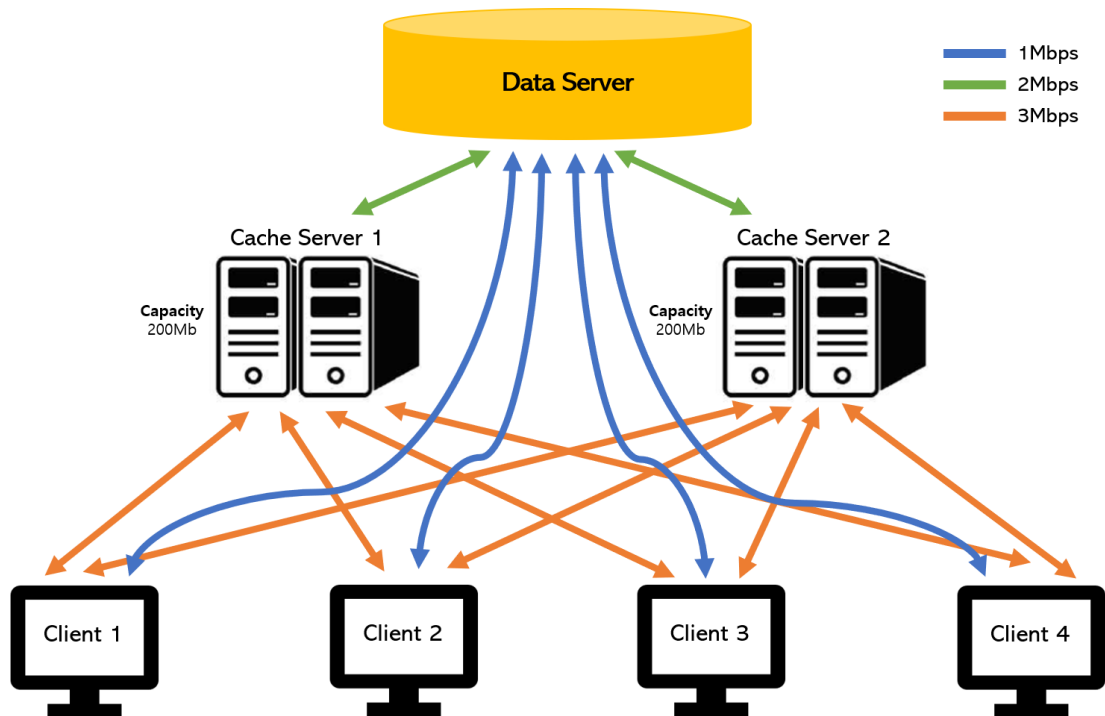


HW#2 데이터통신 프로그래밍 과제

Cache Server 및 Multi-Thread를 이용한 파일 다운로드 가상 환경 구현

- DUE: 2024.10.14(월) 23:59 (Assigned: 2024.09.30. 16:30)
- How: 1개화일 제출 – G조이름HW2.zip (조별로 한명만 제출하면 됨)
 - ◆ 모든 소스화일 및 결과화일 전체를 압축한 G조이름HW2.zip으로 제출 ex) G1HW2.zip

1. (시뮬레이션 환경 및 구성요소) 다음과 같은 실험환경을 구성한다. 1개의 Data Server, 2개의 Cache Server, 4개의 Client는 thread간 Socket 통신으로 동작하며 연결 구조의 제약은 없다. 각각의 객체는 다음의 기능을 수행한다.



- ① Data Server: 전역변수 혹은 임의의 객체인 System Clock을 유지, 관리, 업데이트 하고 프로그램이 수행되면 System Clock(msec 단위로 기록)을 수행시킨다. Data Server 에서는 1~10000 까지의 가상의 파일(객체 또는 변수로 구현)을 생성한다. 이때 각 파일은 1~10000kb라고 가정(예시: 1번파일-1kb, 130번파일-130kb, n번파일-nkb)하며 해당 파일들을 Cache Server와 Client의 요청시 전송하도록 한다.
- ② Cache Server: Cache Server는 Client의 데이터 요청 시 Data Server 보다 빠른 응답이 가

능하다. 환경에서 Cache Server는 2개 존재하며, 각 Cache Server에서는 저장 가능한 파일의 개수의 제한은 없지만 전체 파일의 용량은 200Mb까지 저장 가능하다. 따라서 모든 파일의 용량 합이 200Mb를 초과하면 안되며, 각 파일을 분할해서 저장하는 방법 또한 불가능하다. 만약 Client에서 요청한 파일을 Cache Server에서 보유하고 있지 않다면 Cache Server는 2Mbps의 속도로 Data Server에서 파일을 얻어 전달 해 주거나, Client에게 직접 Data Server로 요청하라는 메시지를 전송한다. 방식에 대한 제약은 없으며 구현한 규칙 및 알고리즘은 명시하여야 한다.

- ③ Client: 각각의 Client는 시작이후 1~10000번 파일 중 무작위로 선택된 1000개의 파일을 전송 받아야 하며, 전송 받을 파일의 순서는 변경이 불가능 하다. 파일을 요청 시 Cache Server와 Data Server 중 하나를 선택하여 진행되며, 하나의 파일을 분할해서 받는 것은 불가능하지만 같은서버에서 다른 파일을 동시에 전송 받는 것은 가능하다. 어떤 서버에서 파일을 전송 받을지 선택하는 방식의 제약은 없으며 구현한 규칙 및 알고리즘을 명시하여야 한다. 따라서 "최적의 방식을 구현하여 최대한 단축된 시간내에 모든 파일을 전송 받을수 있도록" 하여야 한다.

- ④ 전송 속도: 모든 네트워크 속도는 양방향 동일하며, 각 구성 요소 별 전송 속도는 다음과 같다

1) Data Server <--> Client: 1Mbps,

2) Data Server <--> Cache Server: 2Mbps

3) Cache Server <--> Client: 3Mbps,

모든 구성 요소의 네트워크 연결은 독립적으로 구성되어 있다고 가정하며, 예를 들어 하나의 Cache Server에 대해 4개의 Client가 연결될 경우 각각 3Mbps의 전송 속도를 보장한다. 또한 Client가 30kb 크기인 30번 파일을 Cache Server에서 전송 받는다고 하면 10msec 소요되며, Data Server에서 전송 받을 시 30msec가 소요된다. 또한 Data Server와 Cache Server는 하나의 네트워크 연결에서 두개 이상의 파일을 Client로부터 전송이 가능하며, 모든 파일은 동일한 비율로 전송이 된다. Data Server로부터 3개의 파일을 Client로 전송 시 각 파일은 0.33Mbps의 속도로 동시에 전송받게 된다. 또한 파일 요청을 위한 Client와 Cache Server의 메시지와 같은 파일 전송이외의 데이터 처리 및 추가 통신에 소요되는 시간은 없으며, 명시한 전송 속도 이외의 delay는 없다고 가정한다. 각 구성요소의 모든 통신 및 파일 송수신에 관한 event는 System Clock과 함께 Log에 기록한다.

※ 모든 Server는 반드시 AWS나 구글클라우드 등 Physically 외부서버에 구현 되어야함!!

※ 모든 Server와 Client의 통신은 반드시 Socket으로 구현되어야 함!!

※ 모든 Server와 Client는 Thread를 통해 구현되어야 함!!

(해당 세가지 조건은 **과제수행의 전제조건**으로 만족되지 않으면 과제수행 인정불가)

2. (시뮬레이션 시나리오) 다음과 같은 실험 시나리오를 수행하여 각각의 구성요소 별 Log를 기록한다. 즉, Data Server와 2개의 Cache Server, 4개의 Client는 각각 모든 이벤트와 파일 송수신 정보를 Data.txt, Cache1.txt, Cache2.txt, Client1.txt, Client2.txt, Client3.txt, Client4.txt에 기록한다.

- ① 시나리오: 프로그램 시작 후, 모든 구성 요소 간 통신 환경을 초기화 후 Data Server는 가상의 system clock을 생성합니다. System clock은 Data Server에서 관리하며, 실제 시간이 아닌 가상의 파일 및 데이터 전송 시 소요되는 시간이다. 그 다음 system clock 0sec에서 시작하며 모든 Client는 무작위로 선택된 파일 1000개를 순차적으로 Server에 요청하게 되며, 동일한 파일을 분할하여 전송받는 것은 불가능 하다. 다만 여러개의 파일을 각각 하나 이상의 Server에서 전송 받을 수는 있지만, 모든 구성요소간 네트워크 속도에 따라 종속되며, 예를 들어 하나의 Data Server에서 Client로 n개의 파일을 전송 받는다면 각 파일은 $(1/n)$ Mbps의 속도로 동시에 전송받는다. 이때 동일한 Client가 다른 Cache Server에서 또 다른 m개의 파일을 동시에 전송받는다면 $(3/m)$ Mbps로 전송받는다. 즉 다른 네트워크 연결일 경우 각 네트워크의 속도에 간섭받지 않는다. Client가 어떤 Server에게 요청을 할지, Cache 서버에 요청하는 파일이 존재하지 않는다면 어떻게 관리할지에 대한 방식의 제약은 없으며 모든 규칙 및 알고리즘은 명시하여야 한다. 1-4에서 명시한 네트워크 속도에 따른 전송 작업의 소요 시간은 작업이 끝난 후 Data Server에 존재하는 System Clock에 업데이트 하여야 하며, 이때의 "네트워크 비용은 존재하지 않는다고 가정"한다. 모든 Client에서 각각 1000개의 파일을 모두 전송 받았다면, 모든 구성요소는 각각 연결을 종료한다(모든 구성요소는 Gracefully Termination 되어야 함). 그리고, 파일 전송 이외 메시지 전달은 전송 비용이 없다고 가정한다.(단, Cache Server의 파일교체의 전송시간은 고려한다.) 프로그램 수행동안 구성요소들은 모든 event(파일 요청 메시지, 파일 송수신 내역, 각 구성요소의 전체 수행시간)와 평균 파일 전송 속도를 system clock과 함께 log에 기록한다. 또한 Cache Server는 Cache hit(Client 요청한 파일이 Cache Server에 있을 경우)및 Cache miss(Client 요청한 파일이 Cache Server에 없을 경우)의 횟수, Client는 각 파일 수신에 소요되는 평균 시간을 각 system clock과 함께 log에 기록한다. log의 마

지막에는 Data Server는 전송한 전체 파일 크기에 대한 전송 속도, Cache Server는 송신과 수신에서 각각 발생한 전체 파일 크기에 대한 전송 속도, Client는 전송 받은 전체 파일 크기에 대한 전송 속도를 기록한다.

3. (과제제출) 다음 화일을 조별로 기한안에 제출한다.

➤ G조이름HW2.zip (ex. G1HW2.zip)

- ◆ 모든 소스 화일들
- ◆ AllDefinedLogs.txt
 - 조별로 프로그램에서 정의한 모든 Log 메시지 명세 및 설명
- ◆ 서버 및 클라이언트별 출력된 모든 Log 화일들
- ◆ download.txt
 - G조이름HW2.mp4 (ex. G1HW2.mp4) : 5분이내의 설명 동영상을 제작하여, 해당동영상을 다운로드 할 수 있는 link를 포함하는 화일
 - 반드시 동영상이 다운되는 링크를 삽입할 것
 - 포함된 링크로 동영상 다운이 안되거나, 공유권한이 없거나, 다운 후 동영상 실행이 안되거나 하는 등의 에러는 프로그램 수행이나 포함내용의 미포함처럼 모두 해당조의 과실로 감점대상임
- ◆ Readme.txt
 - 조 이름, 모든 조원 학번&이름
 - 학생 별 역할 명시
 - 프로그램 구성요소 설명
 - 소스코드 컴파일방법 명시
 - 프로그램 실행환경 및 실행방법 설명
 - 규칙 및 알고리즘 설명
 - Error or Additional Message Handling에 대한 사항 설명
 - Additional Comments: 추가로 과제제출관련 언급할 내용 작성