Chapter 1 소개

튜토리얼에 대하여…

이 튜토리얼은 Vulkan 그래픽스와 Compute API에 대해 가르칠 것입니다. Vulkan은 Khronos 그룹의 새로운 API로 현대(Modern) 그래픽 카드에게 더 향상된 추상화(Abstraction)을 제공할 것입니다. 이 새로운 인터페이스는 당신의 응용 프로그램이 더 좋은 성능을 이끌어내고, OpenGL과 Direct3D와 같은 API와 비교해서 덜 예외적인(Less surprising) 행동을 할 수 있도록 작성하기 쉽도록 제공되고 있습니다. 이러한 Vulkan의 아이디어는 Direct3D 12와 Metal과 비슷할 수 있습니다. 하지만 Vulkan은 완전한 크로스 플랫폼으로써 Windows, Linux 그리고 Android에서도 동시에 개발할 수 있다는 점에서 장점(Advantage)을 가지게 됩니다.

하지만 이러한 장점(benefits)를 취하기 위해서는 굉장히 장황한 API작업을 해야 합니다. 당신이 급조한 응용 프로그램에서도 프레임 버퍼 생성 초기화, 버퍼나 텍스처 이미지와 같은 메모리 관리자와 같은 그래픽 API와 관련해서 상세하게 작성(detail)해야 합니다. 그래픽 드라이버에 의존하는(less hand-holding)게 많이 줄어들기 때문에 올바른 행동을 할 수 있도록 더 많은 작업을 해야 합니다.

여기서 전달하는 메시지는 Vulkan은 모두를 위한 API가 아니라는 것입니다. 이는 고성능의 컴퓨터 그래픽에 열광하는(enthusiastic) 프로그래머와 기꺼이 그렇게 하려는 사람(are willing to put some work in)을 타겟으로 잡고 있습니다. 만약에 당신이 컴퓨터 그래픽스보다 게임 개발에 더 흥미가 있다면 Vulkan에 대한 흥미가 곧 떨어져 OpenGL이나 Direct3D을 다시 선호하게 될 수 있습니다. 다른 방법이라면 Unreal 엔진이나 유니티는 Vulkan을 사용할 수 있을 뿐만 아니라 더 높은 API를 제공할 것입니다.

이걸로 이제 이 튜토리얼을 따라오기 위한 준비물을 공개하겠습니다.

* Vulkan과 호환되는 그래픽 카드와 드라이버 (NVIDIA, AMD, Intel)
* C++에 대한 경험 (RAII와 초기화 리스트에 친숙)
* C++ 11이 호환되는 컴파일러 ( Visual Studio 2013 이상, GCC 4.8 이상)
* 3D 컴퓨터 그래픽스에 대한 어느정도의 경험

이 튜토리얼에서는 OpenGL이나 Directe3D의 개념(Concepts)에 대한 얘기는 하지 않을 것입니다. 그러나, 3D 컴퓨터 그래픽스에 대한 기초 지식은 갖추고 있어야 합니다. 원근 투영과 같은 수학적 배경을 설명하지 않기 때문에 컴퓨터 그래픽스 개념에 대해서는 온라인 자습서를 참조하는 게 좋습니다.

원한다면 C++ 대신 C를 사용할 수 있지만, 예제와 다른 선형 대수 라이브러리를 사용해야 하고, 본인만의 코드를 작성해야 합니다. 우리는 로직과 자원의 생존주기(Lifetime)을 설명하기 위해서 C++의 클래스와 RAII와 같은 기능을 이용할 것입니다.

E-book

E-book으로 이 튜토리얼을 읽고 싶으면 EPUB파일이나 PDF 파일을 다운로드 받을 수 있습니다.

* EPUB
* PDF

튜토리얼 내용(Structure)

우리는 첫 번째 삼각형을 화면에 띄우는 방법과, Vulkan의 작동 방식에 대한 개요로 시작하겠습니다. 모든 작은 단계의 목적은 당신이 전체적인 그림 안에서 기본적인 역할을 이해할 수 있도록 할 것입니다. 다음으로, Vulkan SDK와 선형 대수 계산을 위한 GLM 라이브러리, 창 생성을 위한 GLFW로 개발 환경을 설정할 것입니다. 튜토리얼은 Windows에서 Visual Studio, Ubuntu Linux에서 GCC를 이용한 설정 방법을 알려 줄 것입니다.

그리고 삼각형을 그리는데 필요한 Vulkan 프로그램의 요소들을 모두 구현할 것입니다. 각 챕터는 간략하게 다음과 같은 구조를 따르게 됩니다.

* 새로운 개념에 대한 소개와 목적 소개
* 관련 API를 모두 호출하고 프로그램에 통합
* 추상화 된 헬퍼 함수 작성

각 챕터는 이전 챕터의 후속 내용으로 작성이 되어있지만, Vulkan 기능을 소개하고 알아볼 수 있는 독립 챕터로써 읽을 수도 있습니다. 이는 사이트에서 유용한 참고 자료로 이용할 수도 있습니다. 모든 Vulkan 함수와 타입들은 사양서(Specification)에 링크되어 있으니, 더 알아보시고 싶으시면 클릭하시면 됩니다. Vulkan은 매우 새로운 API입니다. 그래서 사양서에 문제점(Shortcomings)가 있을 수 있습니다. 문제점을 발견하면 Khronos 저장소에 피드백을 남겨주시길 바랍니다.

이전에 언급했듯이, Vulkan API는 그래픽스 하드웨어를 최대한 통제하기 위해 많은 인자(Paramter)가 포함된 장황한 API를 가지고 있습니다. 이는 간단한 텍스처 생성과 같은 작업에도 많은 단계를 거치고, 매번마다 반복해야 하는 작업을 야기할 수 있습니다. 그러므로 우리는 우리만의 튜토리얼에서 사용할 헬퍼 함수들을 작성할 것입니다.

모든 챕터는 코드가 작성되어 있는 링크를 포함합니다. 만약에 당신이 코드 구조에 대해서 궁금증(Doubts)이 들거나, 디버깅 또는 비교를 하고 싶을 때 알아볼 수 있습니다. 모든 코드 파일은 다양한 인증된 공급자(Vendors)에서 테스트되었습니다. 또한 각 챕터의 마지막에서는 특정 주제에 대해서 질문을 할 수 있는 섹션이 있습니다. 플랫폼과, 드라이버 버전, 소스코드 그리고 예상했던 행동과 실제 행동을 작성하면 도움을 드리겠습니다.

이 튜토리얼은 커뮤니티의 영향을 위해서 작성되었습니다. Vulkan은 여전히 새로운 API이며, 대표적으로 좋은 예제는 아직 없습니다. 만약에 튜토리일과 사이트에 대해서 피드백이 있다면 망설이지 마시고 GitHub 저장소에 issuse나 pull request를 남겨주시길 바랍니다. 저장소에 Watch를 해두신다면 튜토리얼 업데이트에 대한 알림을 받으실 수 있습니다.

당신이 Vulkan의 삼각형을 화면에 띄우는 절차를 마친 후에, 선형 변환을 포함한 텍스처, 3D 모델작업을 시작할 것입니다.

이전에 그래픽스 API를 다뤄보았다면 첫 번째 지오메트리를 띄우는데 많은 작업이 있음을 알 수 있습니다. Vulkan에는 많은 초기화 단계들이 있지만, 각각의 단계들이 이해하기 쉬울것이고, 중복되지 않다고 느낄 것입니다. 또한 보이는게 지루한 삼각형을 띄우고 나면 완벽히 그려진 3D 모델을 띄우는데 큰 노력이 들지 않으며, 각 단계를 넘을 때마다 보람참을 느끼는 게 중요합니다.

만약에 튜토리얼을 따라오던 중 문제를 맞닥뜨리게 되면, 첫 번째로 FAQ에서 당신의 문제와 해결법이 있는지 확인해보십시오. 이후에도 문제가 계속 된다면 가장 가까운 챕터의 질문 섹션에서 편하게 도움을 요청하세요.

고성능 그래픽스 API에 뛰어들 준비가 되셨나요? Let’s go!