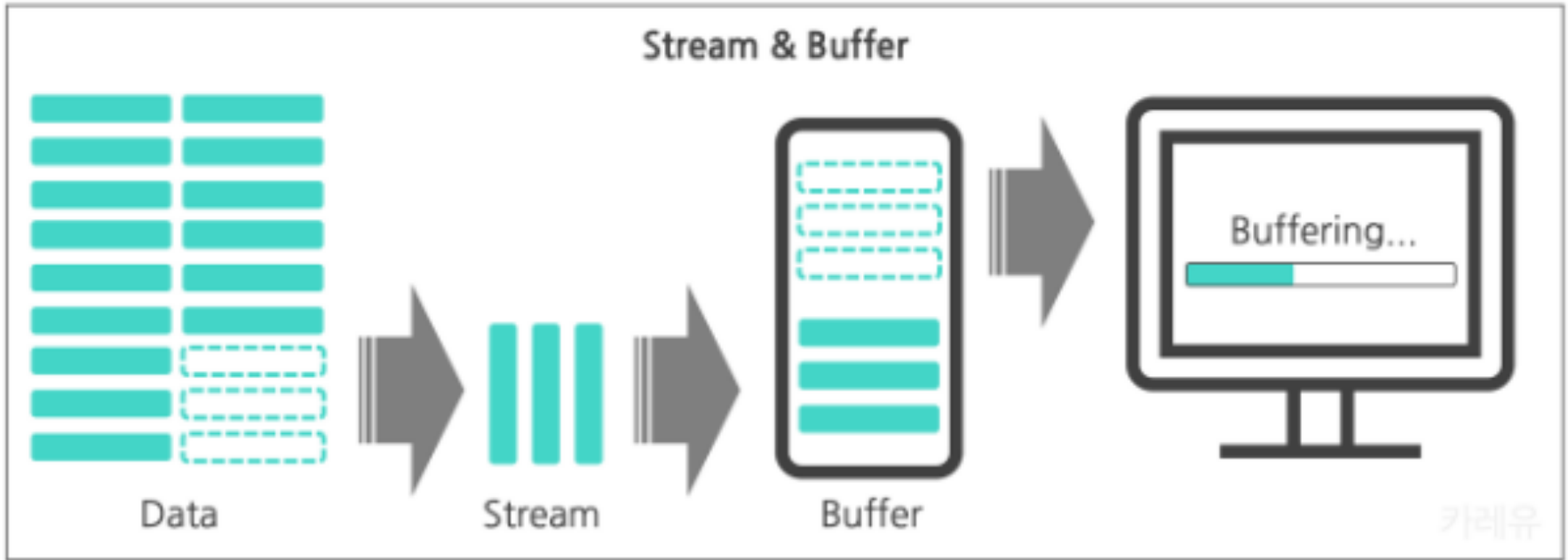


스트림(Stream)이란?

"스트림이란 시간의 흐름에 따라 발생하는 연속적인 데이터의 흐름이다."



스트리밍의 원리 (출처: 직접 그림)

이게 무슨 말인지 이해하기 위해
먼저 스트림과 파일의 기본적인 차이부터 잡고 넘어가자.
(이해하기 쉽게 단순화해서!)

파일은 mp3 처럼 4분 30초짜리 음원 데이터 전체가 기록되어 있다.
따라서 고정되어 있는 크기의 데이터를 처음부터 끝까지 읽어서 처리할 수 있다.

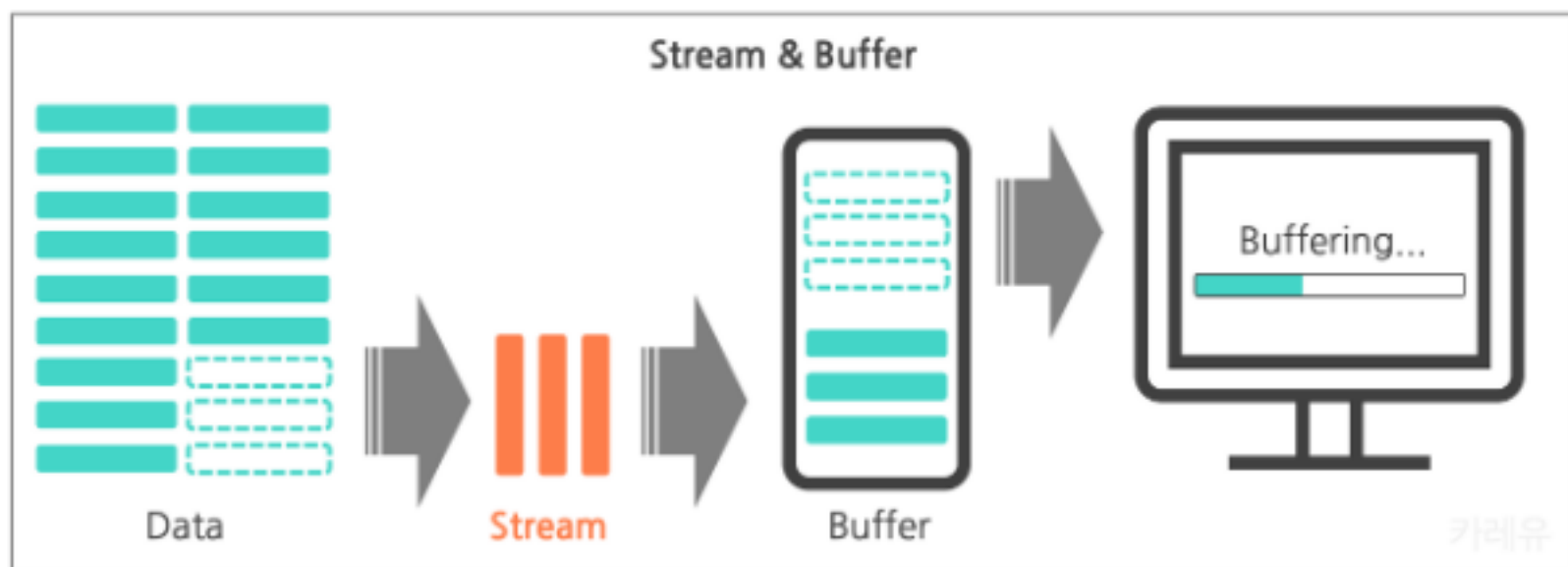
하지만 스트림은 파일과는 다르다.
스트림이란 계속해서 흘러 들어오는 데이터를 의미한다.

즉, 마이크에 대고 하는 이야기나, 비디오 카메라로 찍는 동영상 처럼
입력장치를 통해 계속해서 들어오는 데이터의 흐름을 의미한다.

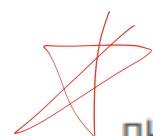
따라서 실제 데이터가 입력되기 전에는 어떤 데이터가 들어올지 알 수 없으며,
실시간으로 들어오는 데이터를 그때 그때 읽고 처리해야 한다.

혹은 유튜브 처럼 대용량의 데이터를 서버를 통해 받아서 재생하는 동영상의 경우,
데이터를 모두 다운로드 받아서 재생하려면 긴 딜레이가 발생하게 된다.

이를 막기 위해서는
먼저 들어오는 데이터부터 조금씩 메모리의 임시공간에 저장하여 재생시켜 주어야 한다.



Stream (출처: 직접 그림)



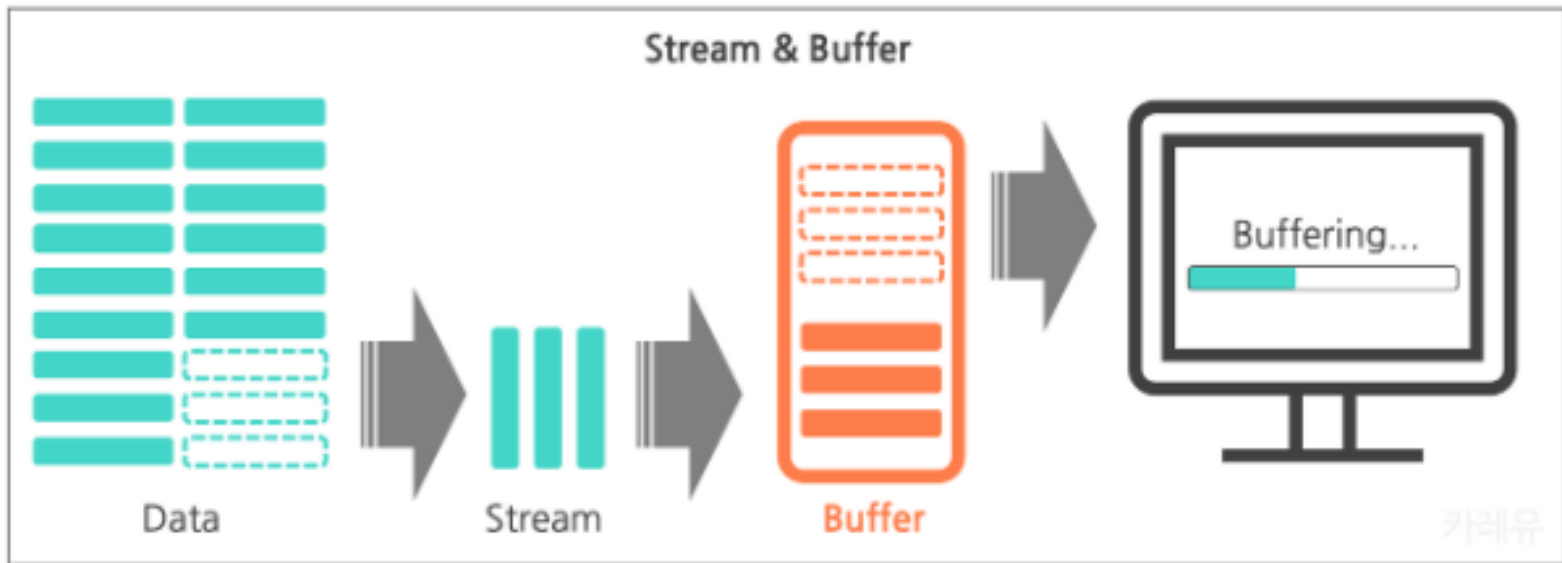
구체적으로 프로그래밍 세계에서의 "**스트림(stream)**"은
마이크와 같은 입력장치나 서버로 부터 들어오는 일련의 데이터 흐름을 읽어 들이면서,
스피커와 같은 출력장치나 동영상플레이어로 데이터의 흐름을 내보내는 역할을 하는 "객체"라고 볼 수 있다.

스트림이 있기 전까지는 데이터 전체가 다운로드 되어야만 처리가 가능했다.
스트림이 가능해지면서 동영상 스트리밍, 오디오 스트리밍 등의 서비스가 가능해진 것이다.

그런데 컴퓨터는 계속해서 흘러 들어오는 데이터들을 정확히 어떤 방식으로 처리할까?
 여기서 바로 버퍼(Buffer)가 출현한다.

버퍼(Buffer)란?

버퍼는 일반적으로 스트림 데이터를 조금씩 읽고, 저장하고, 처리하고, 비우기를 반복하는 메모리 공간이다.
이런 행위를 버퍼링이라고 하며, 임시로 데이터를 저장하는 메모리 공간 혹은 데이터 자체를 버퍼라고 한다.
(일반적으로 RAM메모리의 일부공간이 버퍼를 위해 사용된다.)



Buffer (출처: 직접 그림)

즉, Buffer란 특정 용도로 활용하기 위해 임시로 저장하는 Binary 형태의 데이터 저장 객체이다.

정리하면,

1. **스트림**은 버퍼를 이용해 흘러 들어오는 데이터를 처리(입력&출력)하는 객체이고,
2. **버퍼**는 스트림을 위해 데이터를 chunk단위로 쪼개어 임시로 저장해 두는 객체이다.

개인적으로는 스트림은 물레방아가 돌아가는 원리와 비슷하다고 생각한다.

물(입력데이터)이 계속해서 통(버퍼)에 담기다가,

특정 수위가 되면 통이 얹어지면서 물(출력데이터)이 떨어져 물레방아를 돌리고,
떨어진 물(출력)이 물레방아를 돌리고 있는 동안, 새로운 물(입력)이 통(버퍼)에 또 채워지길 반복하는 것이다.