Graphics Coordinates and Pipeline

Mike Bailey

mjb@cs.oregonstate.edu

Oregon State University
Computer Graphics

The Basic Computer Graphics Pipeline

MC

Model Transform → WC → **View Transform** → EC → **Per-vertex Lighting** → EC → **Projection Transform** → CC → **Homogeneous Division**

→ NDC

Viewport Transform

→ SC

Rasterization → SC → **Fragment Processing, Texturing, Per-fragment Lighting** → SC → **Raster Ops** → SC → **Framebuffer**
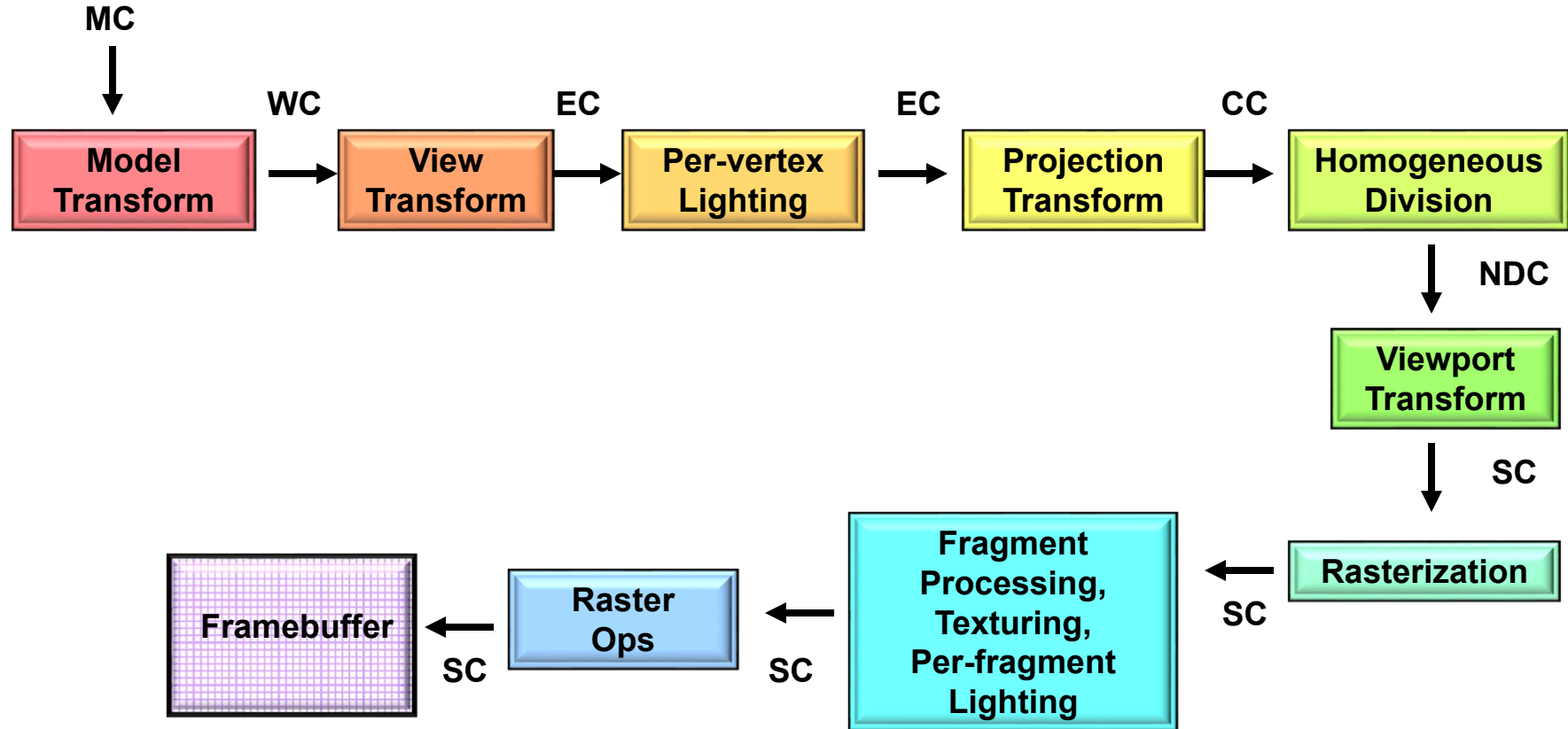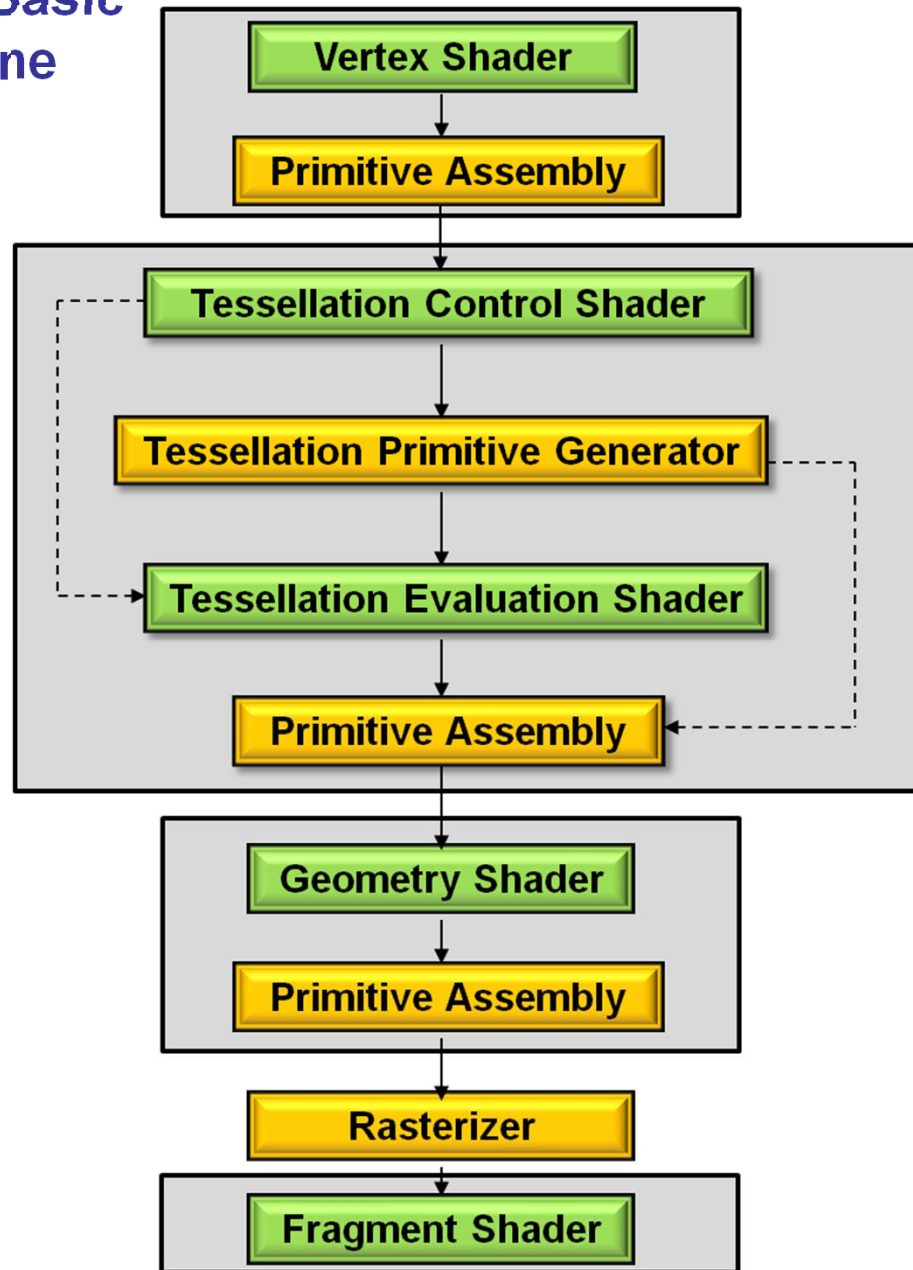
MC = Model Coordinates
WC = World Coordinates
EC = Eye Coordinates
CC = Clip Coordinates
NDC = Normalized Device Coordinates
SC = Screen Coordinates

Oregon State
University
Computer Graphics

The Shaders' View of the Basic Computer Graphics Pipeline

• In general, you want to have a vertex and fragment shader as a minimum.

• A missing stage is OK. The output from one stage becomes the input of the next stage that is there.

• The last stage before the fragment shader feeds its output variables into the **rasterizer**. The interpolated values then go to the fragment shaders

| | = Fixed Function |
| | = Programmable |

Vertex Shader

Primitive Assembly

Tessellation Control Shader

Tessellation Primitive Generator

Tessellation Evaluation Shader

Primitive Assembly

Geometry Shader

Primitive Assembly

Rasterizer

Fragment Shader

Oregon State
University
Computer Graphics