

# 총기 연사음 개선 R&D

## 목차

- R&D 배경
- 이슈 및 개선 방향
  - 이슈 및 원인
  - 개선 방향
- 총기연사음 플레이백 시스템
  - 기존 플레이백 시스템
    - Polyphonic 방식
    - Granular 방식
  - 개선 플레이백 시스템
    - Hybrid (Polyphonic + Granular) 방식
    - Granular (Wwise Sync) 방식
  - 추가 아이디어
    - Thread 분리 방식
    - 재생지연 + 카운트다운 방식
- Demo 프로젝트
- 결론
- 문의

## R&D 배경

FPS 게임에서 사실적인 총기음을 표현하기 위해서는 총기 사운드를 직접 녹음하고 편집/가공한 후 게임에 적용하는 과정이 필요합니다. 총기 단발(Semi-Auto)방식의 경우에는 발사를 시도할 때마다 오직 1발의 총알만 발사하기 때문에 사운드 적용이나 구현이 매우 쉽고 간단합니다. 하지만 연사모드(Full-Auto)에서는 Game Thread의 가변 Frame rate의 영향으로 발사 간격이 일정하지 않다는 문제가 존재할 수 있습니다. 특히 발사 간격이 일정하지 않은 상황에서 연사 사운드를 자연스럽게 표현하기 위해서는 사운드 플레이백 시스템 측면의 세심한 고려가 필요합니다. LLL 총기연사음 개선 R&D는 현재 LLL 총기연사음 플레이백 시스템을 분석하고, 개선점을 파악하여 더 사실적인 총기연사음을 표현하기 위해 진행되었습니다.

## 이슈 및 개선 방향

### 이슈 및 원인

LLL 프로젝트 개발 과정에서도 구현 방식에 따라 아래와 같은 이슈가 있었습니다. ([상세 테스트 로그](#))

#### (A) Polyphonic 재생 방식 사용 시

- 현상: 불규칙한 발사음 트리거 간격으로 인한 사운드 품질 저하
  - 총기 연사 시 발사음이 고르게 들리지 않음
  - [Gun\\_재생비교\\_Polyphonic\\_vs\\_Granular.mp4](#)
- 원인
  - 엔리얼 Frame rate에 따라 발사음 트리거 간격이 달라져서 사운드 재생 간격도 균일하지 않음

#### (B) Granular 재생 방식 사용 시

- 현상: Fire 개수와 발사음 재생 개수 불일치
  - e.g. 게임에서는 1발 발사했는데 발사음은 2번 재생, 2발 발사했는데 발사음은 3번 재생, ...
  - [Gun\\_재생비교\\_발사음개수불일치.mp4](#)
- 원인
  - 엔리얼에서 사용하는 clock과 Wwise에서 사용하는 clock이 일치하지 않음
    - 엔리얼에서 생각하는 1초와 Wwise에서 생각하는 1초가 실제로는 서로 다른 길이
    - 테스트 결과 Wwise 시간이 실제 시간에 가까운 것으로 보임 ([\[Log\] 테스트 로그](#))
  - 엔리얼 frame rate에 따라 발사음 트리거 간격이 균일하지 않아 사운드와 싱크가 어긋나게 됨
  - 엔리얼-Wwise 통신에 0.01~0.04초 가량이 소요되므로 고RPM 총기에서 싱크 차이 발생







## 개선 방향

2.1에 따라, 아래 두 가지 조건을 만족하는 시스템을 제안하고자 R&D를 진행했습니다.

#### (A) 균일한 발사음 트리거 간격 보장

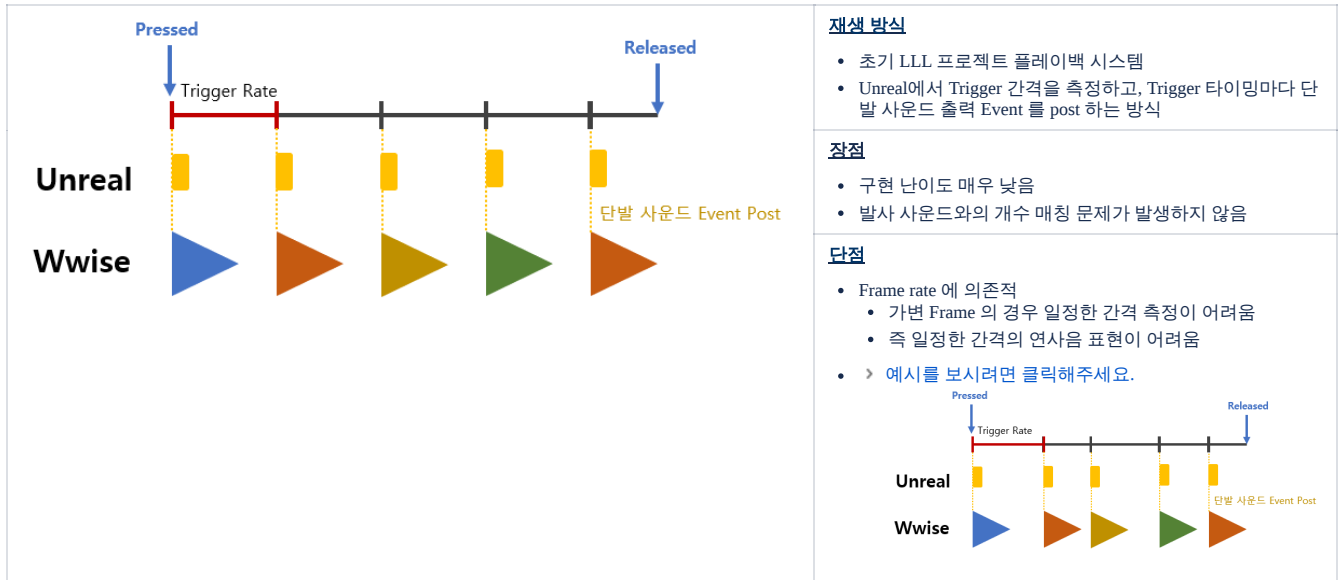
#### (B) 발사음 개수 일치 보장

## 총기연사음 플레이백 시스템

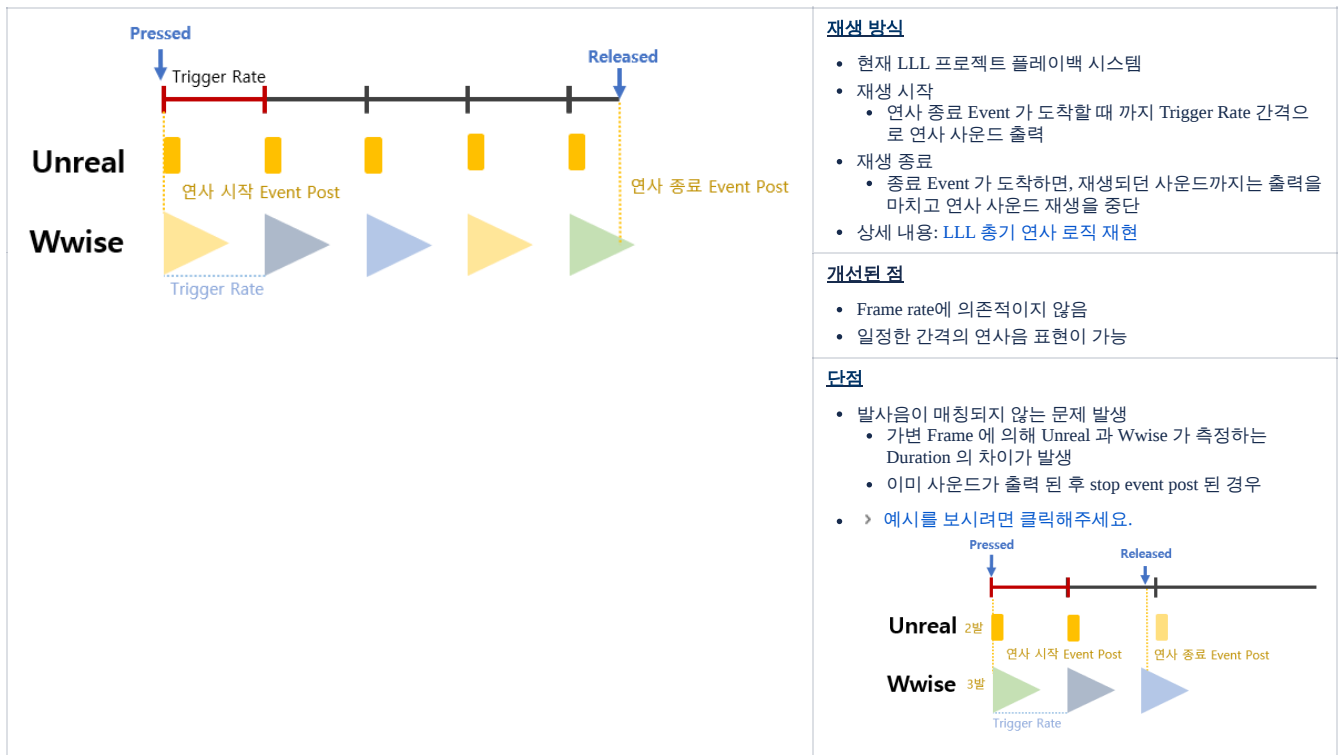
Projectile	Single fire sound source	Rapid fire sound source	Event Post	Delegate	Fire Event Action
					

## 기존 플레이백 시스템

### Polyphonic 방식

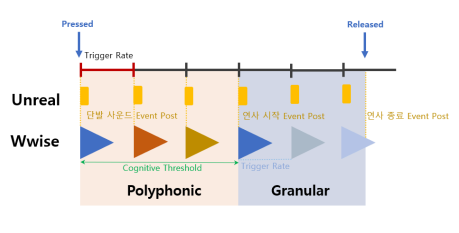


### Granular 방식



# 개선 플레이백 시스템

## Hybrid (Polyphonic + Granular) 방식

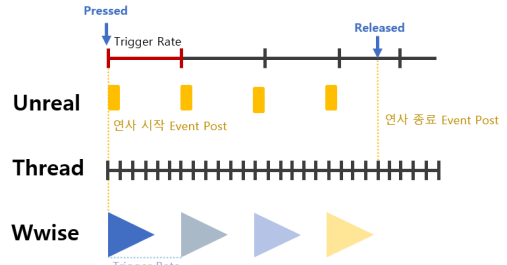
 <p>Unreal</p> <p>Wwise</p> <p>Polyphonic</p> <p>Granular</p>	<p><b>재생 방식</b></p> <ul style="list-style-type: none"><li>이번 R&amp;D에서 고안한 플레이백 시스템 1</li><li>Polyphonic 방식과 Granular 방식을 혼합하여 Granular 방식의 초기 발사음 매칭 문제를 인지적인 측면에서 Masking한 시스템</li><li>발사 개수가 늘어남에 따라 사람의 인지 경계(Cognitive Threshold)를 기준으로 Polyphonic 방식에서 Granular 방식으로 전환함</li><li>상세 내용: [공유] Hybrid (Polyphonic + Granular) 플레이백 시스템</li></ul>
	<p><b>개선된 점</b></p> <ul style="list-style-type: none"><li>사람의 인지 측면에서 초기 발사음 개수 매칭 문제를 해결할 수 있음</li><li>Polyphonic 구간 이후부터는 일정한 간격의 연사음 표현이 가능</li></ul>
	<p><b>단점</b></p> <ul style="list-style-type: none"><li>Polyphonic 구간에서는 Frame Rate 의존성 문제로 일정한 간격의 연사음 표현이 어려움</li></ul>

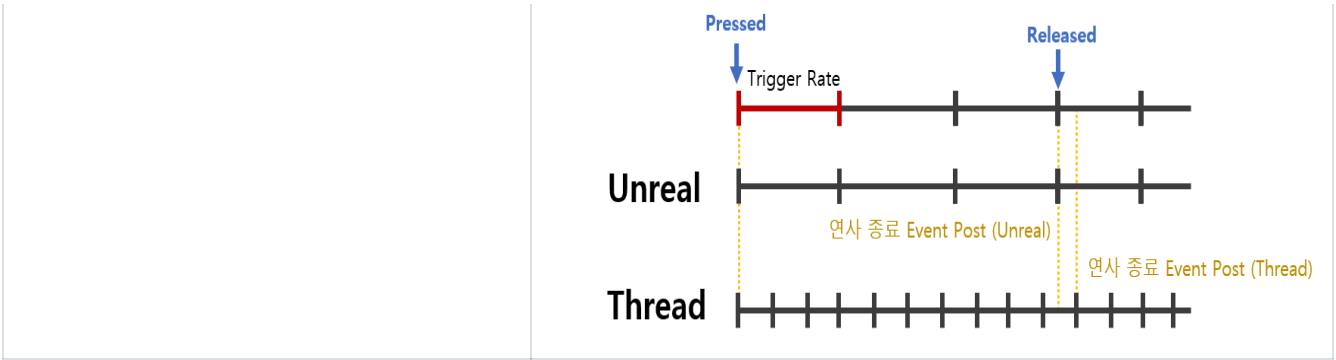
## Granular (Wwise Sync) 방식

 <p>Unreal</p> <p>Wwise</p>	<p><b>재생 방식</b></p> <ul style="list-style-type: none"><li>이번 R&amp;D에서 고안한 플레이백 시스템 2</li><li>총알의 발사를 Wwise Time 에 맞춘 시스템<ul style="list-style-type: none"><li>Rapid 사운드 재생 마다 delegate 작동-&gt; projectile spawn</li></ul></li><li>상세 내용: Granular (Wwise Sync) 플레이백 시스템</li></ul> <p><b>개선된 점</b></p> <ul style="list-style-type: none"><li>Frame rate에 의존적이지 않음</li><li>일정한 간격의 연사 표현이 가능</li><li>모든 경우에서 개수 매칭 문제가 발생하지 않음</li></ul> <p><b>단점</b></p> <ul style="list-style-type: none"><li>발사 로직에 Wwise Time 에 대한 Dependency 가 존재</li></ul>
---	---

## 추가 아이디어

### Thread 분리 방식

 <p>Unreal</p> <p>Thread</p> <p>Wwise</p>	<p><b>재생 방식</b></p> <ul style="list-style-type: none"><li>언리얼 Frame Rate가 가변적이라는 특징에 주목하여 고안한 시스템</li><li>더 고정적이고 빠른 프레임 레이트를 제공해줄 수 있는 스레드를 생성, 해당 스레드에서 연사 종료 Event를 제어</li><li>새 Thread에서 지속적으로 언리얼의 입력 변화를 감시하면서, Released가 발생했을 때 연사 종료 Event Post를 수행</li></ul> <p><b>이슈</b></p> <ul style="list-style-type: none"><li>언리얼에서 입력 상태 변화를 가져오기 위해서는 결국 언리얼 프레임에 의존해야 한다는 한계가 존재<ul style="list-style-type: none"><li>즉, 실제로는 왼쪽의 구상도대로 동작하지 않음</li><li>이에 따라, 기존 Granular 방식에 비해 뚜렷하게 개선된 점이 존재하지 않음</li></ul></li><li>입력 변화 이벤트를 Tick 초기에 감지하여 이벤트를 동기적으로 Post할 수 있는 기존 방식이 Thread 분리 방식에 비해 Wwise에서 더 빠르게 처리되는 경우도 존재<ul style="list-style-type: none"><li>&gt; 예시를 보시려면 클릭해주세요.</li></ul></li></ul>
--	---



재생지연 + 카운트다운 방식

**재생 방식**

- 사전 작업: Fire 사운드 마다 0번 샘플에 FireStart 마커 배치
- On Input Pressed: 최대 0.15초 이후에 Rapid Fire Play 이벤트 호출
- On Input Released: Fire Count 값과 Audio Count 값을 비교하여 모자란 사운드 개수 만큼 더 재생한 뒤 Break 이벤트 호출

[Gun\\_DelayWithCountdown\\_Overall.mp4](#)

**개선된 점**

- 균일한 발사음 간격 보장
- 인지 가능한 범위 내에서 발사음 개수 일치 (10발 이하)

**이슈**

- RPM 900 이상 총기에서는 로직 상 적절한 시점에 Break 이벤트를 호출해도 사운드가 한 번 더 재생되는 이슈가 있음
  - 언리얼-Wwise 통신에 시간이 소요되는 관계로, 사운드 재생 시작 marker 콜백을 받는 동안 사운드가 한 번 더 재생되는 것으로 추정
- Fire 이벤트 호출 시점에 이미 Input released & Fire count = 1 이면 즉각 Break 이벤트를 호출하는 방식으로 임시 처리 → 2발 이상인 경우 여전히 사운드가 한 번 더 재생됨

Demo 프로젝트

- 해당 문서에 소개된 여러 플레이백 시스템을, 데모 프로젝트에서 비교해가며 테스트를 수행해보실 수 있습니다.
- 자세한 사용법은 아래 문서를 참고해주세요!
  - [총기연사음 R&D Demo 프로젝트](#)

결론

총기연사음 플레이백 시스템	Frame Rate 의존성	발사음 매칭 여부	균일한 발사음 간격 보장	사운드 품질	구현 난이도	영상	총평
Polyphonic	O	O	X	Low	1	<a href="#">Polyphonic.mp4</a>	구현이 간단하고 발사 개수와 사운드 개수가 일치함을 무조건 보장받을 수 있으나, 엔진 Frame Rate에 의존적인 특징 때문에 발사 간격이 일정하지 않아 사운드 역시 불균일하게 발생한다는 문제점이 있습니다.  또한, 사운드 소스로 단발 사운드만 사용하기 때문에 평면적인 사운드로 느껴질 수 있으며 Burst 사운드 표현이 힘들다는 문제점이 있습니다.
Granular	X	X	O	Middle	1.5	<a href="#">Granular.mp4</a>	개선 시스템에 비해서는 구현이 비교적 간단하고 Looping을 사용하기 때문에 엔진 Frame Rate와 독립적으로 균일한 사운드를 보장받을 수 있으며, Polyphonic과 다르게 Burst 등 여러가지의 사운드 표현이 가능합니다.  그러나, 게임 엔진과 사운드 엔진의 처리 시간 차로 인해 발사 개수와 사운드 개수가 불일치할 수 있다는 문제점이 있습니다.

총기연사음 플 레이백 시스템	Frame Rate 의존 성	발사 음 매 칭 여 부	균일한 발 사음 간격 보장	사운 드 품 질	구 현 난 이 도	영상	총평
Hybrid (Polyphonic + Granular)	△	△	△	Middle	2	<a href="#">Hybrid.mp4</a>	<p>Granular의 발사 개수 불일치 문제를 해결하 고자, 발사 개수의 인지 경계 아래에서는 Polyphonic을 적용하고, 경계 개수를 넘어갔 을 때만 Granular를 적용하는 방법입니다.</p> <p>기존 Polyphonic 방법의 평면적인 느낌을 해 소하기 위해, Polyphonic이 적용되는 동안에 는 첫 발사음과 이후 발사음을 구분하여 사 운드를 적용하였습니다.</p> <p>청각적으로는 Granular의 단점을 해결한 방 식이지만, 결국 기존의 Polyphonic과 Granular 방식을 혼합한 방식이기 때문에 발 사 개수의 인지 경계 아래에서는 Polyphonic 의 장단점을, 위에서는 Granular의 장단점을 그대로 가지게 됩니다.</p>
Granular (Wwise Sync)	X	O	O	High	2	<a href="#">Granular_WwiseSync.mp4</a>	<p>기존 Granular 시스템의 장점을 모두 가지면 서, 발사 개수와 사운드 개수가 불일치한 경 우를 보완한 방법입니다. 가장 안정적인 성 능과 사운드 품질을 제공하지만, 구현 난이 도가 다소 있으며 시스템을 사용하기 위한 설정이 다른 시스템에 비해 다소 번거로울 수 있습니다. (예를 들면, 사용하고자 하는 사운드 소스마다 마커 삽입을 진행해야 합 니다.)</p> <p>또한, 발사를 일반적인 엔진 Frame이 아닌 사운드 신호에 맞추는 방법이므로 상황에 따라서는 적용이 어려울 수 있습니다.</p>

- 결론적으로, 개선 플레이백 시스템 중 종합적인 성능과 안정성 측면에서는 Granular (Wwise Sync) 방식이 우세를 보이지만, 설정의 용이함 측면에서는 Hybrid (Polyphonic + Granular) 방식이 우세를 보이는 것으로 보입니다.

# 문의

- 궁금하신 사항이나 추가 피드백등이 있으신 경우 Sound Center 직속 TAD팀으로 문의주세요. 😊