)을 제공하는 아키텍처"로 정의한다.

개발자 관점에서는 연산 로직이 CPU 중심에서 스토리지 중심으로 옮겨가는 구조적 변화를 의미하며, 이는 애플리케이션 성능 향상과 아키텍처 설계의 혁신으로 이어질 수 있다.

2.2 동작 방식과 구성 요소

이 기술은 고속 저장장치(예: SSD, NVMe), 내장형 컴퓨팅 장치(CPU, FPGA, ASIC 등), 그리고 이들 간의 인터페이스로 구성된다. 호스트는 스토리지에 연산을 요청하고, 스토리지 내부 장치가 직접 데이터를 읽어 연산을 수행한 뒤, 결과만 전달한다. 이 구조는 데이터 이동 최소화과 병렬 처리 최적화라는 두 가지 이점을 동시에 제공한다.

2.3 장점과 활용 분야

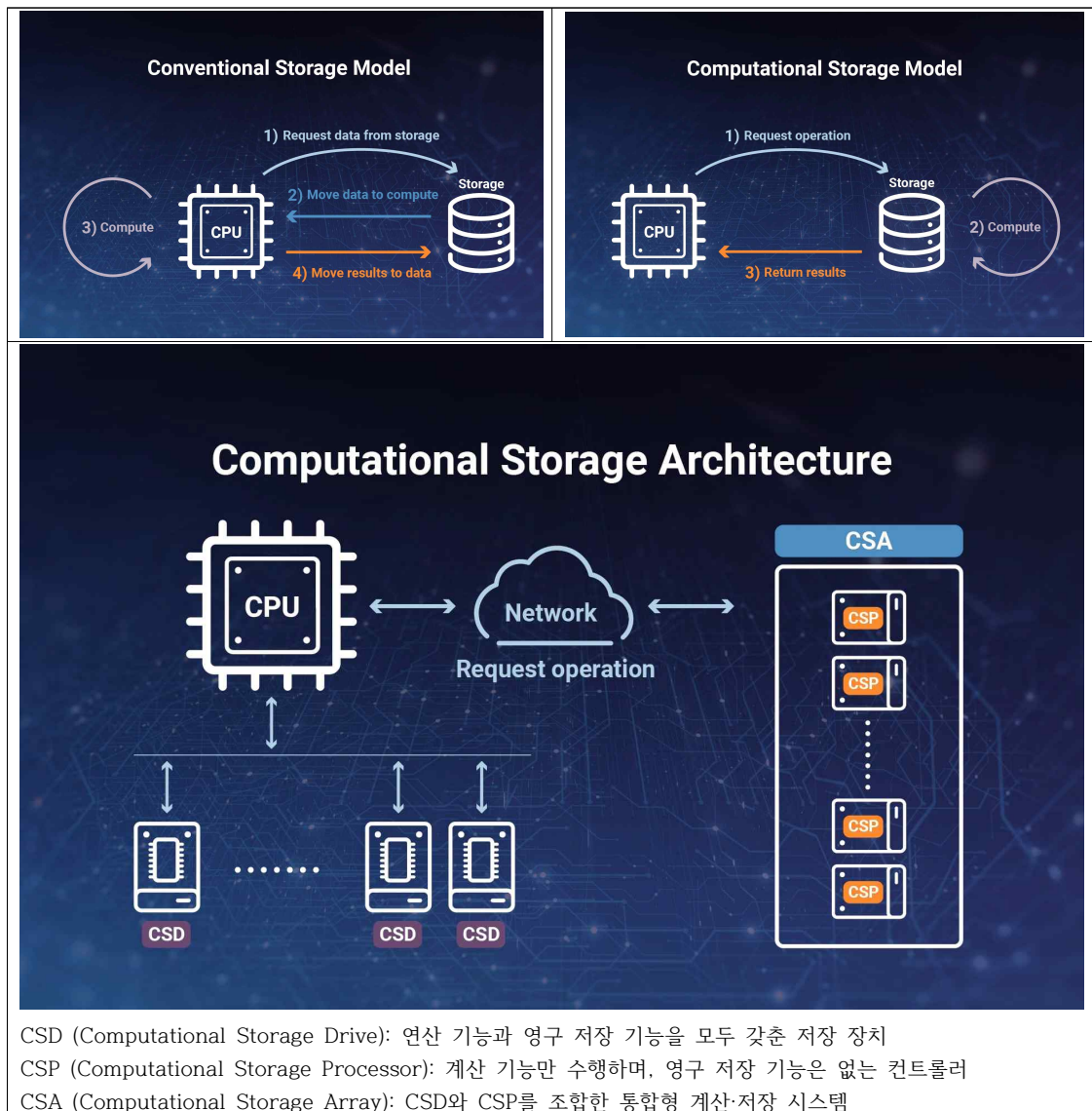
Computational Storage는 다음과 같은 효과를 기대할 수 있다.

- 성능 향상: 데이터 이동을 줄여 연산 지연 최소화, 스토리지 병렬 처리를 통한 속도 개선
- 대역폭 절감: CPU와 스토리지 간 불필요한 데이터 전송을 줄여 전체 시스템 대역폭 절약
- 전력 효율 증대: 데이터 이동에 필요한 에너지를 절감 -> 시스템 전체의 전력 사용량 줄임
- CPU 부하 감소: 일부 연산을 오프로드함으로써 CPU는 더 중요한 작업에 집중 가능

이러한 장점은 다음과 같은 분야에서 특히 주목받고 있다.

- 데이터베이스: 조건 필터링, 집계 등을 스토리지에서 처리하여 쿼리 응답 속도를 향상시킴
- 빅데이터 분석: 저장소에서의 병렬 연산을 통해 대규모 분석 시간 단축
- AI/머신러닝: 데이터 전처리 작업을 스토리지에서 수행하여 모델 학습 속도 개선
- 네트워크 보안: 패킷 필터링, 암호화 등의 작업을 분산 처리하여 네트워크 성능 향상

이 외에도 스토리지 내부에서의 데이터 처리 특성상, 보안성 확보와 표준화된 인터페이스 제공이 중요한 이슈로 부상하고 있으며, 현재 업계에서는 관련 기술 규격 정립도 활발히 논의되고 있다.



3. 결론

Computational Storage는 기존의 중앙 집중식 컴퓨팅 구조가 가진 한계를 극복할 수 있는 실질적인 대안으로 주목받고 있다. 데이터 이동을 최소화하고 연산을 분산시킴으로써 시스템 성능, 효율성, 에너지 절감 등 여러 측면에서 긍정적인 효과를 가져올 수 있다.

이러한 변화는 개발자에게도 새로운 시사점을 던진다. 연산 로직을 스토리지로 분산하는 설계 방식은 기존 CPU 중심의 프로그래밍 모델과는 전혀 다른 접근을 요구하기 때문이다. 앞으로는 데이터가 '어디에 저장되었는가'뿐 아니라 '어디서 처리되는가'를 고려한 시스템 아키텍처 설계 역량이 더욱 중요해질 것이다.