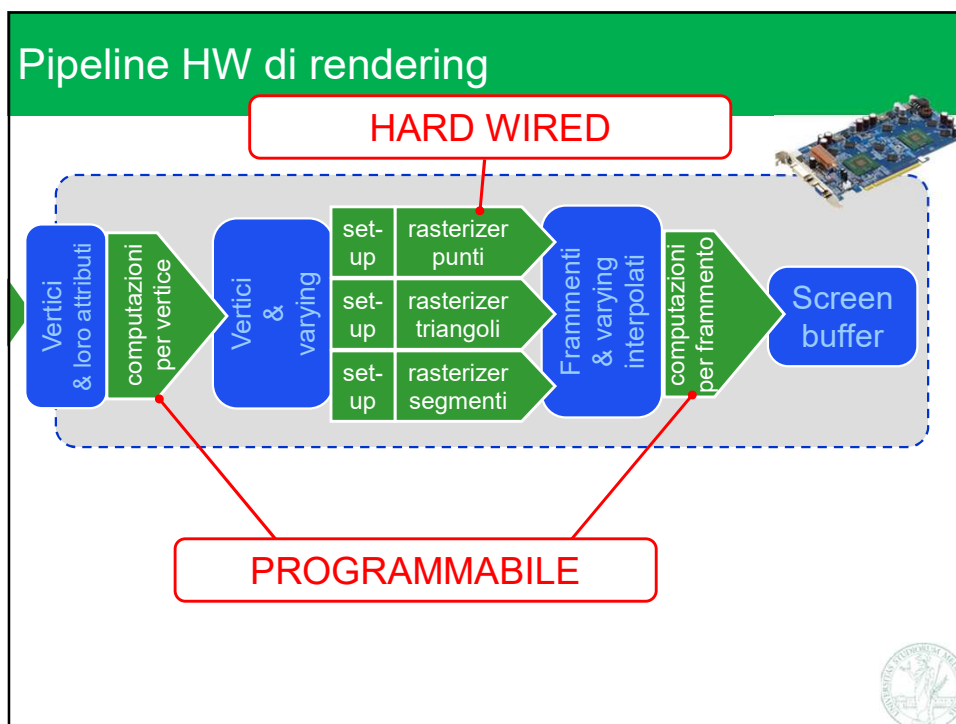
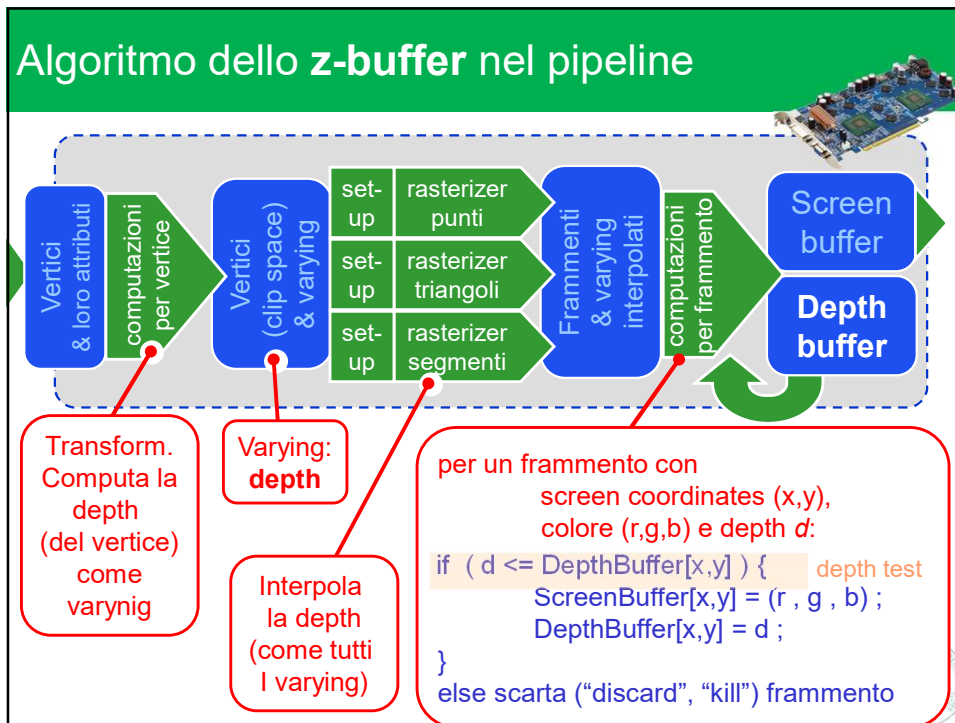


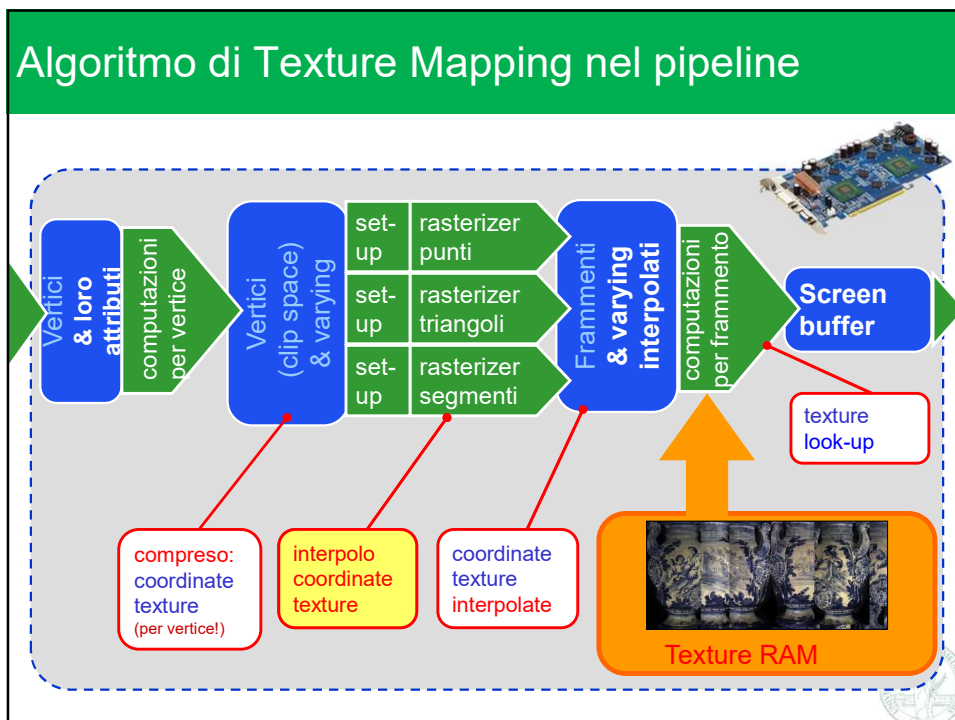
1



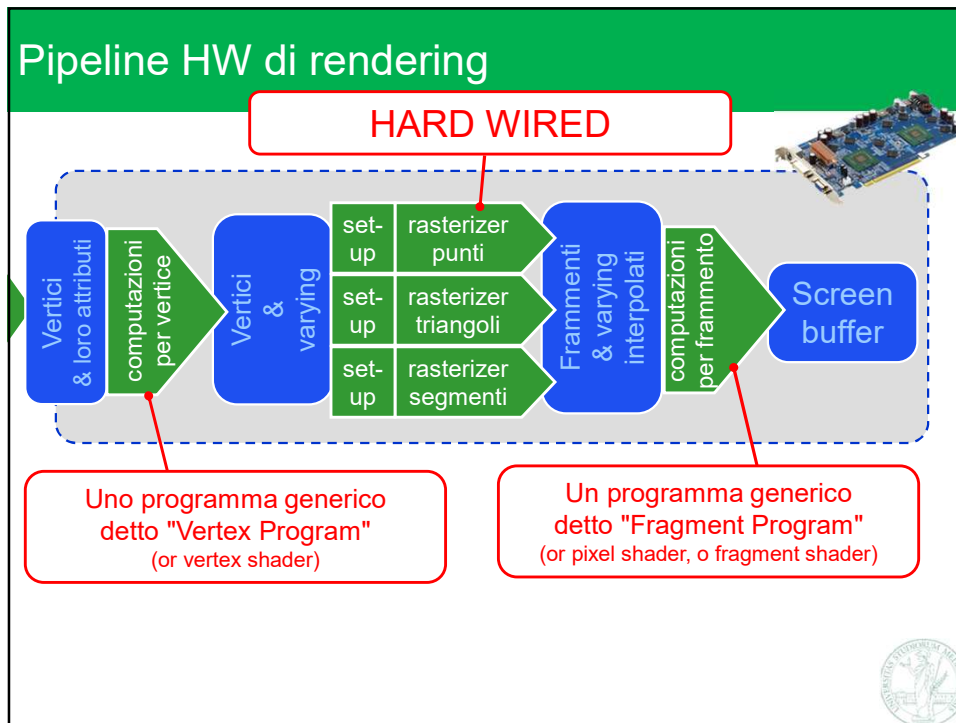
2



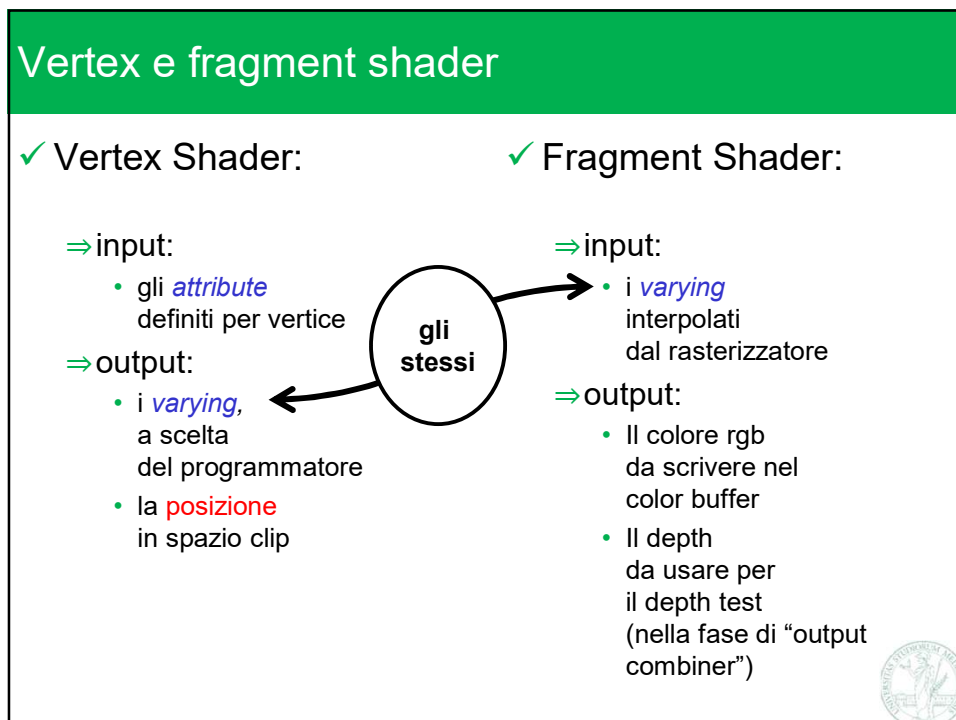
3



4



5



6

## CPU e shaders negli API

- ✓ Gli shaders sono piccoli programmi eseguiti in GPU nelle fasi per frammento e per vertice
- ✓ Possono essere programmati liberamente
- ✓ Il linguaggio dipendono dall'API
- ✓ In DirectX / OpenGL / Metal :
  - ⇒ linguaggi ad alto livello appositi (simile al C)
  - ⇒ HLSL, GLSL, Metal Shading Language rispettivamente
  - ⇒ L'API compila gli shader, e questi vengono caricati sulla GPU
- ✓ In Vulkan :
  - ⇒ Linguaggio bytecode a basso livello per shader
  - ⇒ simile all'assembler, o al bytecode Java



7

## Shaders in Three.js

- ✓ In three.js, gli shader sono inclusi nella classe «material»
- ✓ Nel progetto visto a lezione, vengono usati shader di default definiti nel materiale «simpleMeshMaterial»,
- ✓ Questi shader fanno solo il minimo sindacale:
  - ⇒ Il vertex shader produce le coordinate clip trasformando le coordinate oggetto con la matrice MVP (non vengono generati varying oltre al depth)
  - ⇒ Il fragment shader assegna un colore costante ad ogni frammento
- ✓ Nell'oggetto che usa le tessiture, gli shader vengono modificati:
  - ⇒ il vertex shader «passa» le coordinate texture come varying al rasterizzatore
  - ⇒ il fragment shader effettua un accesso alla tessitura (texture lookup) e restituisce il colore risultante



8