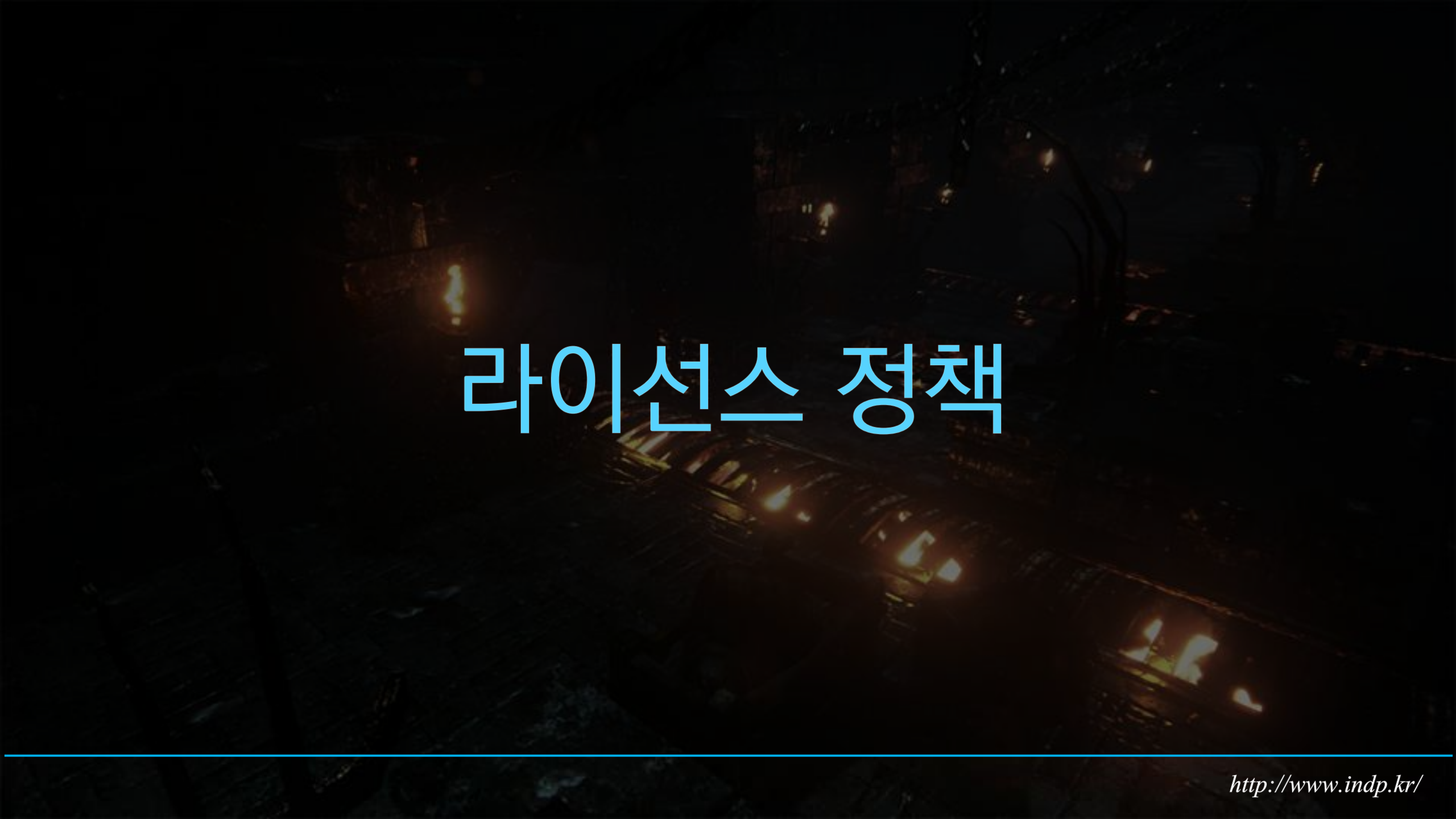


언리얼 엔진과 유니티엔진 무엇이 다른가?

Indie Developer Partners
Dustin Lee

An aerial night photograph of a city, likely Seoul, showing the Han River and surrounding urban areas. The city lights are reflected on the water's surface. The text '라이선스 정책' is overlaid in the center.

라이선스 정책

유니티 엔진의 라이선스



5 Professional

- Unity 5 engine with all features
- Team License
- Unity Cloud Build Pro
- Access to Betas/Previews
- Unity Analytics Pro (in Beta)
- Game Performance Reporting (in Preview)
- Unity Asset Store Level 11 Deals

**Starting at \$75/month
or \$1,500
No Royalties**

5 Personal

- Unity 5 engine with all features
- Revenue/Funding less \$100,000

FREE

No Royalties



유니티 엔진의 라이선스

- 연 매출 1억원 이상의 기업 및 개인 사용자의 이용은 불법
- 무료로 사용하더라도 기타 AddOn 기능들은 Subscription
- 학교에서는 교육용 라이선스를 별도로 구매해야 함.
- 구매 모델에서 Subscription 모델로 점차 이동 중.

사실상 작년과 달라진 부분은 별로 없음.

언리얼 엔진의 라이선스



언리얼 엔진의 라이선스

- 5% 로열티 부과 (매출 기준)
- 분기별 300만원까지는 로열티 없음
- 단방향 미디어(애니메이션), 영화, 건축, 오락실 등에는 로열티 부과 없음.
- 풀 소스코드 제공 (GitHub)
- 무료 교육 적극 장려

작년에 비해 월 치킨 한 마리 가격이 절약됨.

유니티 엔진 4 vs 언리얼 엔진 4 퀄리티

- 아티스트의 역량이 제일 중요!
- 렌더링 방식의 차이
 - 손맵 vs 물리 기반 렌더링 시스템 (Physically Based Shading)
 - 반사 효과의 손쉬운 구현
 - 적은 작업으로 사실적이면서도 자연스러운 렌더링의 표현 가능
 - 성능이 많이 필요함
- Global Illumination의 구현
 - 3rd Party Solution vs 자체 솔루션
 - 라이트맵부터 라이팅까지 일관된 렌더링 파이프라인

유니티 5의 등장

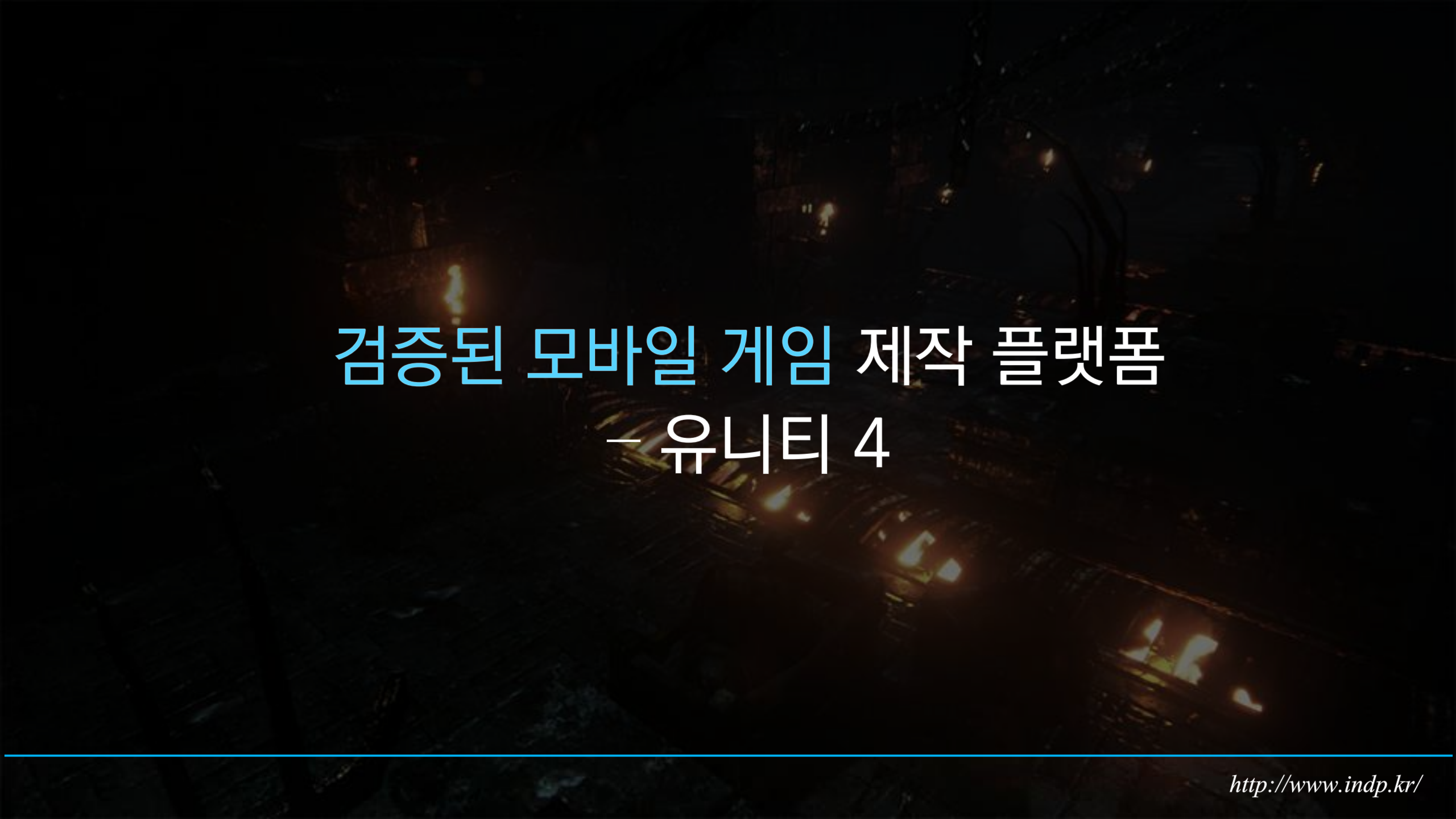
- 안드로이드 기기의 발전이 여전히 더딘 수준에 머무르고 있음
- 유니티 5에서 물리기반 렌더링 시스템 탑재
- 라이트맵 솔루션의 강화 (Beast -> Enlighten)

스펙 상으로만 본다면 언리얼 4의 주요 기능을
유니티 5에서 커버하는 상황.

우리는 왜 언리얼 엔진을 선택했는가?

- 사실 유니티로 개발하는게 너무 편함.
 - 서버 연동은 유니티로 더미 클라 만들어서 하고 있음
- 하지만 왜 언리얼로 게임을 만드는가?
 - 사실 유니티로 대부분의 모바일 게임을 충분히 만들 수 있다.
 - 개발 과정의 다양성에 대한 기대감
 - 새로운 기술에 대한 개인적인 호기심
 - 적은 인원으로 고퀄리티 그래픽의 표현
 - 효과적인 확장성 (Scalability)

완전 비효율적인 선택!

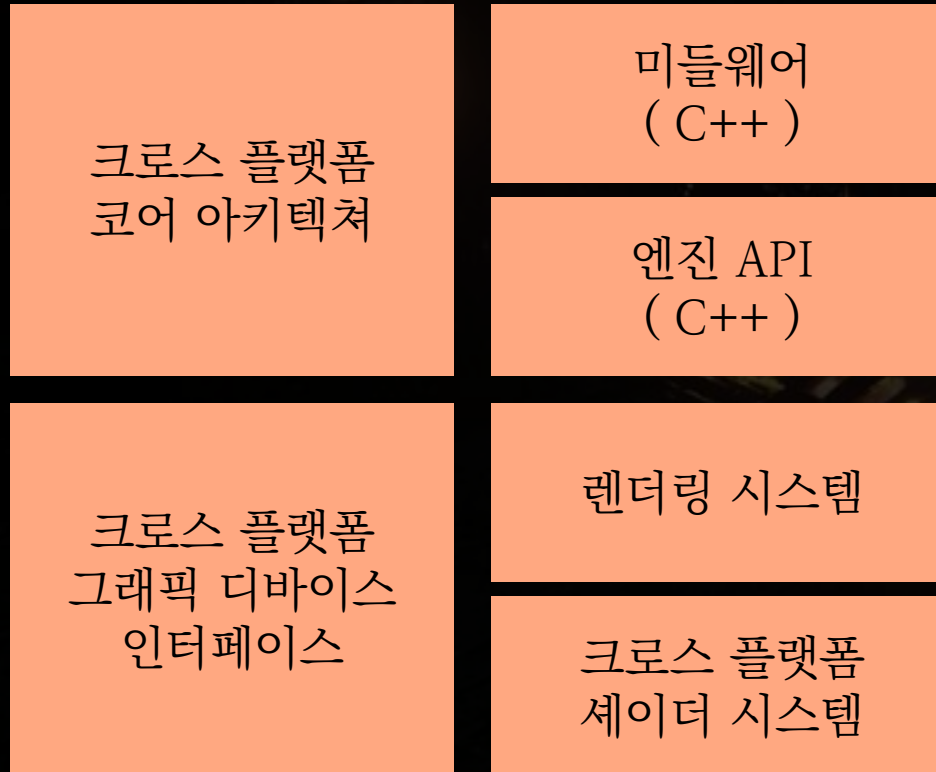


검증된 모바일 게임 제작 플랫폼

– 유니티 4

유니티 게임 엔진의 구성

엔진 레이어



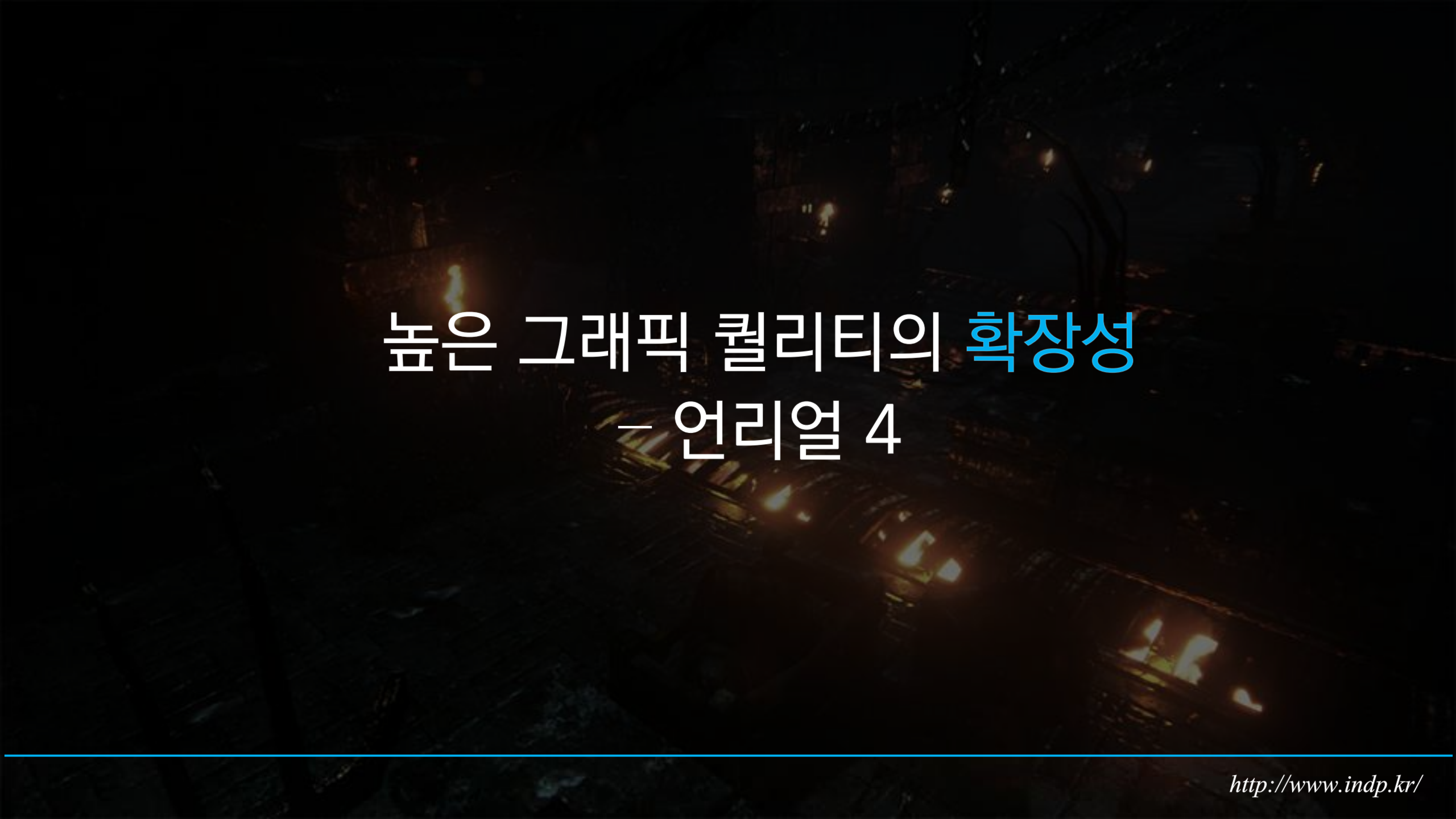
사용자 레이어



모바일 플랫폼에서 검증된 유니티 엔진

- 멀티 플랫폼을 지원하는가? (안드로이드 , iOS)
- 게임 제작에 필요한 보조 기능을 제공하는가? (UI, 다국어 지원, HTTP 통신)
- 완성도 높은 에디터로 높은 생산성을 내는가? (원클릭 빌드 및 리소스 관리)
- 최종 산출물이 적절한 용량으로 에러 없이 만들어지는가? (리소스 패킹)
- 플랫폼 게임 서비스 연동이 잘 되는가? (구글 플레이 서비스)

수 년간의 노하우로 완성된 모바일 게임 제작 워크플로우



높은 그래픽 퀄리티의 확장성

- 언리얼 4

멀티 플랫폼의 미래



스마트 폰과 데스크탑, 콘솔 사이 무너지는 경계

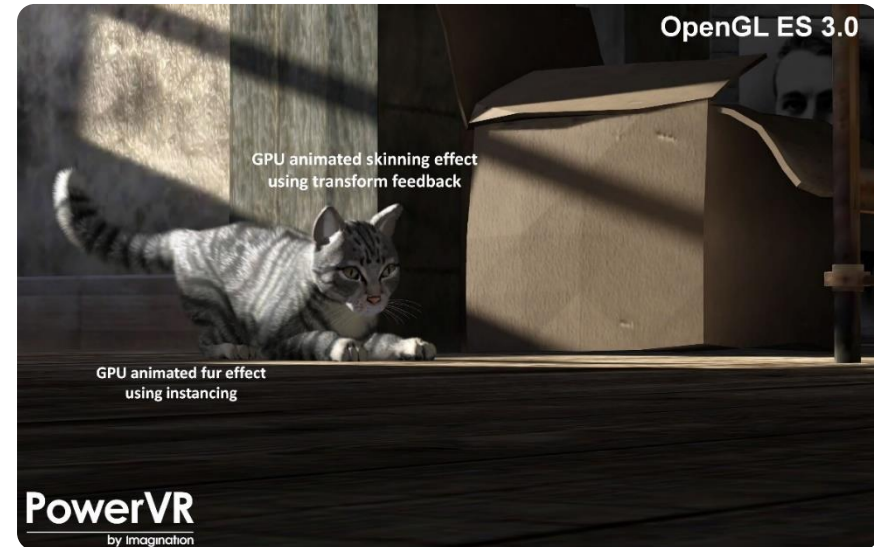
성장하는 모바일 기술



iPhone 5S

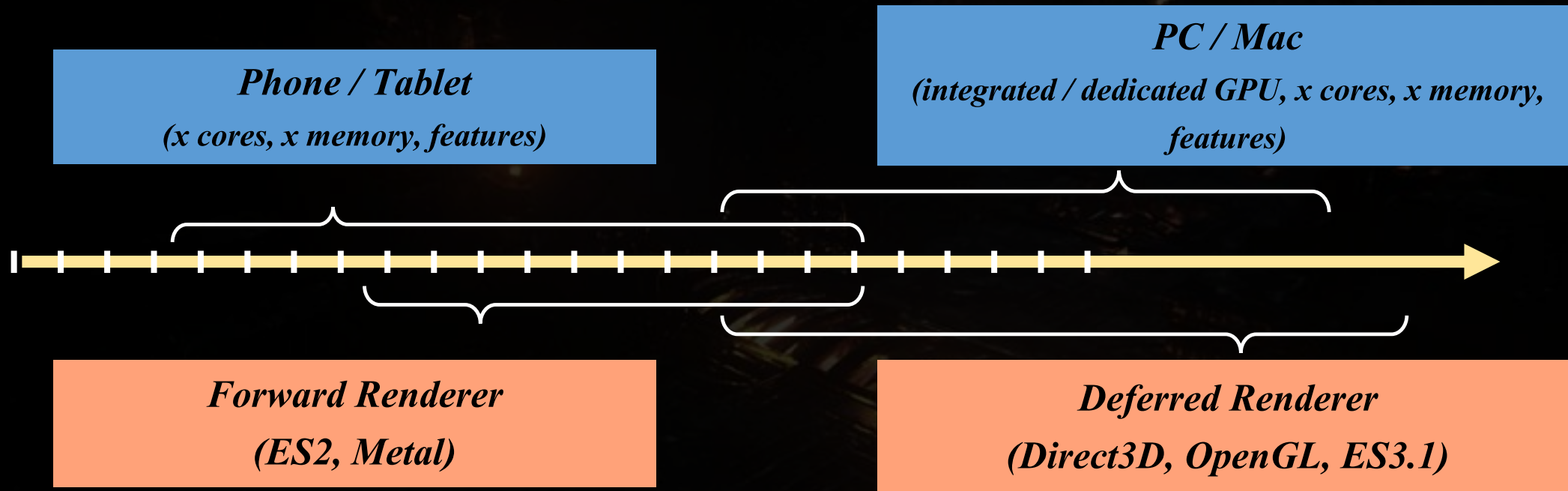
iPhone 6

iPhone 6 Plus



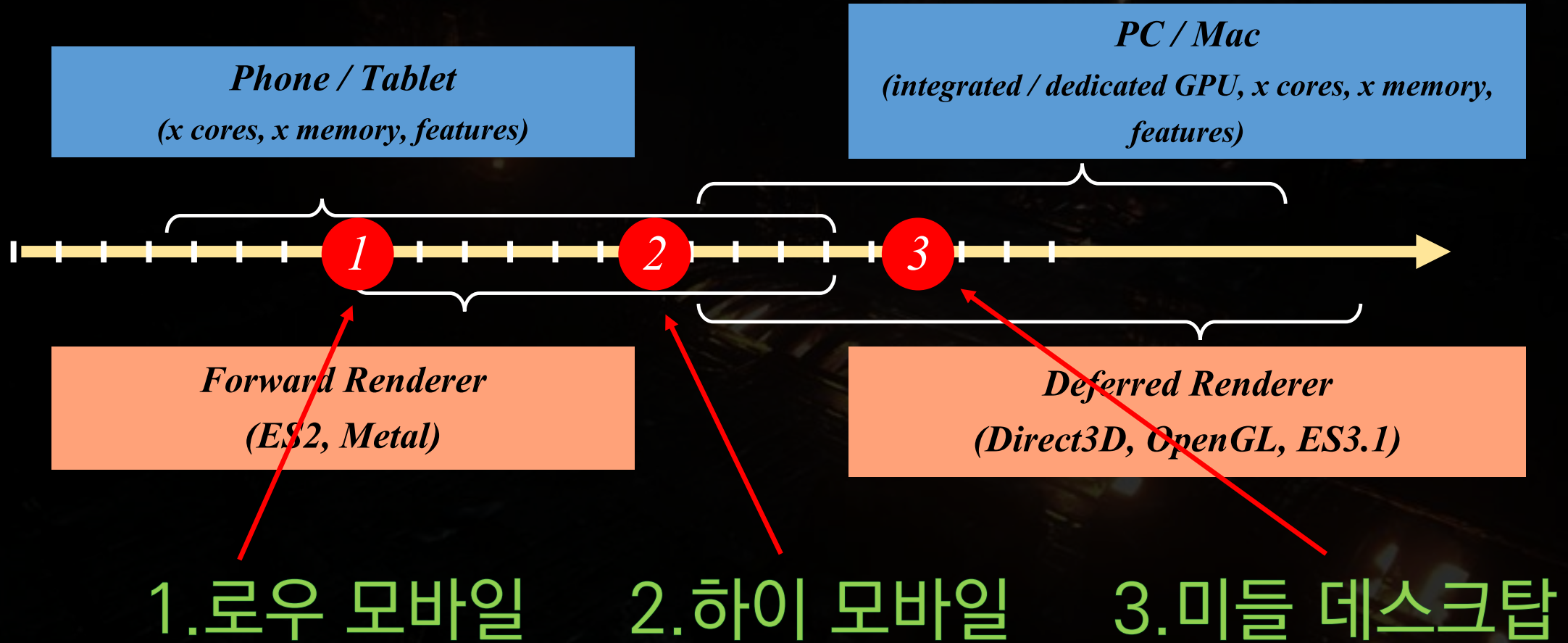
저사양 폰도 고려하면서 미래를 대비

하드웨어 플랫폼에 따른 그래픽 기술 스펙트럼



하나의 워크 플로우에서 모두 커버할 수 있을까?

이번 프로젝트의 목표



언리얼 4의 물리기반 렌더링 사용기

- 최적의 씬을 설정하기 위해 많은 시행 착오를 거침
 - 목표 : 저 사양 모바일 폰까지 커버할 수 있는 설정
 - 에픽에서 제공하는 다양한 모바일 샘플의 씬 설정 분석
- 결론
 - Mobile HDR 옵션에 따라 많은 퀄리티 차이 발생
 - 물리 기반 렌더링(PBR)은 환상적이지만 성능에 대한 반대 급부가 있음
 - 현재 하드웨어에서는 무겁지만 포기하면 언리얼 엔진을 쓰는 이유가 없어진다.

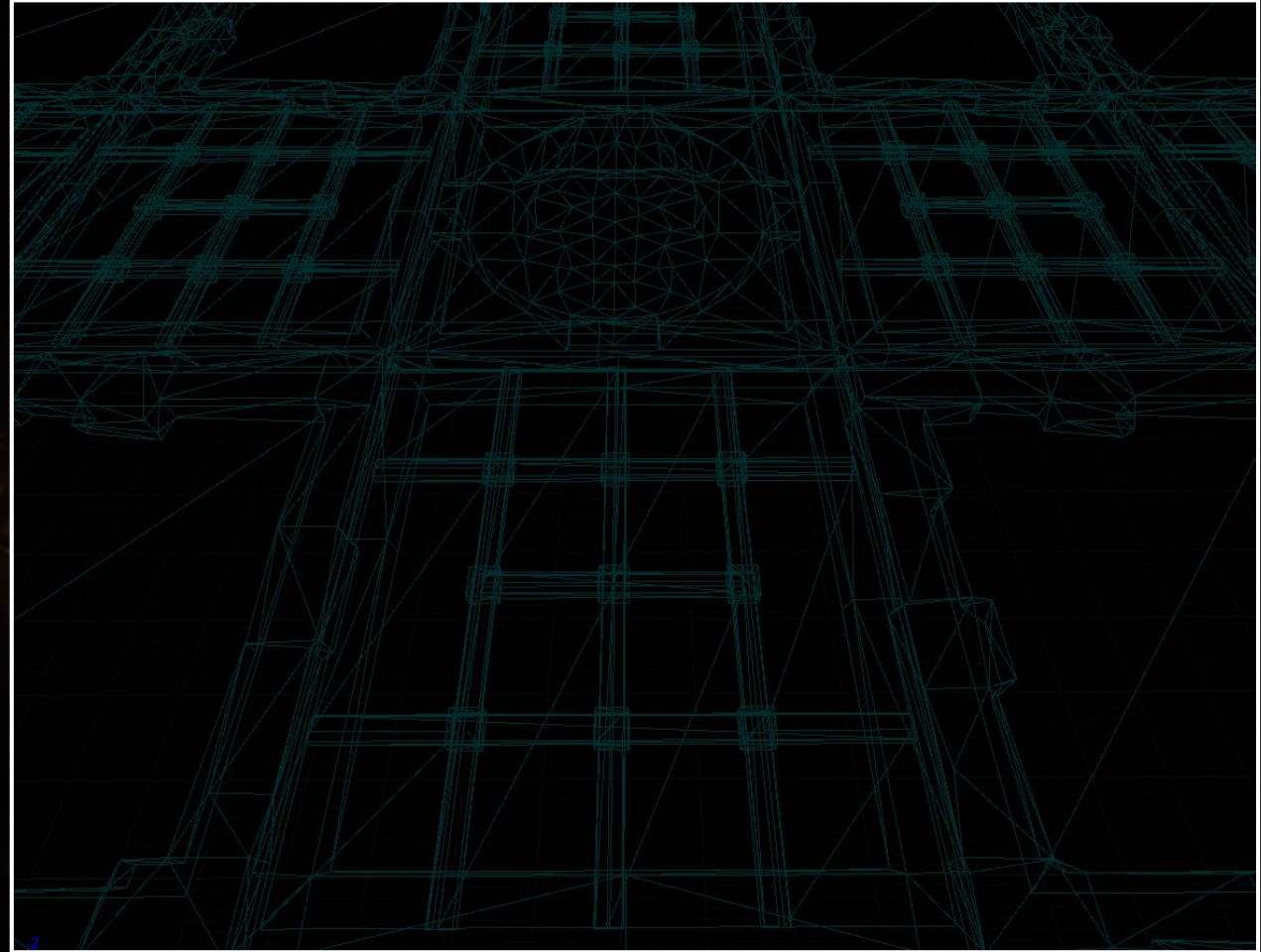
모두 다 안고 갈 수는 없다. 버릴 것은 과감하게..

언리얼 4를 위한 최적화 작업

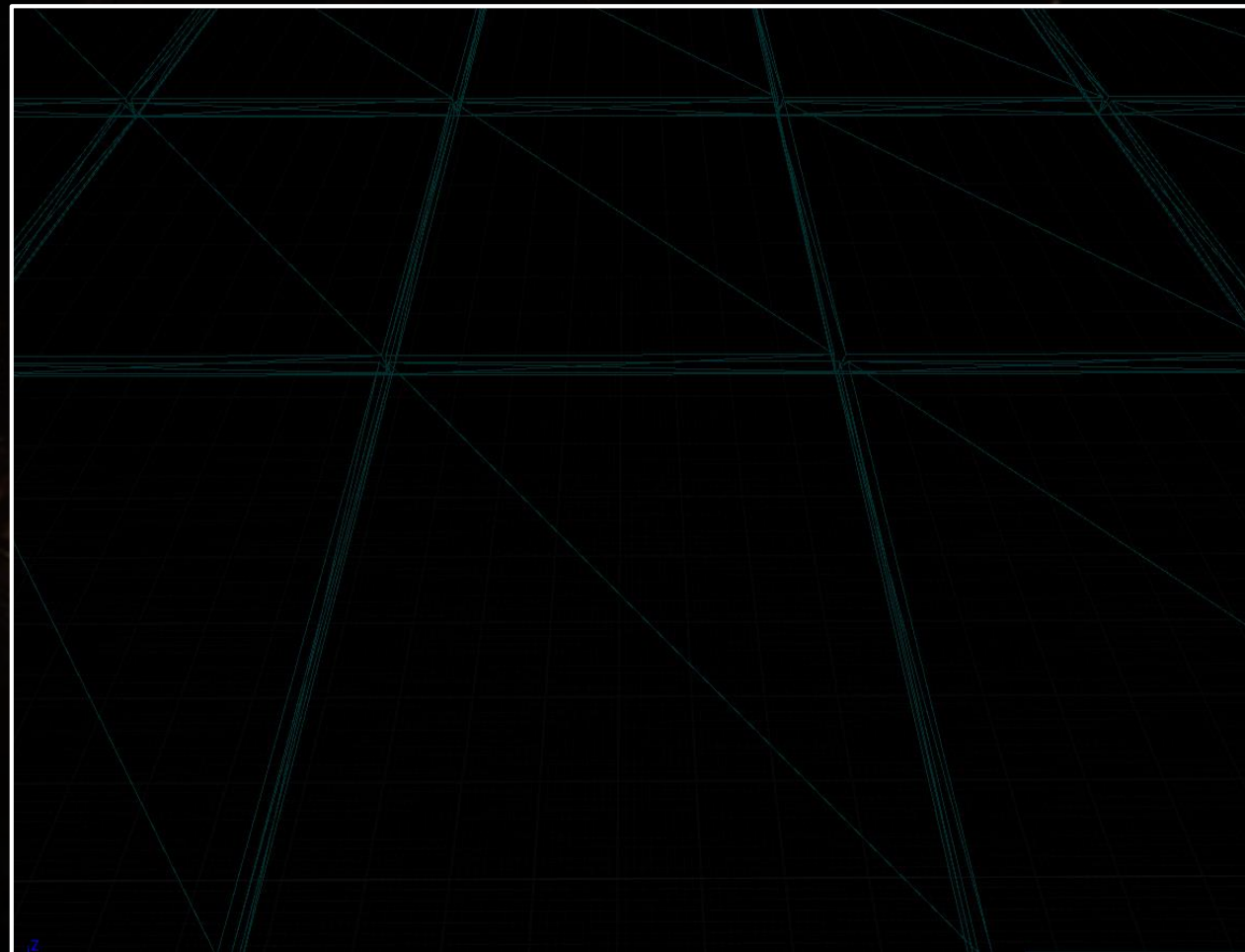
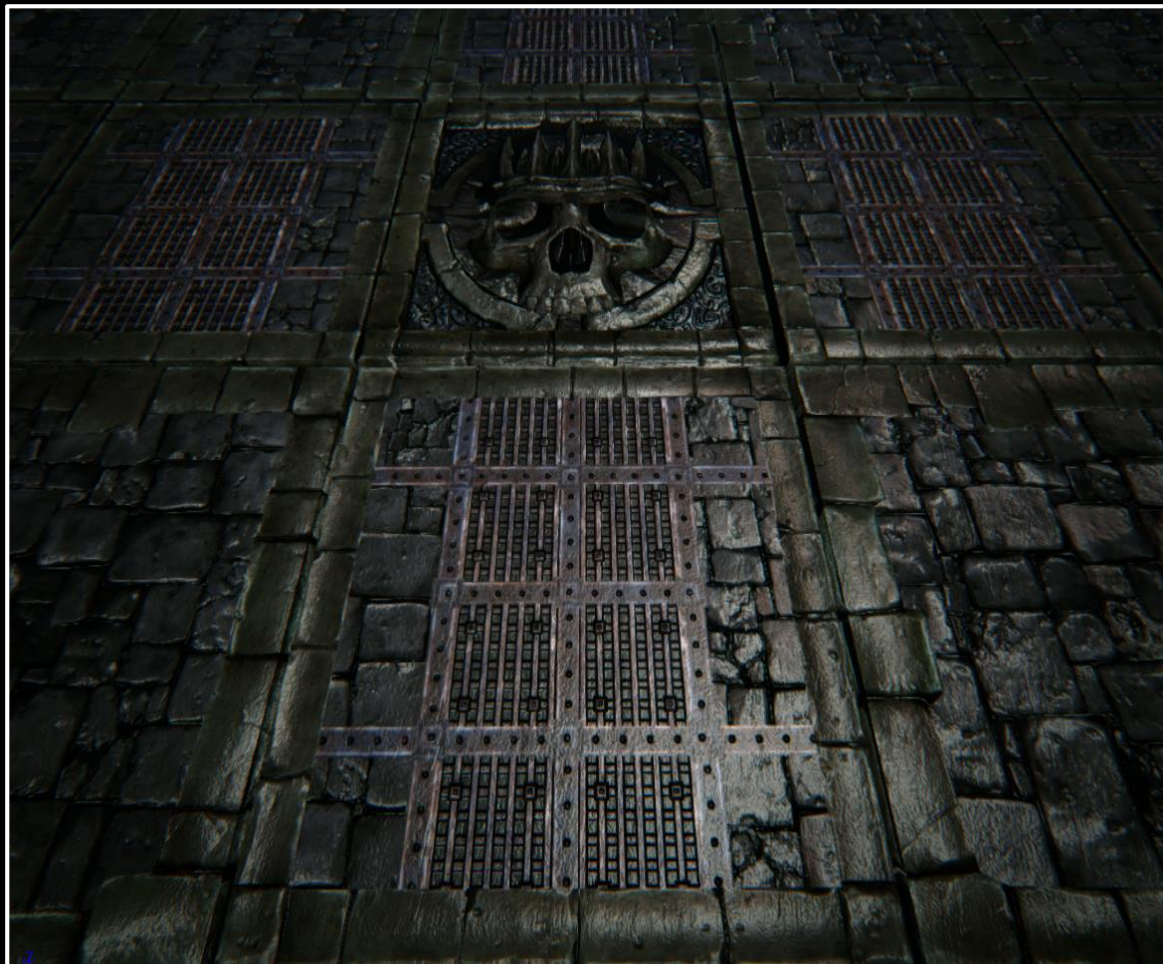
- 머티리얼의 설계
 - 여러가지 저렴한 방법의 시도
 - PBR로 노멀 느낌을 극대화하는 데만 치중
 - 부족한 부분은 이펙트로 둘러치기
- 모델링 데이터(Vertex)를 최소화 하기
 - 스태틱 메시 데이터 최적화
 - 스켈레탈 메시 데이터 최적화

추가로 컬링을 설정하니 갤럭시 S2에서도 돌아감!

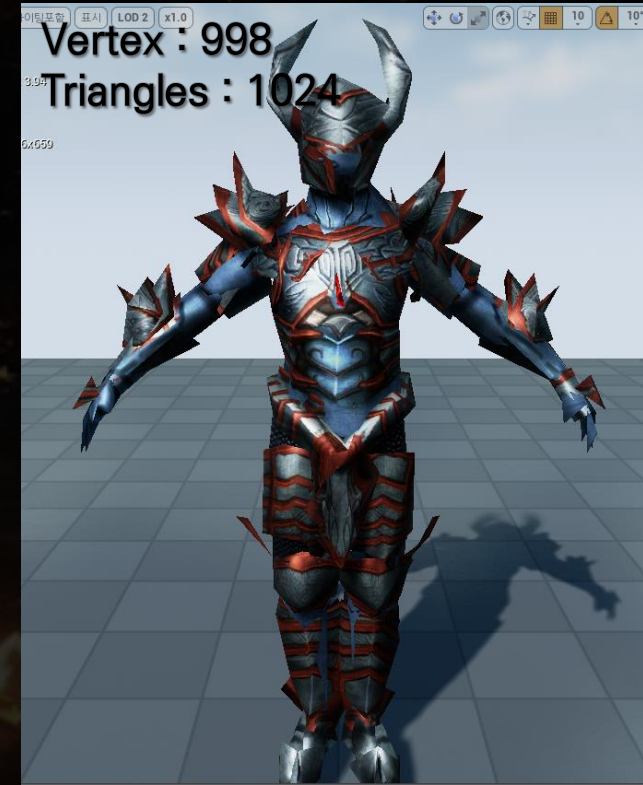
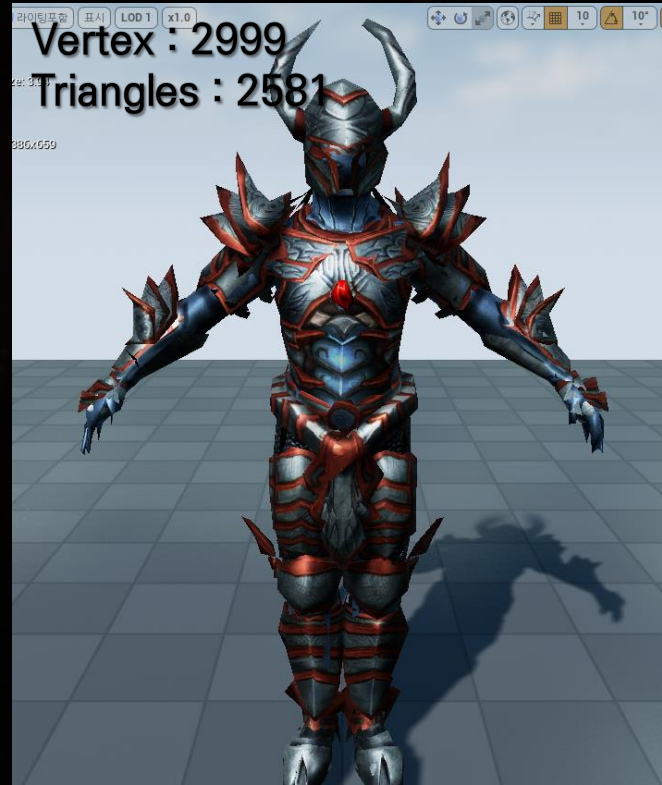
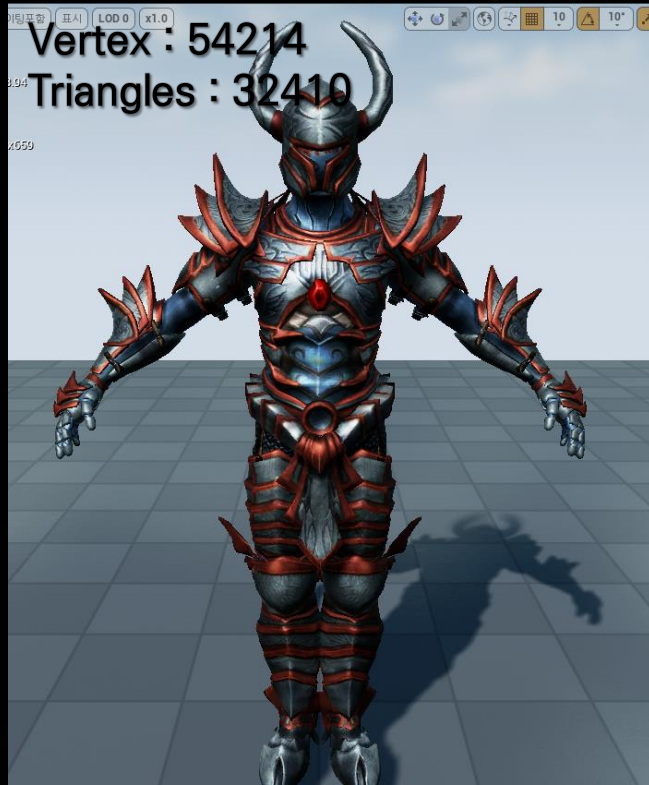
데이터 최적화 예시 - 스태틱 메시



데이터 최적화 예시 – 스택틱 메시

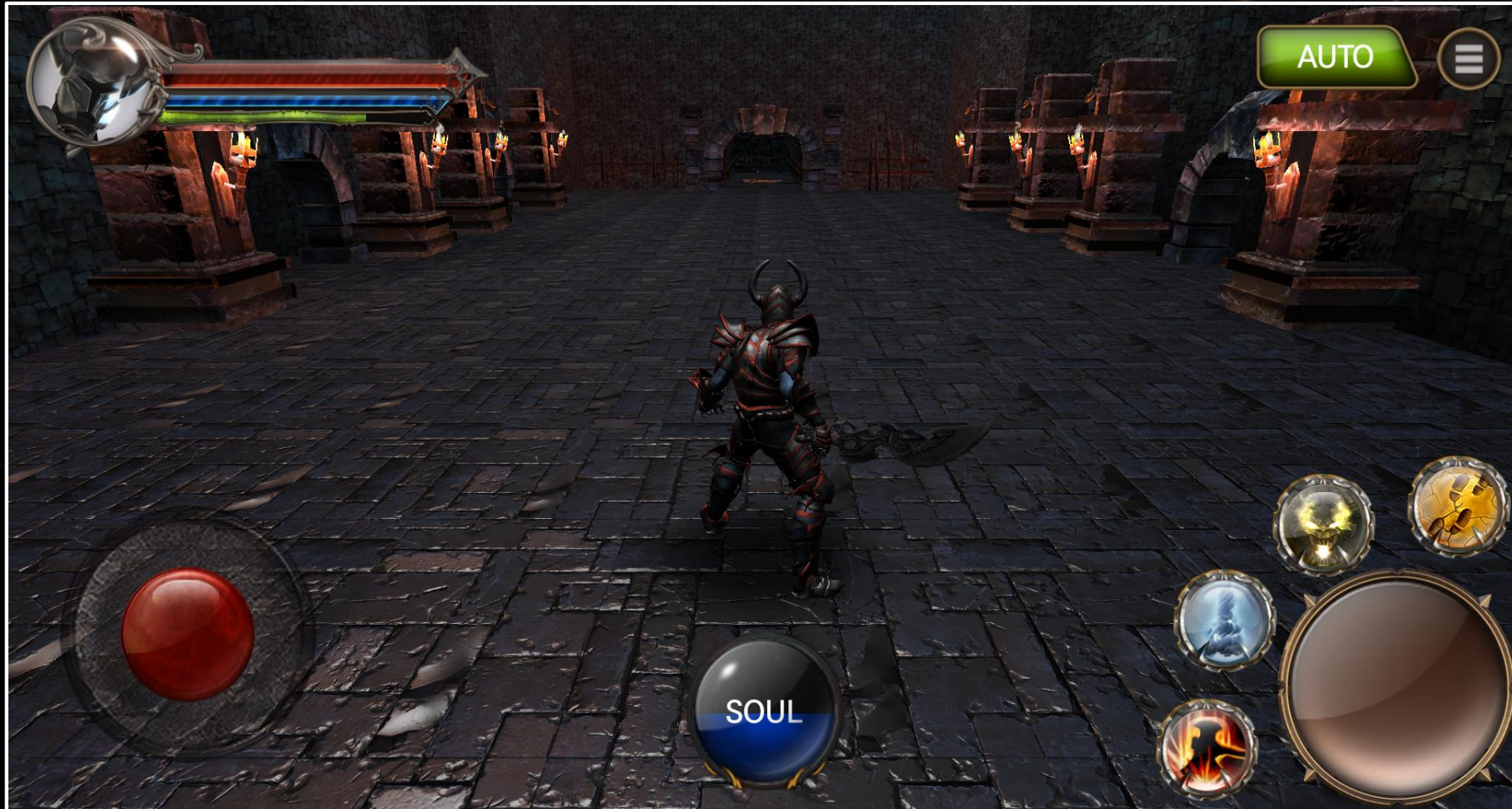


데이터 최적화 예시 - 스켈레탈 메시

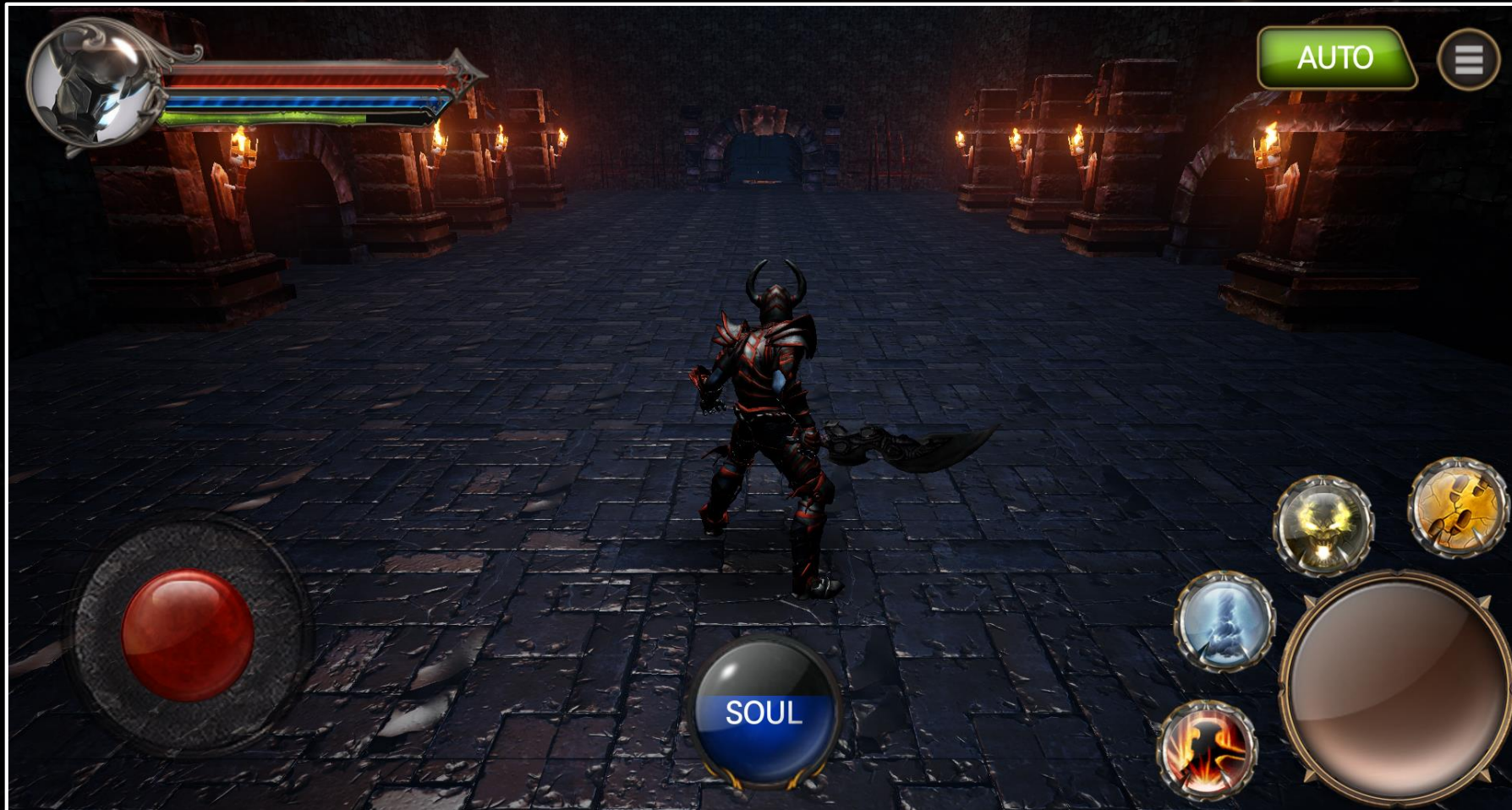


Simpolygon을 활용한 작업양 절약.

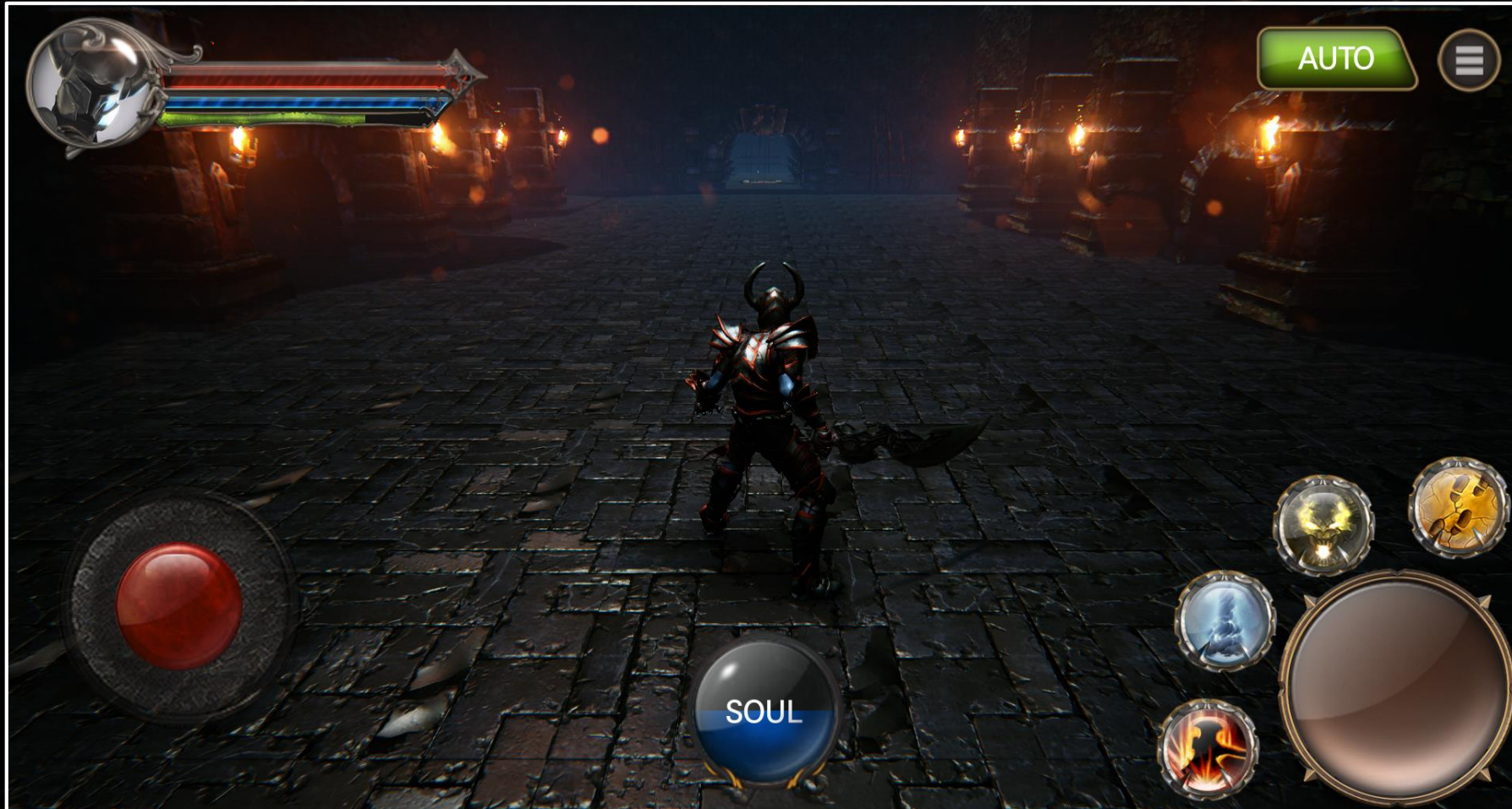
퀄리티 레벨 1 – 안드로이드 S4 이하



퀄리티 레벨 2 – 노트 4 이상, 아이폰 6



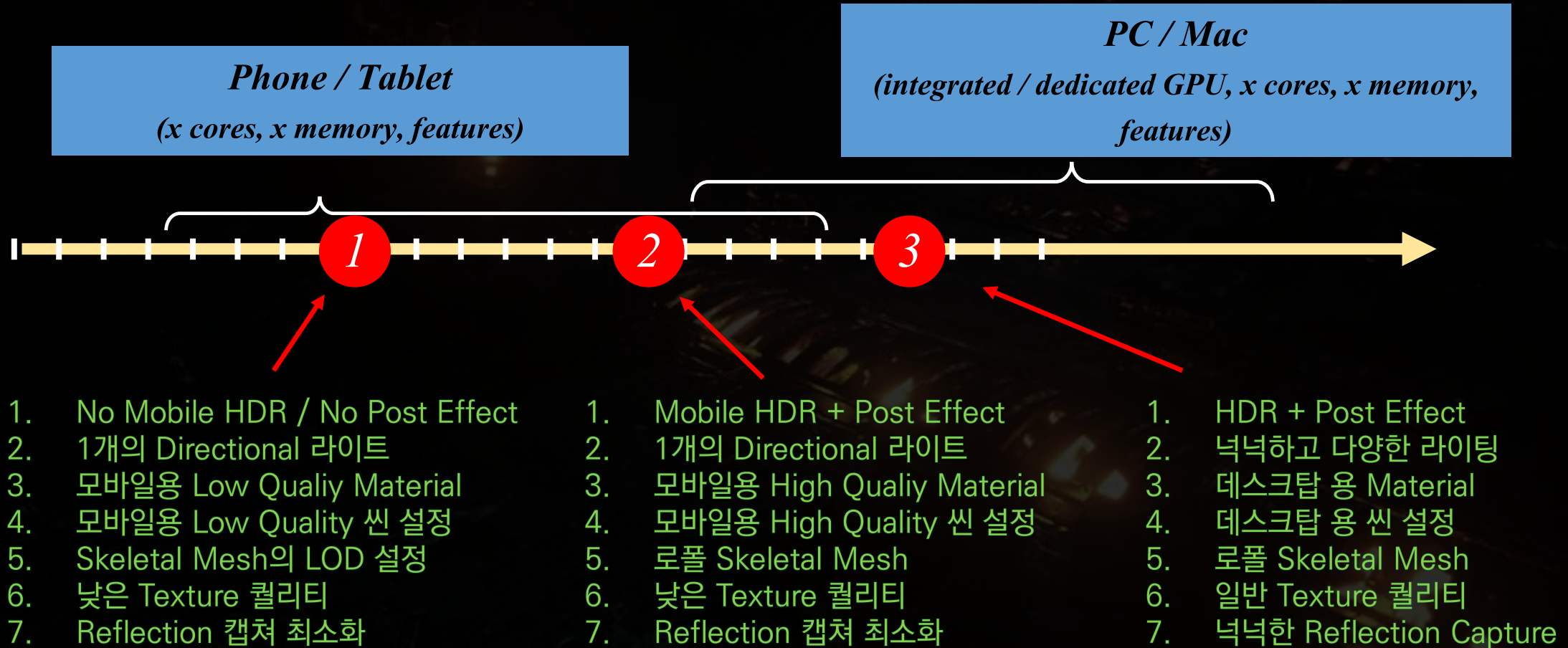
퀄리티 레벨 3 – 데스크탑 , 콘솔

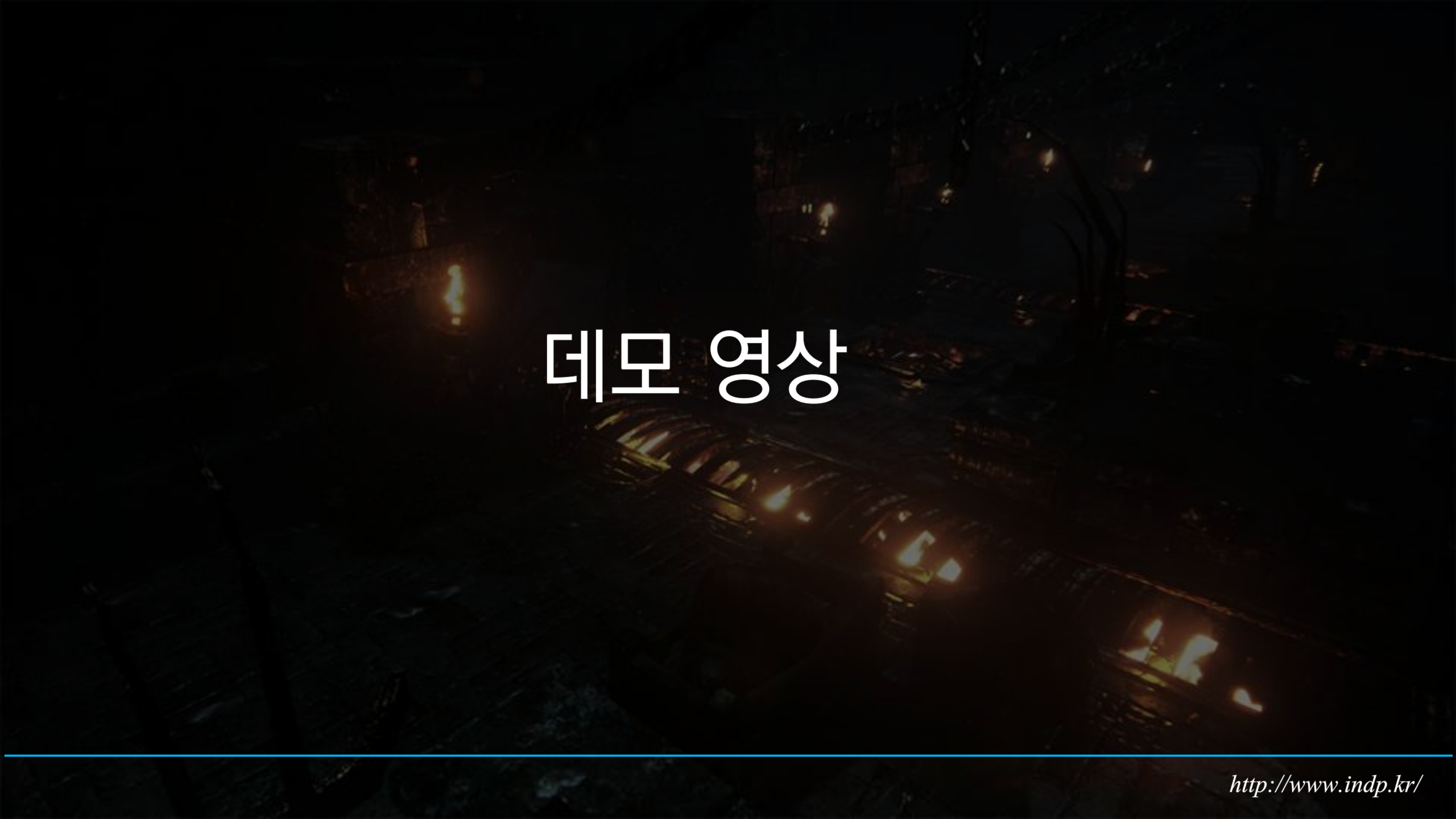


퀄리티 레벨 3 – 데스크탑 , 콘솔



하드웨어 타겟에 따른 스케일 설정

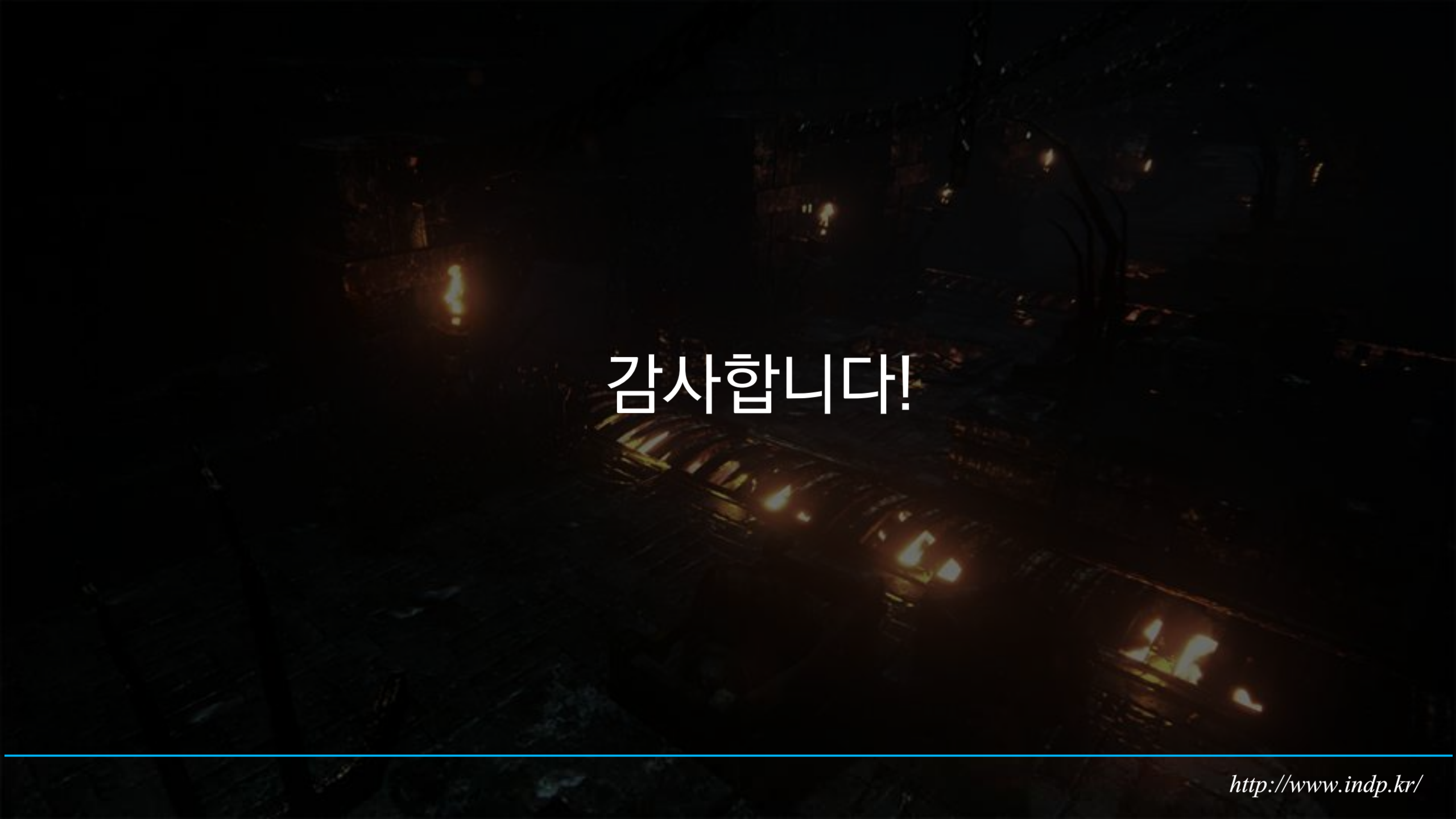




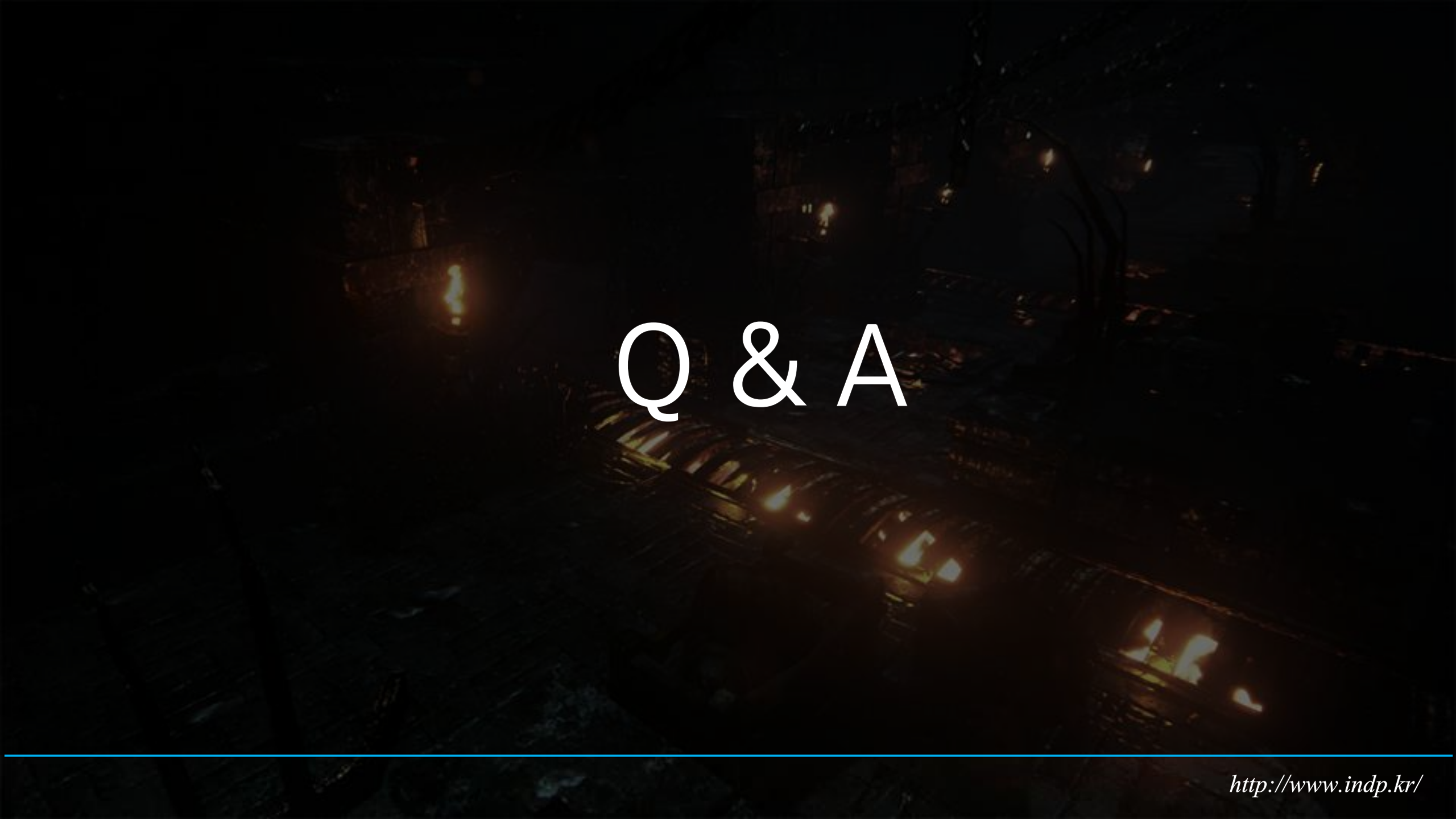
데모 영상

결론

- 유니티의 장단점
 - 생산성 높은 3D 모바일 게임 프로젝트 제작 검증
 - 평범한 그래픽 퀄리티 - 유니티 5를 통해 이를 보완
 - 소스코드 제공 없음 - 제작 파이프라인이 고정되어 있음
 - 개발에 대한 Risk 없이 독특한 콘텐츠로 승부
 - 개발자 구하기 쉬움
- 언리얼의 장단점
 - 검증된 AAA 렌더링 파이프라인
 - 안드로이드에서는 아직 보완할 점이 많은 솔루션 - 빠르게 진화 중
 - 소스코드 제공 - 기존에 없는 새로운 파이프라인
 - 엔진을 믿고 기존에 없는 새로운 퀄리티로 승부
 - 개발자 구하기 어려움



감사합니다!



Q & A