Vulkanised 2025

The 7th Vulkan Developer Conference
Cambridge, UK | February 11-13, 2025

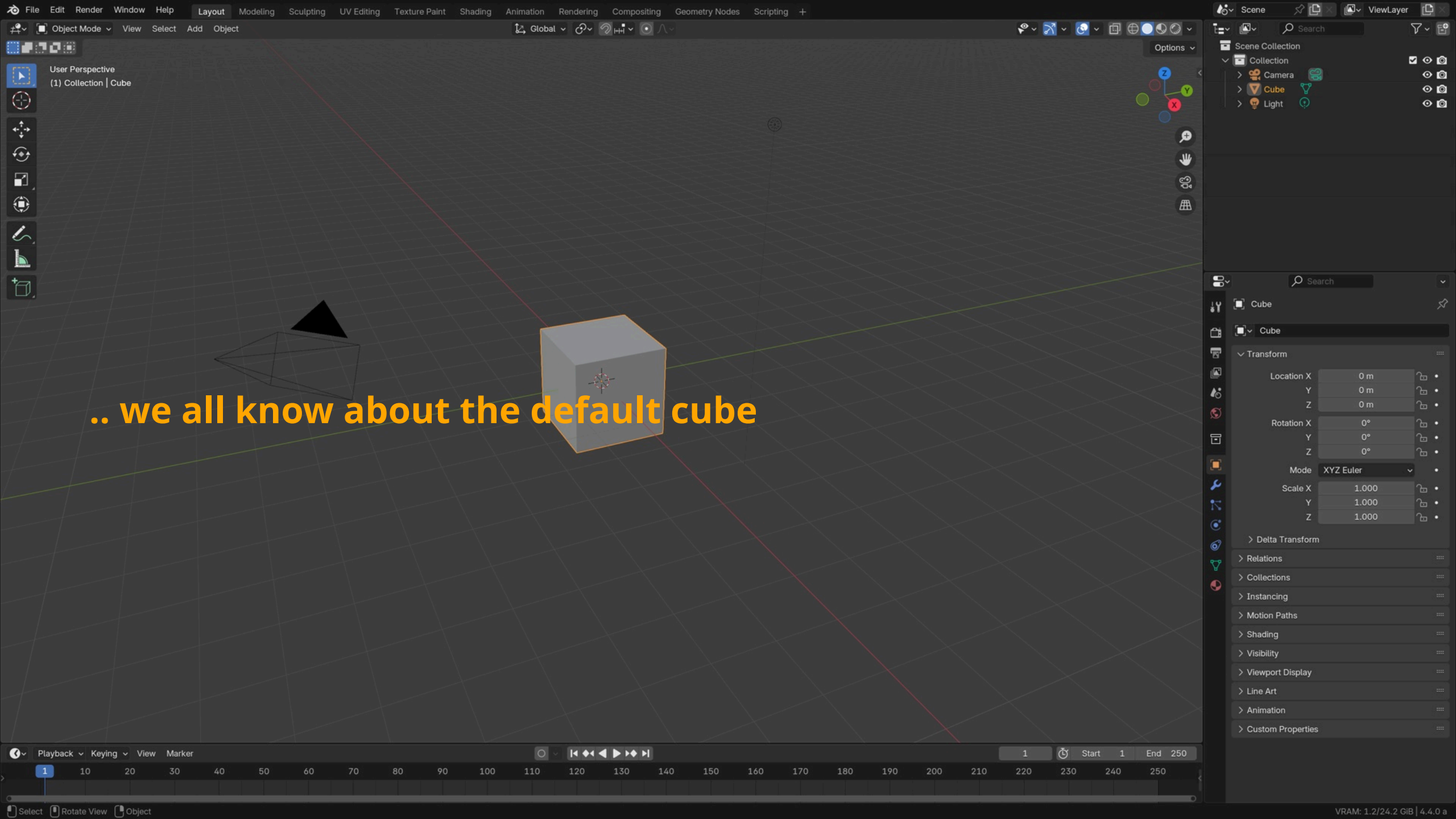
Blender Transition Towards Vulkan.

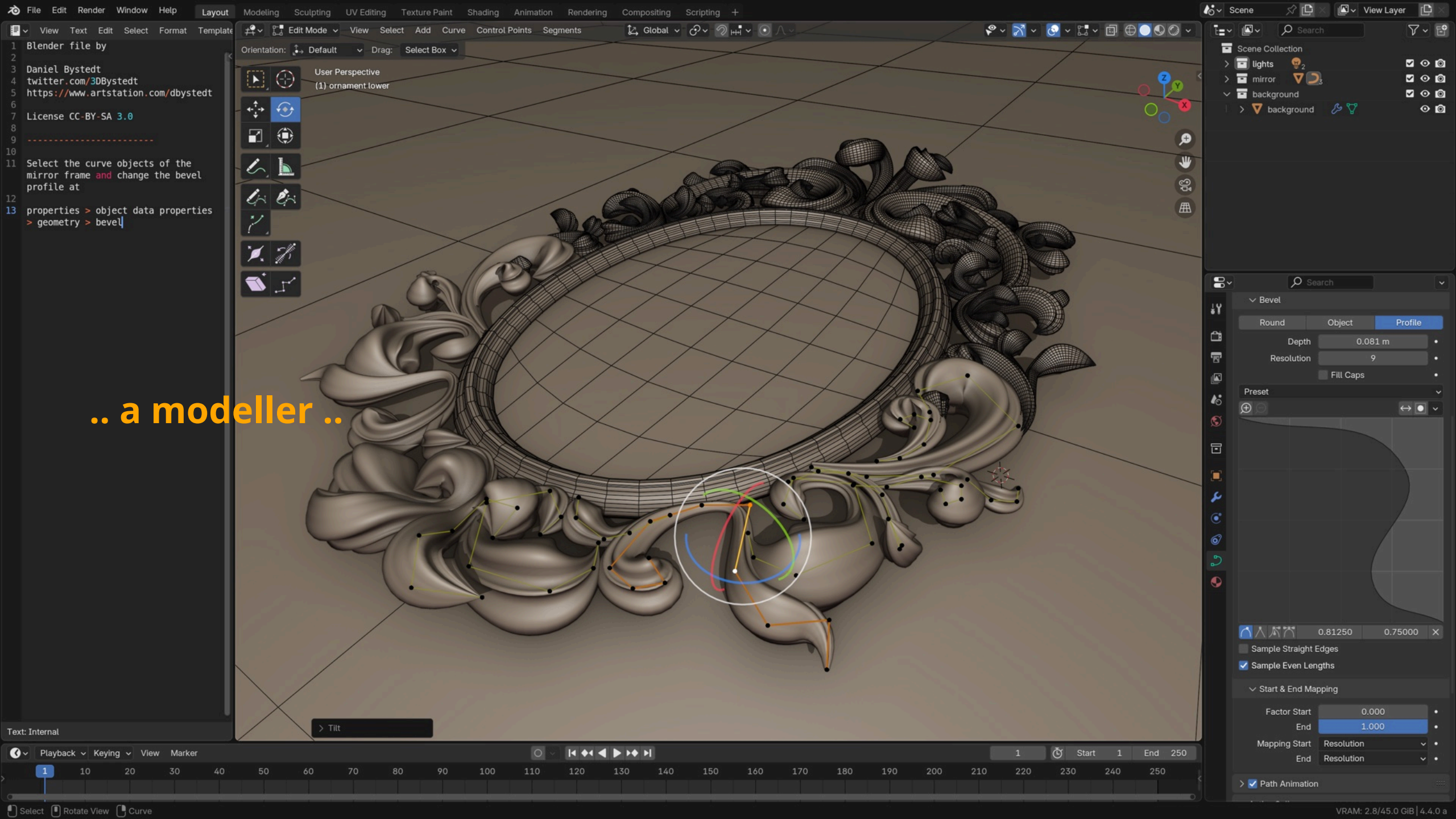
Jeroen Bakker, Blender Institute

Content

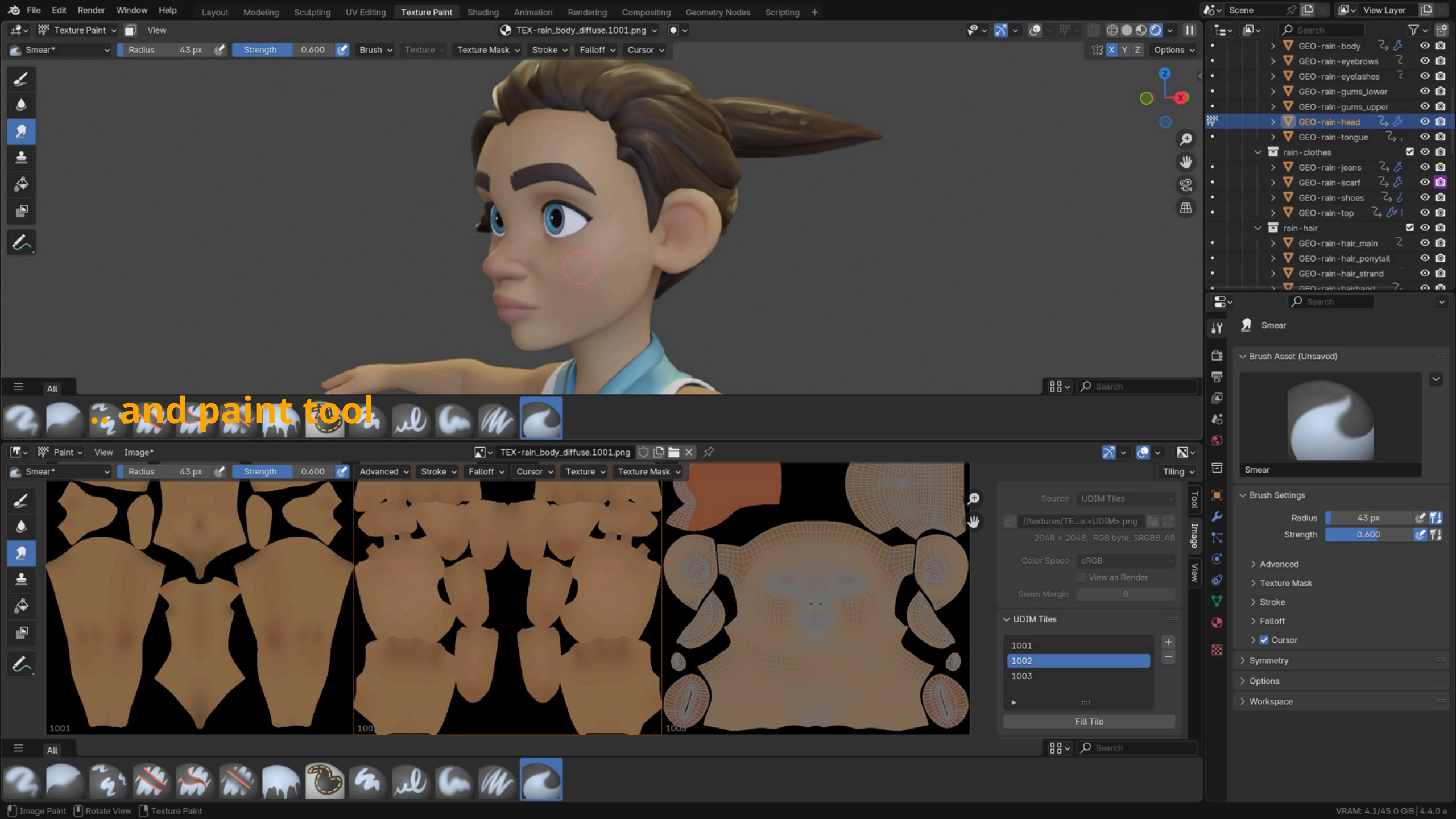
- What is Blender?
- Why Vulkan?
- Render graphs
- Shader compilation
- Platform support
- Tooling

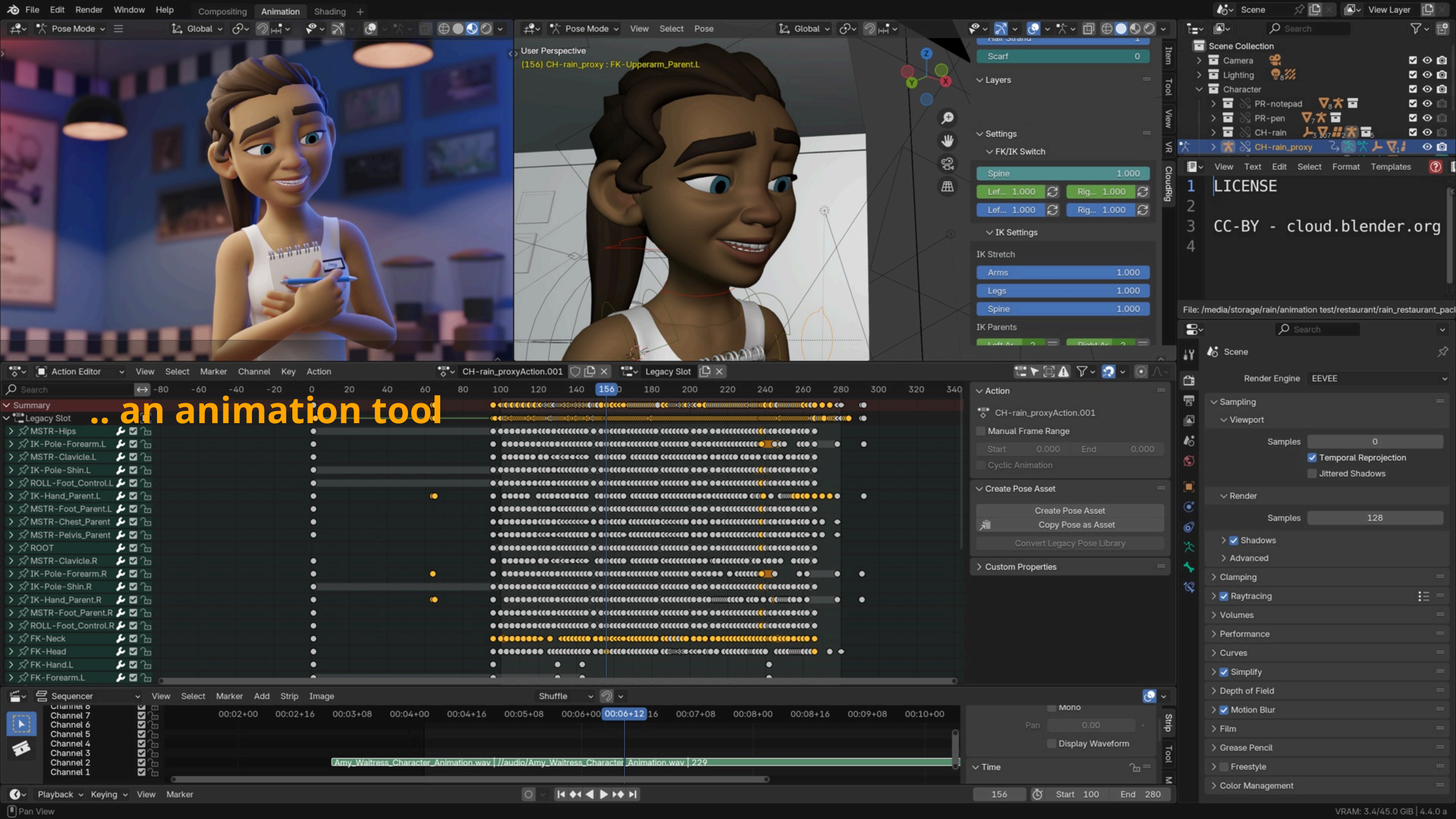
Blender is ..


File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Object Mode View Select Add Object
Global
Options
User Perspective
(1) Collection | Cube
.. we all know about the default cube
Scene
ViewLayer
Search
Scene Collection
Collection
Camera
Cube
Light
Cube
Transform
Location X 0 m
Y 0 m
Z 0 m
Rotation X 0°
Y 0°
Z 0°
Mode XYZ Euler
Scale X 1.000
Y 1.000
Z 1.000
Delta Transform
Relations
Collections
Instancing
Motion Paths
Shading
Visibility
Viewport Display
Line Art
Animation
Custom Properties
Playback Keying View Marker
1
Start 1
End 250
Select
Rotate View
Object
VRAM: 1.2/24.2 GiB | 4.4.0 a


Blender file by
Daniel Bystedt
twitter.com/3DBystedt
https://www.artstation.com/dbystedt
License CC-BY-SA 3.0
Select the curve objects of the mirror frame and change the bevel profile at
properties > object data properties > geometry > bevel
.. a modeller ..
User Perspective
(1) ornament lower
Bevel
Round
Object
Profile
Depth
0.081 m
Resolution
9
Fill Caps
Preset
0.81250
0.75000
Sample Straight Edges
Sample Even Lengths
Start & End Mapping
Factor Start
0.000
End
1.000
Mapping Start
Resolution
End
Resolution
Path Animation
Scene Collection
lights
mirror
background
background
Tilt
Text: Internal


File Edit Render Window Help
Layout
Scene
View Layer
Sculpt Mode View Sculpt Mask Face Sets
Draw Radius 105 px Strength 0.500 Brush Texture Stroke Falloff Cursor
Dyntopo Remesh Options
User Perspective
(1) Thief
Active Tool
Brush
Brush Asset
View Text Edit Select Format Templates README
New Grab Silhouette option in Blender 2.92
- The grab silhouette option allows grabbing the silhouette of a mesh from one side without affecting the shape of the other.
- It's useful for moving one side of thin and close meshes (like limbs or fingers) without affecting the other part, that otherwise should be masked.
.. sculpt ..
All Libraries
All
Brushes
Mesh Sculpt
General
Paint
Simulation
Blob Clay Clay Strips Clay Thumb Crease Poli... Crease Sha... Draw Draw Sharp Inflate/Def... Layer Fill/Deepen Flatten/Co... Plateau
Scrape Mul... Scrape/Fill Smooth Trim Boundary Elastic Grab Elastic Sna... Grab Grab 2D Grab Silho... Nudge Pinch/Mag... Pose
Pull Relax Pinch Relax Slide Snake Hook Thumb Twist Density Erase Multi... Face Set P... Mask Smear Mult... Airbrush Blend Hard
Blend Soft Blend Squa... Paint Blend Paint Hard Paint Hard ... Paint Soft Paint Soft ... Paint Square Sharpen Smear Bend Boun... Bend/Twist... Drag Cloth
Expand/Co... Grab Cloth Grab Plana... Grab Rand... Inflate Cloth Pinch Fold... Pinch Point... Push Cloth Stretch/Mo... Twist Boun...
All Cloth General Paint
Search
VRAM: 3.3/45.0 GiB | 4.4.0 a

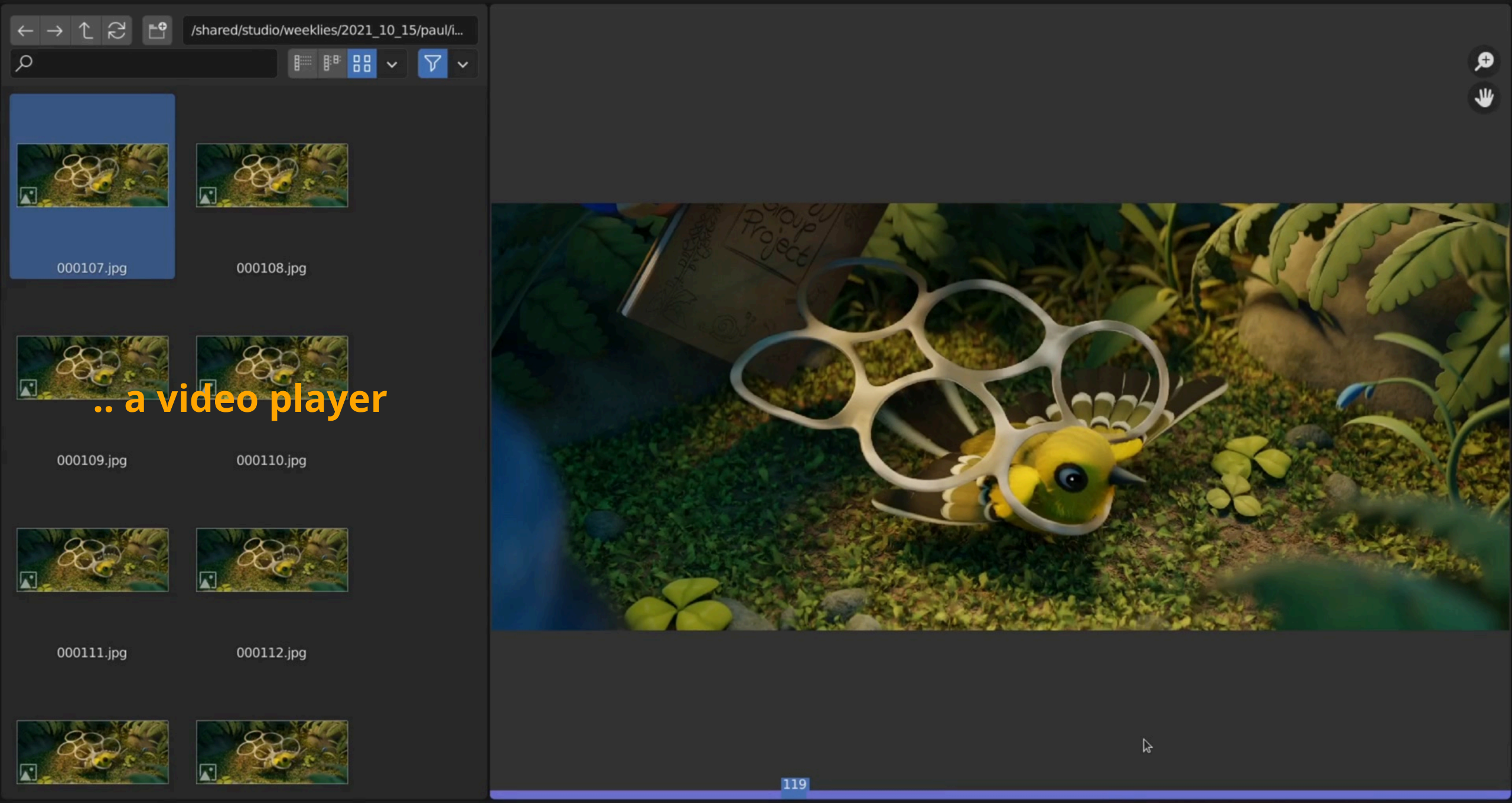

File Edit Render Window Help
Layout Modeling Sculpting UV Editing Texture Paint Shading Animation Rendering Compositing Geometry Nodes Scripting
Texture Paint View
TEX-rain_body_diffuse.1001.png
Smear* Radius 43 px Strength 0.600 Brush Texture Texture Mask Stroke Falloff Cursor
Options
Scene View Layer
GEO-rain-body
GEO-rain-eyebrows
GEO-rain-eyelashes
GEO-rain-gums_lower
GEO-rain-gums_upper
GEO-rain-head
GEO-rain-tongue
rain-clothes
GEO-rain-jeans
GEO-rain-scarf
GEO-rain-shoes
GEO-rain-top
rain-hair
GEO-rain-hair_main
GEO-rain-hair_ponytail
GEO-rain-hair_strand
.. and paint tool
Paint View Image*
TEX-rain_body_diffuse.1001.png
Smear* Radius 43 px Strength 0.600 Advanced Stroke Falloff Cursor Texture Texture Mask
Tiling
Source UDIM Tiles
//textures/TE...e.<UDIM>.png
2048 × 2048, RGB byte, SRGB8_A8
Color Space sRGB
View as Render
Seam Margin 8
UDIM Tiles
1001
1002
1003
Fill Tile
Smear
Brush Asset (Unsaved)
Smear
Brush Settings
Radius 43 px
Strength 0.600
Advanced
Texture Mask
Stroke
Falloff
Cursor
Symmetry
Options
Workspace
Image Paint
Rotate View
Texture Paint
VRAM: 4.1/45.0 GiB | 4.4.0 a


.. an animation tool

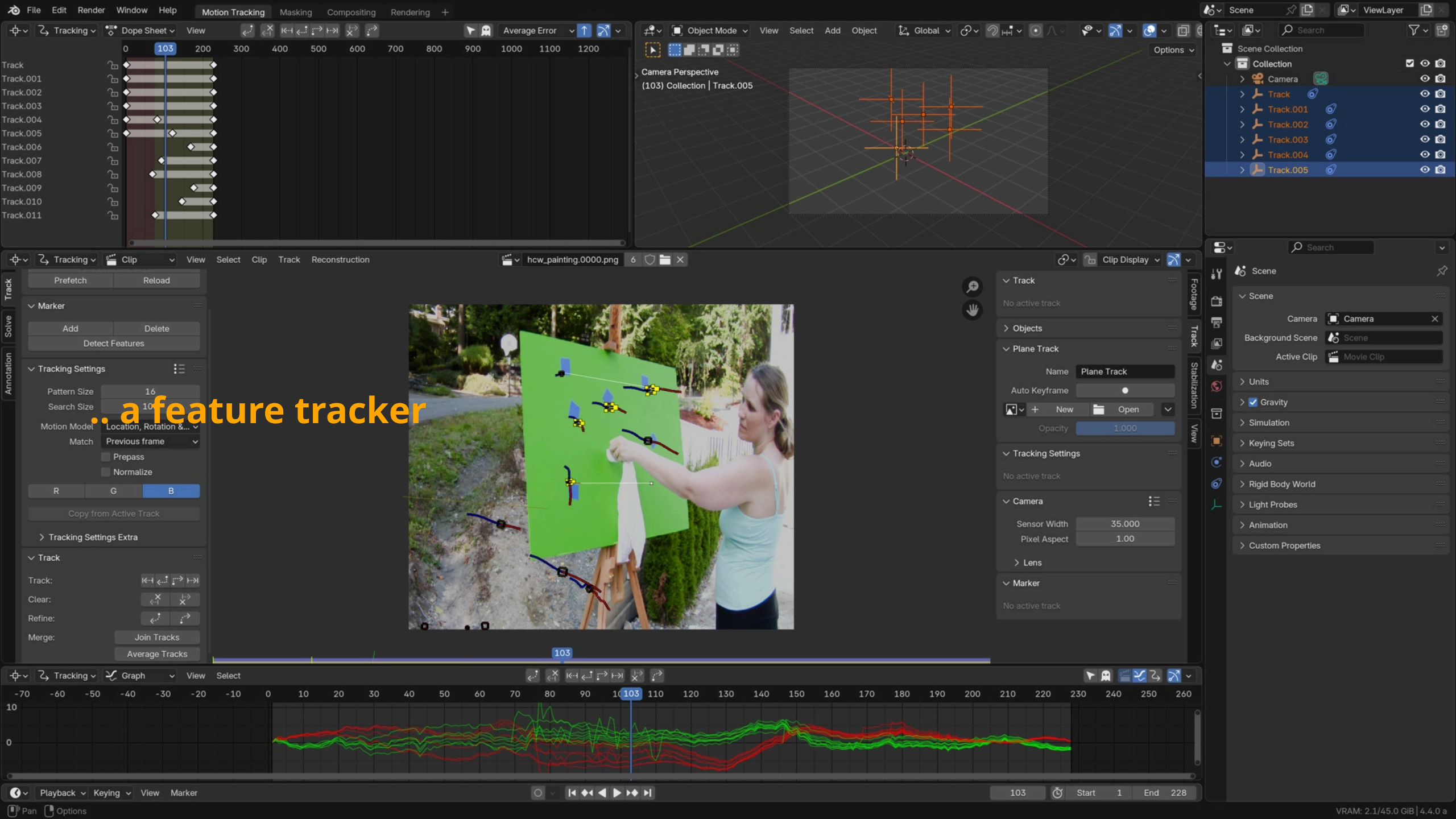

File Edit Render Window Help
3D View Full Animation Compositing Cycles Default Game Logic Motion Tracking Scripting UV Editing Video Editing
Camera Perspective
(1) sponza_322
.. a rasterizer ..
Scene
RenderLayer
Search
Scene Collection
Collection 1
Collection 2
Collection 11
Collection 12
Volume
Render Engine EEVEE
Sampling
Viewport
Samples 16
Temporal Reprojection
Jittered Shadows
Render
Samples 64
Shadows
Advanced
Clamping
Raytracing
Volumes
Curves
Simplify
Depth of Field
Motion Blur
Film
Performance
Grease Pencil
Freestyle
Color Management
Display Device sRGB
View Transform Standard
Look None
Exposure 2.000
Gamma 1.000
Edit Mode View Select Add Mesh Vertex Edge Face UV
Global
Pan Options
VRAM: 4.1/45.0 GiB | 4.4.0 a

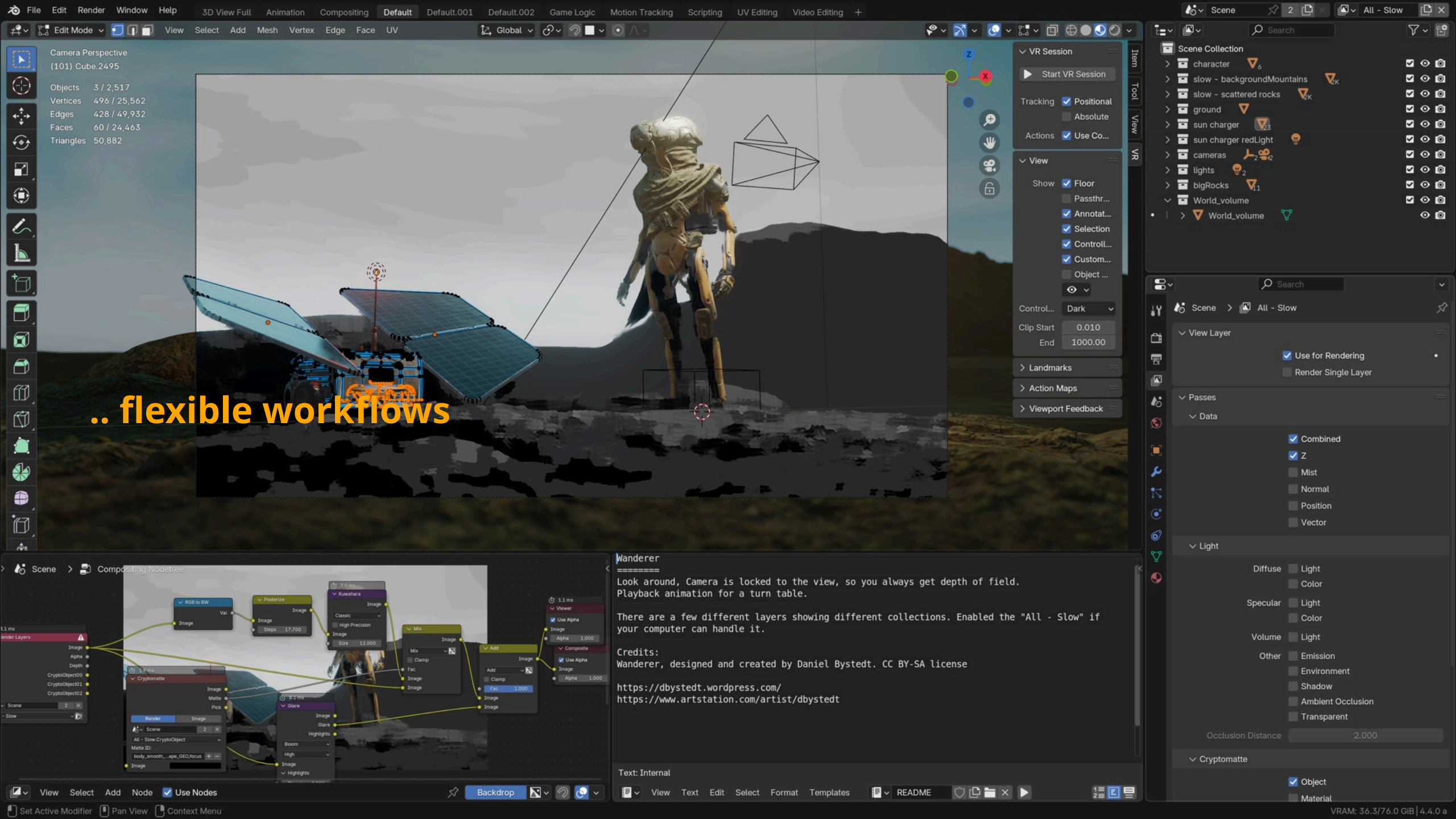

File Edit Render Window Help
3D View Full Animation Compositing Cycles Default Game Logic Motion Tracking Scripting UV Editing Video Editing
Camera Perspective
(1) sponza_322
Sample 22/64
.. and a path tracer
Scene
RenderLayer
Search
Scene Collection
Collection 1
Collection 2
Collection 11
Collection 12
Volume
Render Engine Cycles
Feature Set Supported
Device CPU
Open Shading Language
Sampling
Viewport
Noise Threshold 0.1000
Max Samples 64
Min Samples 0
Denoise
Render
Noise Threshold 0.0100
Max Samples 1000
Min Samples 0
Time Limit 0 s
Denoise
Path Guiding
Lights
Advanced
Light Paths
Volumes
Curves
Simplify
Motion Blur
Film
Performance
Bake
Grease Pencil
Edit Mode View Select Add Mesh Vertex Edge Face UV
Global
Pan Options
VRAM: 2.3/45.0 GiB | 4.4.0 a

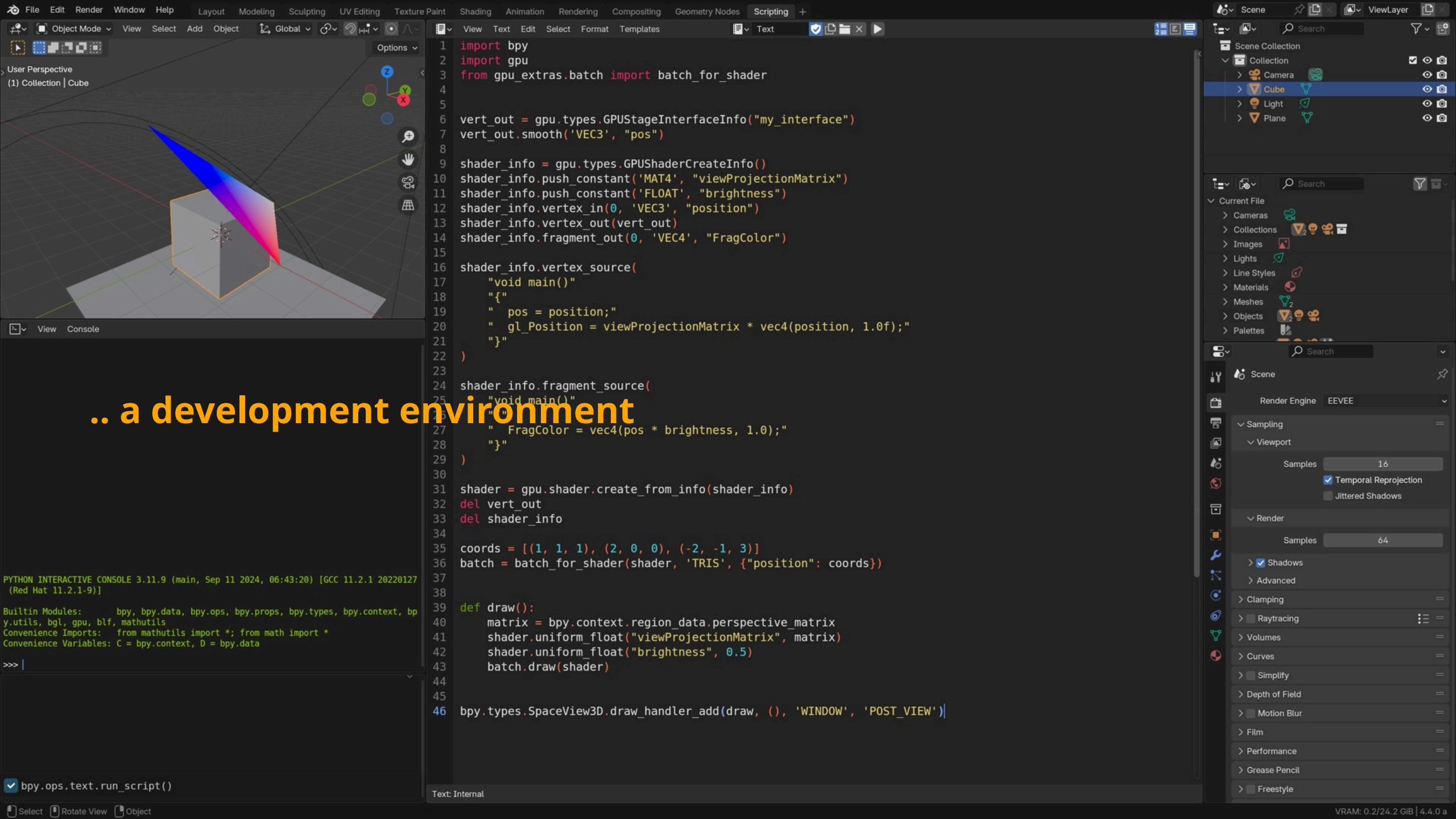

.. a 2D animation tool


/shared/studio/weeklies/2021_10_15/paul/i...
000107.jpg
000108.jpg
000109.jpg
000110.jpg
000111.jpg
000112.jpg
.. a video player
119


.. a video editor


.. a feature tracker


.. flexible workflows
Wanderer
Look around, Camera is locked to the view, so you always get depth of field.
Playback animation for a turn table.
There are a few different layers showing different collections. Enabled the "All - Slow" if your computer can handle it.
Credits:
Wanderer, designed and created by Daniel Bystedt. CC BY-SA license
https://dbystedt.wordpress.com/
https://www.artstation.com/artist/dbystedt



.. a development environment

Search projects

Help Sponsors Log in Register

bpy 4.3.0

```
pip install bpy==4.3.0
```

Latest version

Released: Nov 21, 2024

Blender as a Python module

.. a python module

Navigation

- Project description
- Release history
- Download files

Verified details

These details have been verified by PyPI

Maintainers

BlenderFoundation

Unverified details

Project description

Blender

Blender is the free and open source 3D creation suite. It supports the entirety of the 3D pipeline���modeling, rigging, animation, simulation, rendering, compositing and motion tracking, even video editing.

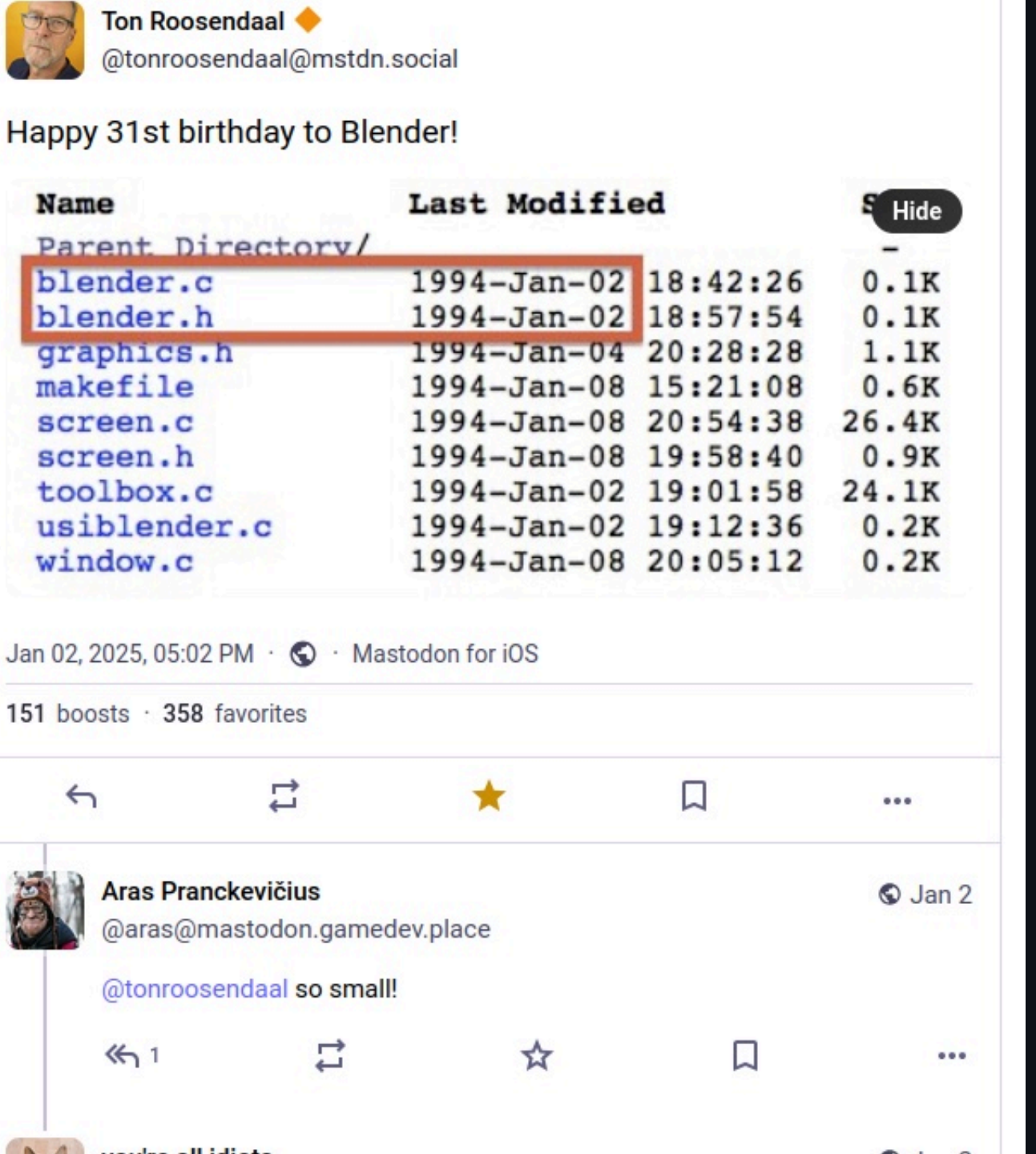
This package provides Blender as a Python module for use in studio pipelines, web services, scientific research, and more.

Documentation

- Blender Python API
- Blender as a Python Module

Requirements

.. aged well


Ton Roosendaal
@tonroosendaal@mstdn.social
Happy 31st birthday to Blender!
Name Last Modified
Hide
Parent Directory/ -
blender.c 1994-Jan-02 18:42:26 0.1K
blender.h 1994-Jan-02 18:57:54 0.1K
graphics.h 1994-Jan-04 20:28:28 1.1K
makefile 1994-Jan-08 15:21:08 0.6K
screen.c 1994-Jan-08 20:54:38 26.4K
screen.h 1994-Jan-08 19:58:40 0.9K
toolbox.c 1994-Jan-02 19:01:58 24.1K
usiblender.c 1994-Jan-02 19:12:36 0.2K
window.c 1994-Jan-08 20:05:12 0.2K
Jan 02, 2025, 05:02 PM · Mastodon for iOS
151 boosts · 358 favorites
Aras Pranckevičius
@aras@mastodon.gamedev.place
Jan 2
@tonroosendaal so small!
1

BLACKDUCK® | Open Hub

Follow @ OH | Sign In | Join Now

Projects | People | Organizations | Tools | Blog | BDSA | Projects ▾ | Search...

Blender 3D

Settings | Report Duplicate

Very High Activity

610

I Use This!

Analyzed about 14 hours ago. based on code collected 1 day ago.

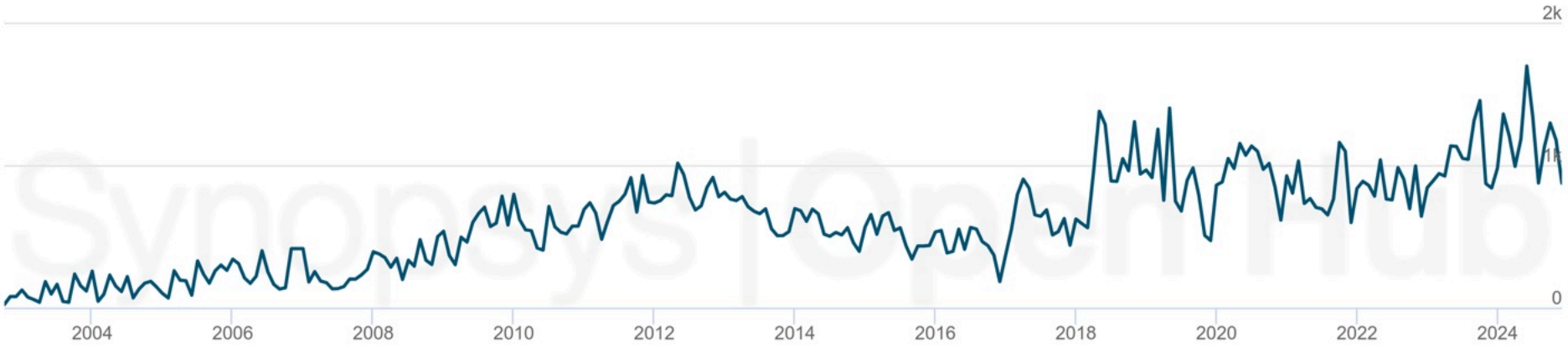
Commits

	All Time	12 Month	30 Day
Commits:	161842	14080	867
Contributors:	1191	244	71
Files Modified:	38727	12888	3606
Lines Added:	43027501	1385837	75027
Lines Removed	31468764	1174710	58738

.. and is still growing

Commits per Month

Zoom 1yr 3yr 5yr 10yr **All**

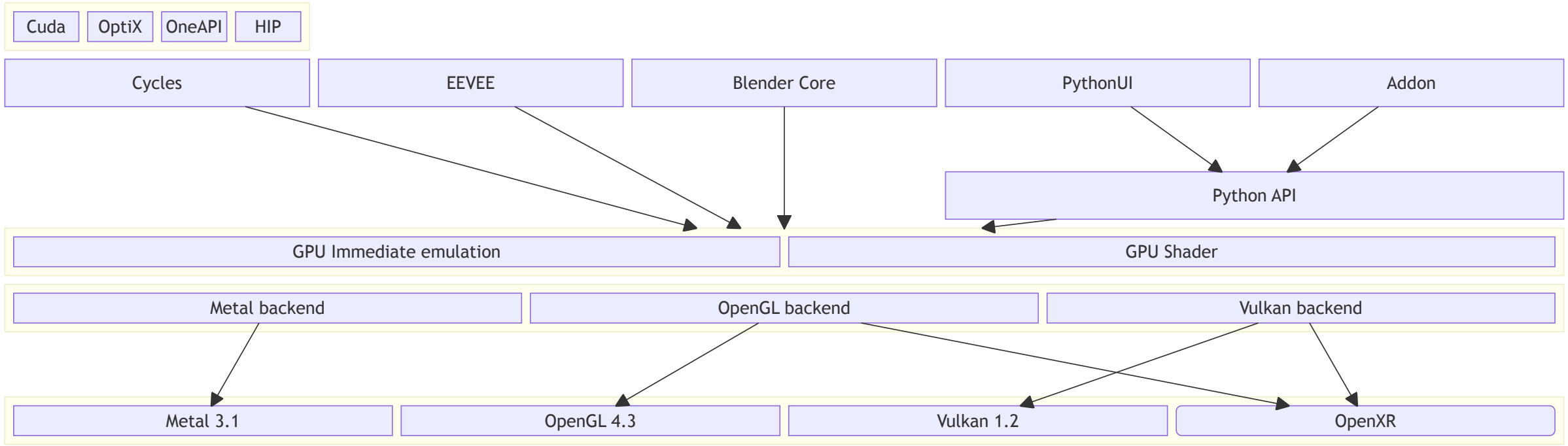


Bender is ..

- Community driven open source 3D content creation suite.

Why Vulkan?

- OpenGL is loosing traction
  - Blender pushes boundaries
  - Increase potential bugs
  - New features not available
  - Stability & support
- Perhaps leverage better GPU utilization.


Cuda
OptiX
OneAPI
HIP
Cycles
EEVEE
Blender Core
PythonUI
Addon
Python API
GPU Immediate emulation
GPU Shader
Metal backend
OpenGL backend
Vulkan backend
Metal 3.1
OpenGL 4.3
Vulkan 1.2
OpenXR

Vulkan backend - high level

- Data management
  - Resource tracking
  - Conversion
- Render graph
- Command building
- Shader compilation

Getting something drawn

- Approach is a render graph
- Nodes with resource dependencies
- Resource tracking per image layer
- Existing API pushed to a node per command.

Render graph nodes

Node	Vulkan API
Data transfer	
`VKBlitImageNode`	`vkCmdBlitImage`
`VKClearColorImageNode`	`vkCmdClearColorImage`
`VKClearDepthStencilImageNode`	`vkCmdClearDepthStencilImageNode`
`VKCopyBufferNode`	`vkCmdCopyBuffer`
`VKCopyBufferToImageNode`	`vkCmdCopyBufferToImage`
`VKCopyImageNode`	`vkCmdCopyImage`
`VKCopyImageToBufferNode`	`vkCmdCopyImageToBufferNode`
`VKFillBufferNode`	`vkCmdFillBuffer`
`VKSynchonizationNode`	`vkCmdPipelineBarrier` (only used for swapchain barriers)
`VKUpdateBufferNode`	`vkCmdUpdateBuffer`
`VKUpdateMipmapsNode`	`vkCmdPipelineBarrier` / `vkCmdBlitImage`

Render graph nodes

Node	Vulkan API
Draw	
`VKBeginRenderingNode`	`vkCmdBeginRendering` / `vkCmdBeginRenderPass`
`VKClearAttachmentsNode`	`vkCmdClearAttachments`
`VKDrawIndexedIndirectNode`	`vkCmdDrawIndexedIndirect`
	optional: [`vkCmdBindPipeline` / `vkCmdBindDescriptorSets` / `vkCmdPushConstants`
	`vkCmdBindVertexBuffers` / `vkCmdBindIndexBuffer`]
`VKDrawIndexedNode`	`vkCmdDrawIndexed`
	optional: [`vkCmdBindPipeline` / `vkCmdBindDescriptorSets` / `vkCmdPushConstants`
	`vkCmdBindVertexBuffers` / `vkCmdBindIndexBuffer`]
`VKDrawIndirectNode`	`vkCmdDrawIndirect`
	optional: [`vkCmdBindPipeline` / `vkCmdBindDescriptorSets` / `vkCmdPushConstants`
	`vkCmdBindVertexBuffers`]
`VKDrawNode`	`vkCmdDraw`
	optional: [`vkCmdBindPipeline` / `vkCmdBindDescriptorSets` / `vkCmdPushConstants`
	`vkCmdBindVertexBuffers`]
`VKEndRenderingNode`	`vkCmdEndRendering` / `vkCmdEndRenderPass`

Render graph nodes

Node	Vulkan API
Compute	
`VKDispatchIndirectNode`	`vkCmdDispatchIndirect`
	optional: [`vkCmdBindPipeline` / `vkCmdBindDescriptorSets` / `vkCmdPushConstants`]
`VKDispatchNode`	`vkCmdDispatch`
	optional: [`vkCmdBindPipeline` / `vkCmdBindDescriptorSets` / `vkCmdPushConstants`]
GPU selection	
`VKBeginQueryNode`	`vkCmdBeginQuery`
`VKEndQueryNode`	`vkCmdEndQuery`
`VKResetQueryPoolNode`	`vkCmdResetQueryPool`

Node

```
template<VKNodeType NodeType,
         typename NodeCreateInfo,
         typename NodeData,
         VkPipelineStageFlags PipelineStage,
         VKResourceType ResourceUsages>
class VKNodeInfo : public NonCopyable {

 public:
  using CreateInfo = NodeCreateInfo;
  using Data = NodeData;

  /**
   * Node type of this class.
   *
   * The node type used to link VKRenderGraphNode instance to a VKNodeInfo.
   */
  static constexpr VKNodeType node_type = NodeType;

  /**
   * Which pipeline stage does this command belongs to. The pipeline stage is used when generating
   * pipeline barriers.
   */
  static constexpr VkPipelineStageFlags pipeline_stage = PipelineStage;

  ...
```

Node

```
  ...
  /**
   * Which resource types are relevant. Some code can be skipped when a node can only depend on
   * resources of a single type.
   */
  static constexpr VKResourceType resource_usages = ResourceUsages;

  /**
   * Update the node data with the data inside create_info.
   *
   * Has been implemented as a template to ensure all node specific data
   * (`Data`/`CreateInfo`) types can be included in the same header file as the logic. The
   * actual node data (`VKRenderGraphNode` includes all header files.)
   *
   * This function must be implemented by all node classes. But due to cyclic inclusion of header
   * files it is implemented as a template function.
   */
  template<typename Node, typename Storage>
  static void set_node_data(Node &node, Storage &storage, const CreateInfo &create_info);

  /**
   * Extract read/write resource dependencies from `create_info` and add them to `node_links`.
   */
  virtual void build_links(VKResourceStateTracker &resources,
                           VKRenderGraphNodeLinks &node_links,
                           const CreateInfo &create_info) = 0;

  /**
   * Build the commands and add them to the command_buffer.
   *
   * The command buffer is passed as an interface as this is replaced by a logger when running test
   * cases. The test cases will validate the log to find out if the correct commands where added.
   */
  virtual void build_commands(VKCommandBufferInterface &command_buffer,
                              Data &data,
                              VKBoundPipelines &r_bound_pipelines) = 0;
};
```

Node implementation

```
struct VKCopyBufferToImageData {
  VkBuffer src_buffer;
  VkImage dst_image;
  VkBufferImageCopy region;
};
struct VKCopyBufferToImageCreateInfo {
  VKCopyBufferToImageData node_data;
  VkImageAspectFlags vk_image_aspects;
};

class VKCopyBufferToImageNode : public VKNodeInfo<VKNodeType::COPY_BUFFER_TO_IMAGE,
                                                   VKCopyBufferToImageCreateInfo,
                                                   VKCopyBufferToImageData,
                                                   VK_PIPELINE_STAGE_TRANSFER_BIT,
                                                   VKResourceType::IMAGE | VKResourceType::BUFFER> {
 public:
  template<typename Node, typename Storage> static void set_node_data(Node &node, Storage &storage, const CreateInfo &create_info)
  {
    node.storage_index = storage.copy_buffer_to_image.append_and_get_index(create_info.node_data);
  }

  void build_links(VKResourceStateTracker &resources, VKRenderGraphNodeLinks &node_links, const CreateInfo &create_info) override
  {
    ResourceWithStamp src_resource = resources.get_buffer(create_info.node_data.src_buffer);
    ResourceWithStamp dst_resource = resources.get_image_and_increase_stamp(create_info.node_data.dst_image);
    node_links.inputs.append({src_resource, VK_ACCESS_TRANSFER_READ_BIT});
    node_links.outputs.append({
      dst_resource, VK_ACCESS_TRANSFER_WRITE_BIT, VK_IMAGE_LAYOUT_TRANSFER_DST_OPTIMAL, create_info.vk_image_aspects});
  }

  void build_commands(VKCommandBufferInterface &command_buffer, Data &data, VKBoundPipelines & /*r_bound_pipelines*/) override
  {
    command_buffer.copy_buffer_to_image(
        data.src_buffer, data.dst_image, VK_IMAGE_LAYOUT_TRANSFER_DST_OPTIMAL, 1, &data.region);
  }
};
```

Node resource dependencies

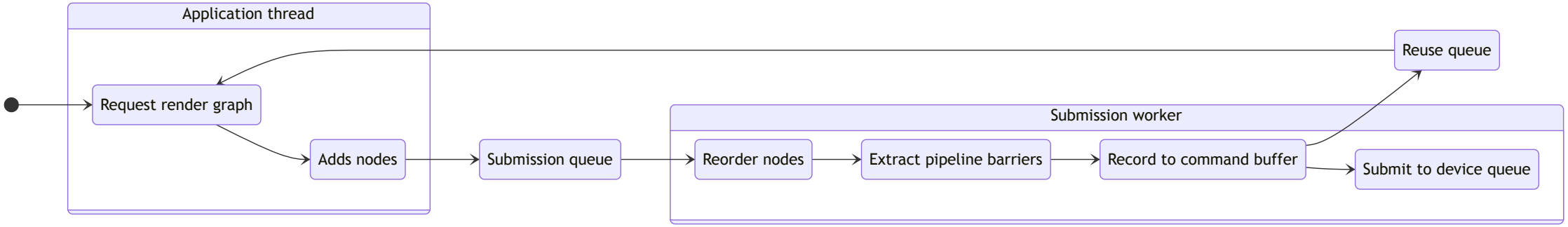
```
groups_build_commands:
  node_group=969,
  node_group_range=1019-1019,
  node_handle=1019,
  node_type="DISPATCH_INDIRECT",
  debug_group="/SPACE_VIEW3D/Viewport/EEVEE/negZ_view/Deferred.Opaque/Shadow/TilemapUpdate/RenderClear"

NODE:
  type="DISPATCH_INDIRECT"

 inputs:
  handle=377, type="BUFFER", vk_handle=140734952600192, vk_access="VK_ACCESS_SHADER_READ_BIT"
  handle=355, type="BUFFER", vk_handle=140734934300544, vk_access="VK_ACCESS_SHADER_READ_BIT"
  handle=351, type="BUFFER", vk_handle=140734934300032, vk_access="VK_ACCESS_INDIRECT_COMMAND_READ_BIT"

 outputs:
  handle=865, type="IMAGE", vk_handle=140734829073920, name="shadow_atlas_tx_",
    vk_access=["VK_ACCESS_SHADER_READ_BIT", "VK_ACCESS_SHADER_WRITE_BIT"],
    vk_image_layout="VK_IMAGE_LAYOUT_GENERAL",
    vk_image_aspect="VK_IMAGE_ASPECT_COLOR_BIT",
    layer_base=0
```

Render graph submission



- Many application threads
  - UI: 1-2 threads
  - EEVEE: 2-4 threads
  - Cycles: `n` threads
- Input order is UI code, not GPU order
- Command buffers are not kept: workload assumed to be always rendering
- Queue submission can be post-poned

Shader compilation

- Compilation performance is essential
  - Latency between user action and the result
  - Partial update before caching before baking
  - Materials fast compilation vs fast execution
  - Material/engine recompilation
- ShaderC allows to compile shaders in parallel, however...

KhronosGroup / SPIRV-Tools

Type / to search

Code | Issues 395 | Pull requests 12 | Actions | Projects 1 | Wiki | Security | Insights

FriendlyNameMapper::to_string(uint32_t) blocking on linux #5802

Edit | New issue

Closed | #5805

jeroenbakker-atmind opened on Sep 12, 2024

Hi!

Blender developer working on integrating Vulkan backend into Blender. Shader compilation performance is really important in our usecase as precompilation is not really possible. However when running more optimizers in parallel the performance degrades very fast due to stdlib implementation.

See the next hotspot where we compile hundreds of shaders using all threads available on the system. It takes almost as long as running on a single thread.

Via hotspot the issue tracks down as stdlib that uses global lock when accessing the locale. This could be evaded by changing the `to_string(uint32_t)` implementation at https://github.com/KhronosGroup/SPIRV-Tools/blob/main/source/name_mapper.cpp#L34 This part of the code doesn't require any locale but as noted in the comments is done this way for certain android compilers/platforms.

Would it be possible to reevaluate the current implementation so it would not stall when using multithreaded compilation on Linux?

I am running on a regular Ubuntu 24.04 `Linux-6.8.0-41-generic-x86_64-with-glibc2.39 64 Bits, X11 UI`.

Assignees: dneto0

Labels: component:optimization

Type: No type

Projects: No projects

Milestone: No milestone

Relationships: None yet

Development:
- Using std::to_string — KhronosGroup/SPIRV-Tools
- Implement to_string(uint32_t) without using the locale — KhronosGroup/SPIRV-Tools

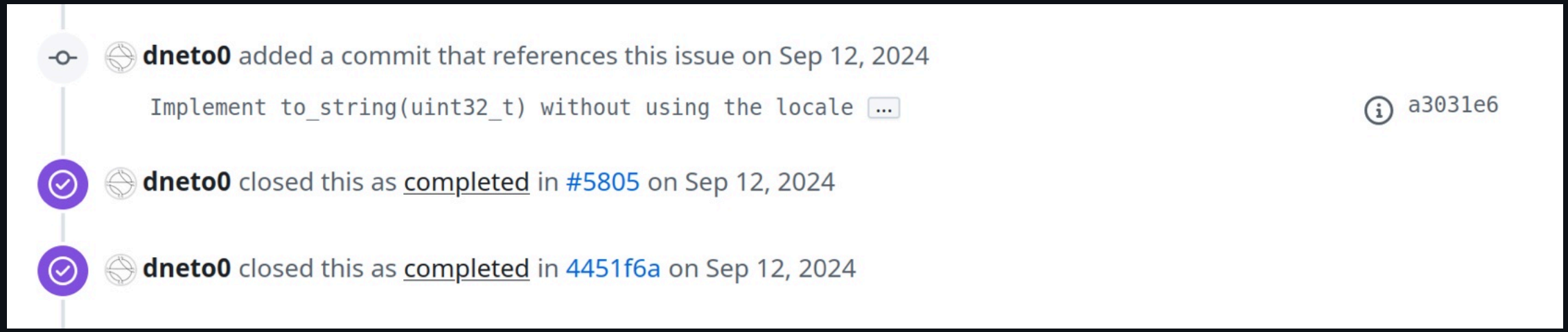
Notifications: Customize

Unsubscribe

You're receiving notifications because you're subscribed to this thread.

Participants

Power of open source



- Must do: Have an issue -> report!
  - Projects want to understand use-cases
  - We like bugs (to be known, so they can be solved)

Benchmark

Test: Start Blender, open scene, final viewport
3000 objects, 550 materials, 200 images

Backend	Test	Duration (s)
OpenGL	**Coldstart/No subprocesses**	1:52
OpenGL	Coldstart/8 subprocesses	0:54
OpenGL	Warmstart/8 subprocesses	0:06
Vulkan	Coldstart single threaded	0:59
Vulkan	Warmstart single threaded	0:10
Vulkan	**Coldstart multi threaded**	0:06
Vulkan	Warmstart multi threaded	0:03

Blender - platforms support

- Development targets Vulkan 1.2.
- In the end all supported platforms can do 1.3.

Extensions and features

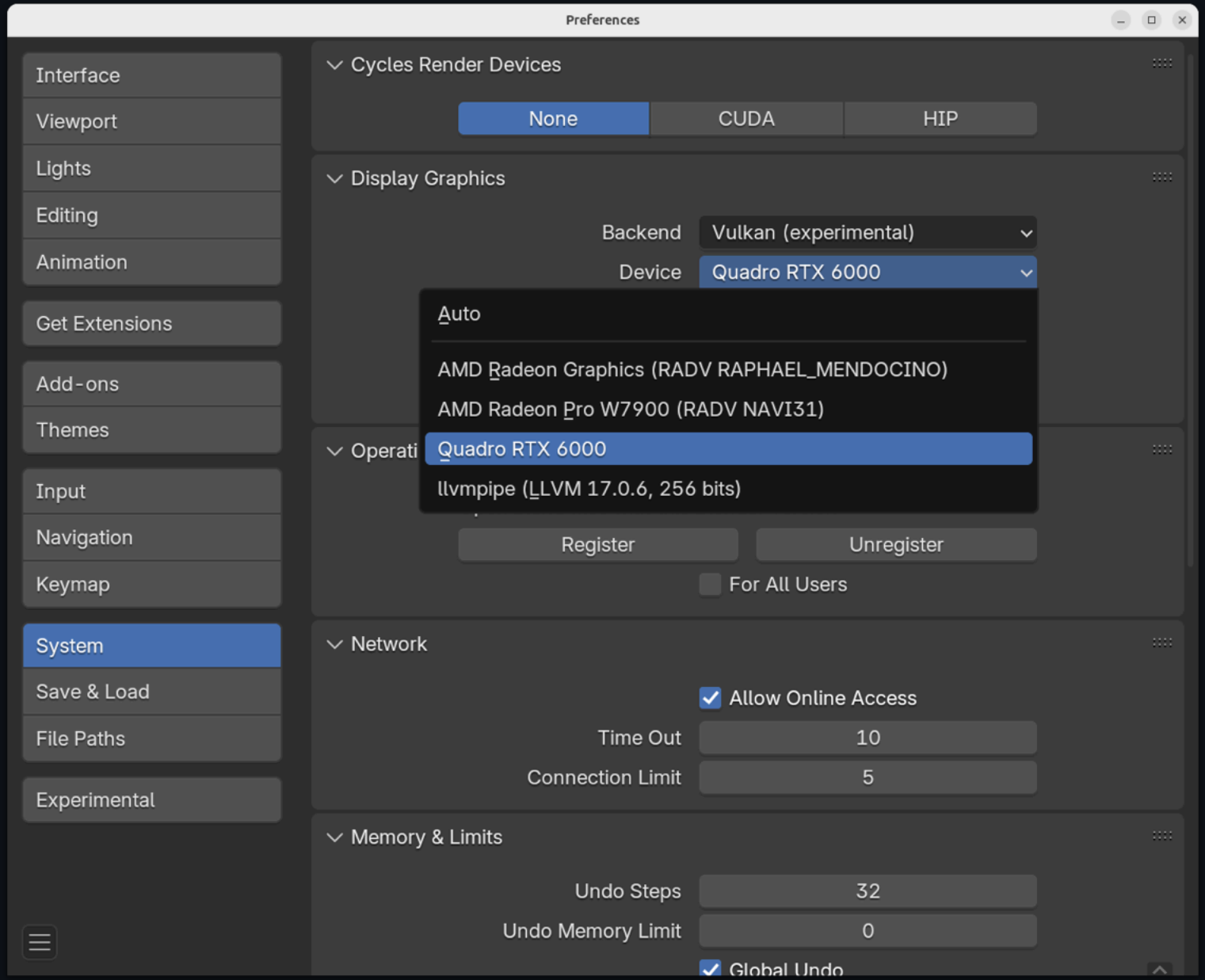
Features	Extensions
Required	**Required**
Geometry shaders [1]	VK_EXT_provoking_vertex_extensions
Logical operations	
Dual source blending	**Optional**
Image cube array	VK_KHR_dynamic_rendering
Multi draw indirect	VK_EXT_dynamic_rendering_unused_attachments
Multi viewport	VK_EXT_dynamic_rendering_local_read
Shader clip distance	VK_EXT_shader_stencil_export
Draw indirect first instance	VK_EXT_swapchain_colorspace
Fragment store and atomics	VK_KHR_fragment_shader_barycentric
	VK_KHR_maintenance4
Optional	
Sampler anisotropy	
Shader draw parameters	
Shader output layer	
Shader output viewport index	

Supported platforms

Devices	Windows	Linux
NVIDIA		
GTX 700 - GTX 800	NVIDIA 500+	*Unsupported*
GTX 900 - GTX 1000	NVIDIA 500+	NVIDIA 550+
RTX 2000 - RTX 5000	NVIDIA 500+	NVIDIA 550+
AMD		
HD 7000 - HD 8000	*Unsupported*	Mesa
R 200, R 300	*Unsupported*	Mesa
RX 400 - RX 600	AMD Official/latest	AMD Official/latest, Mesa
RX Vega	AMD Official/latest	AMD Official/latest, Mesa
RX 5000 - RX 9000	AMD Official/latest	AMD Official/latest, Mesa
Intel		
Intel 6-10th gen iGPU	*Unsupported*	Mesa
Intel 11th gen iGPU and above	Intel Official/latest	Mesa
Intel Arc	Intel Official/latest	Mesa
Qualcomm		
Adreno X1-85 GPU	Qualcomm next release	*Not tested*

Development tooling

- Testing
  - `blender --gpu-backend {vulkan/opengl/metal}`
- Simulate other GPUs
  - `blender --debug-gpu-force-workarounds`
  - `blender --debug-gpu-vulkan-local-read`


Preferences
Interface
Viewport
Lights
Editing
Animation
Get Extensions
Add-ons
Themes
Input
Navigation
Keymap
System
Save & Load
File Paths
Experimental
Cycles Render Devices
None
CUDA
HIP
Display Graphics
Backend
Vulkan (experimental)
Device
Quadro RTX 6000
Auto
AMD Radeon Graphics (RADV RAPHAEL_MENDOCINO)
AMD Radeon Pro W7900 (RADV NAVI31)
Quadro RTX 6000
llvmpipe (LLVM 17.0.6, 256 bits)
Operati
Register
Unregister
For All Users
Network
Allow Online Access
Time Out
10
Connection Limit
5
Memory & Limits
Undo Steps
32
Undo Memory Limit
0
Global Undo

Tools Help

Vulkan Layers Management

Layers Controlled by the Vulkan Configurator

Apply only to the Vulkan Applications List

Edit Applications...

Keep Vulkan Configurator running in system tray

Vulkan Layers Configurations

API dump
Crash Diagnostic
Frame Capture
Portability
Synchronization
Validation
Validation with logging

New...
Edit...
Duplicate
Remove

Vulkan Application Launcher

Application	vkcube
Executable	${VULKAN_SDK}/bin/vkcube
Working Directory	${VULKAN_SDK}/bin
Command-line Arguments	--suppress_popups
Output Log	${VK_LOCAL}/vkcube.txt

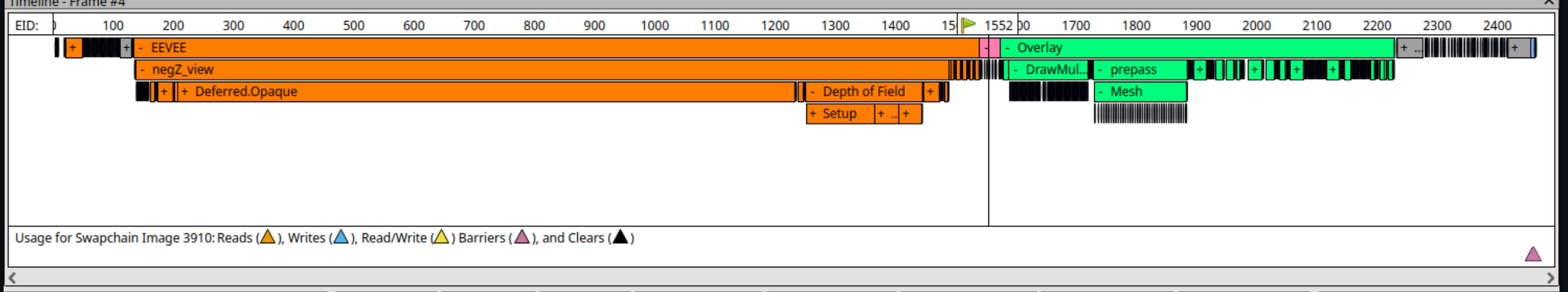
Clear Status Vulkan Loader: layer Launch

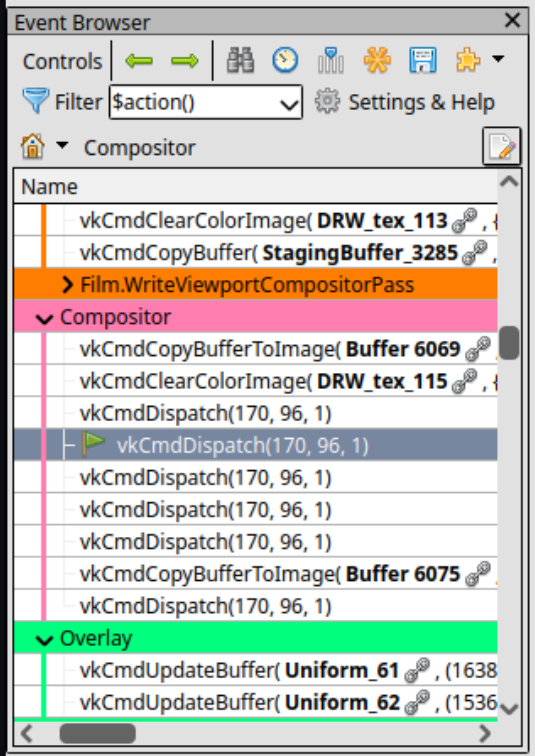
Validation Settings

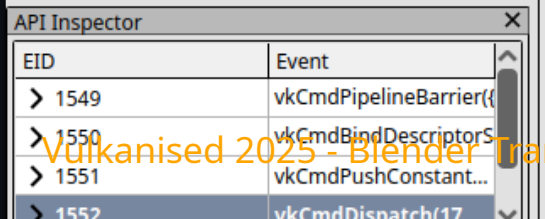
VK_LAYER_KHRONOS_validation

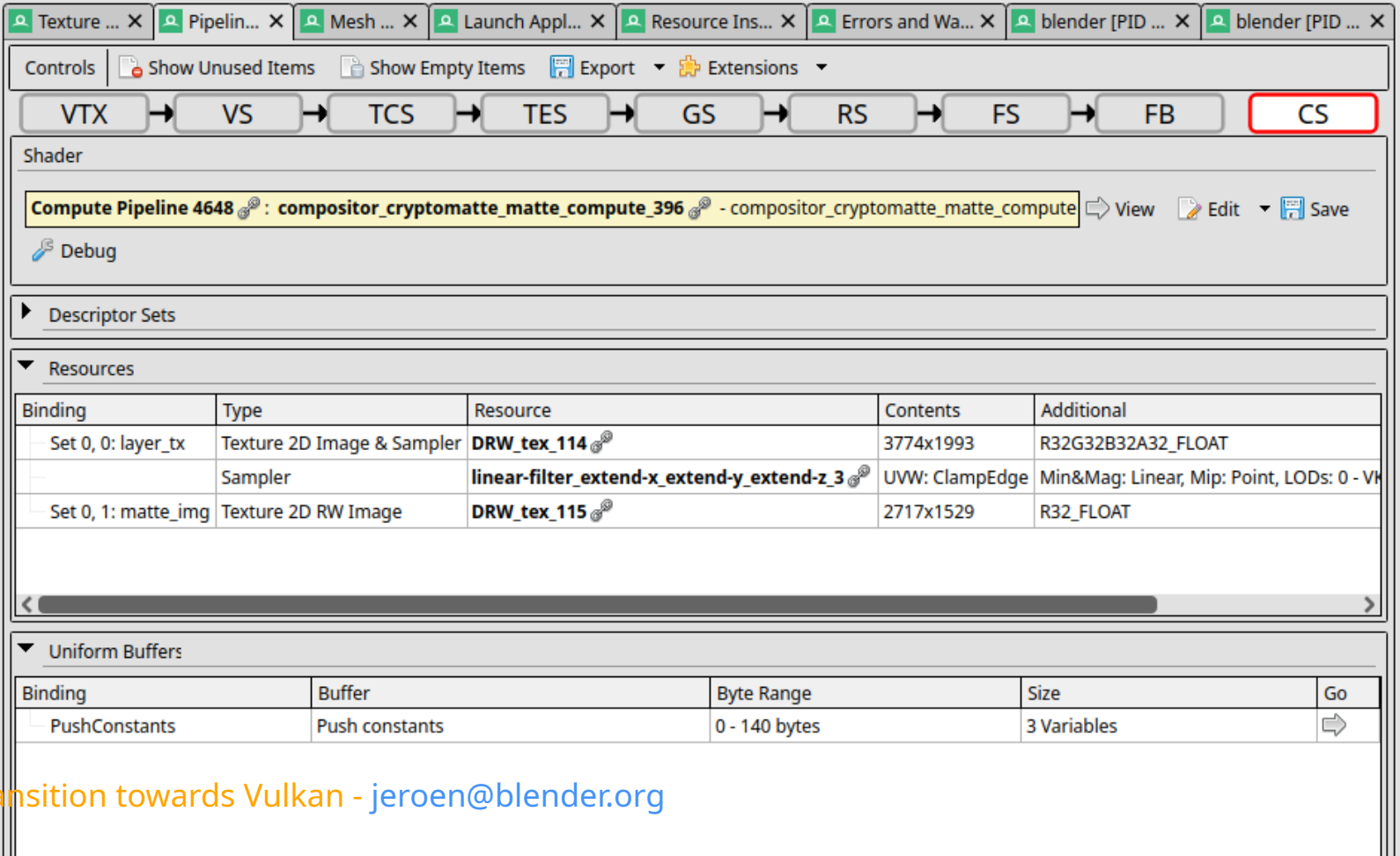
User-Defined Settings

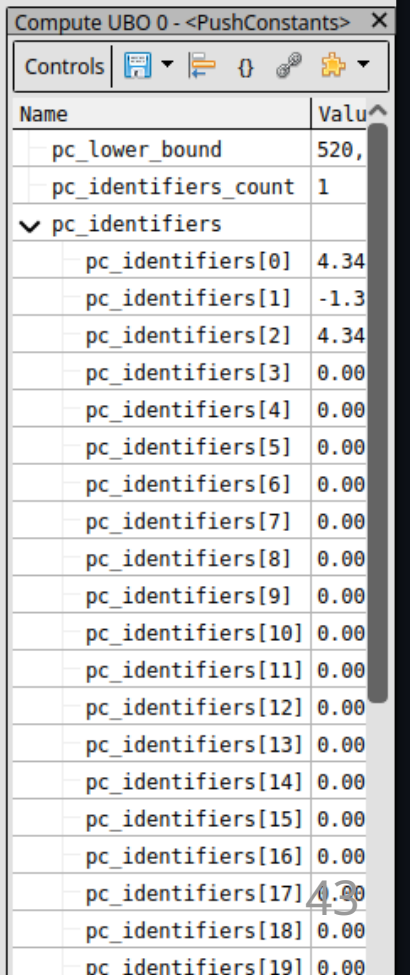
- Validation Areas
  - Fine Grained Locking
  - Core
  - Handle Wrapping
  - Object Lifetime
  - Stateless Parameter
  - Thread Safety
  - Synchronization
  - Debug Printf
  - GPU Assisted Validation
  - Best Practices
    - ARM-specific best practices
    - AMD-specific best practices
    - IMG-specific best practices
    - NVIDIA-specific best practices
- Debug Action
  - Log Message
    - Log Filename
      - stdout
  - Break
- Message Severity
  - Info
  - Warning
  - Performance
  - Error
- Limit Duplicated Messages


Timeline - Frame #4
EID:
EEVEE
negZ_view
Deferred.Opaque
Depth of Field
Setup
Overlay
DrawMul...
prepass
Mesh
Usage for Swapchain Image 3910: Reads ( ), Writes ( ), Read/Write ( ) Barriers ( ), and Clears ( )


Event Browser
Controls
Filter $action()
Settings & Help
Compositor
Name
vkCmdClearColorImage(DRW_tex_113
vkCmdCopyBuffer(StagingBuffer_3285
Film.WriteViewportCompositorPass
Compositor
vkCmdCopyBufferToImage(Buffer 6069
vkCmdClearColorImage(DRW_tex_115
vkCmdDispatch(170, 96, 1)
vkCmdDispatch(170, 96, 1)
vkCmdDispatch(170, 96, 1)
vkCmdDispatch(170, 96, 1)
vkCmdDispatch(170, 96, 1)
vkCmdCopyBufferToImage(Buffer 6075
vkCmdDispatch(170, 96, 1)
Overlay
vkCmdUpdateBuffer(Uniform_61
vkCmdUpdateBuffer(Uniform_62


API Inspector
EID
Event
1549
vkCmdPipelineBarrier(
1550
vkCmdBindDescriptorS
1551
vkCmdPushConstant...
1552
vkCmdDispatch(17


Texture ...
Pipelin...
Mesh ...
Launch Appl...
Resource Ins...
Errors and Wa...
blender [PID ...
blender [PID ...
Controls
Show Unused Items
Show Empty Items
Export
Extensions
VTX
VS
TCS
TES
GS
RS
FS
FB
CS
Shader
Compute Pipeline 4648 : compositor_cryptomatte_matte_compute_396 - compositor_cryptomatte_matte_compute
View
Edit
Save
Debug
Descriptor Sets
Resources
Binding
Type
Resource
Contents
Additional
Set 0, 0: layer_tx
Texture 2D Image & Sampler
DRW_tex_114
3774x1993
R32G32B32A32_FLOAT
Sampler
linear-filter_extend-x_extend-y_extend-z_3
UVW: ClampEdge
Min&Mag: Linear, Mip: Point, LODs: 0 - VK
Set 0, 1: matte_img
Texture 2D RW Image
DRW_tex_115
2717x1529
R32_FLOAT
Uniform Buffers
Binding
Buffer
Byte Range
Size
Go
PushConstants
Push constants
0 - 140 bytes
3 Variables


Compute UBO 0 - <PushConstants>
Controls
Name
Valu
pc_lower_bound
520,
pc_identifiers_count
1
pc_identifiers
pc_identifiers[0]
4.34
pc_identifiers[1]
-1.3
pc_identifiers[2]
4.34
pc_identifiers[3]
0.00
pc_identifiers[4]
0.00
pc_identifiers[5]
0.00
pc_identifiers[6]
0.00
pc_identifiers[7]
0.00
pc_identifiers[8]
0.00
pc_identifiers[9]
0.00
pc_identifiers[10]
0.00
pc_identifiers[11]
0.00
pc_identifiers[12]
0.00
pc_identifiers[13]
0.00
pc_identifiers[14]
0.00
pc_identifiers[15]
0.00
pc_identifiers[16]
0.00
pc_identifiers[17]
0.00
pc_identifiers[18]
0.00

Custom capturing

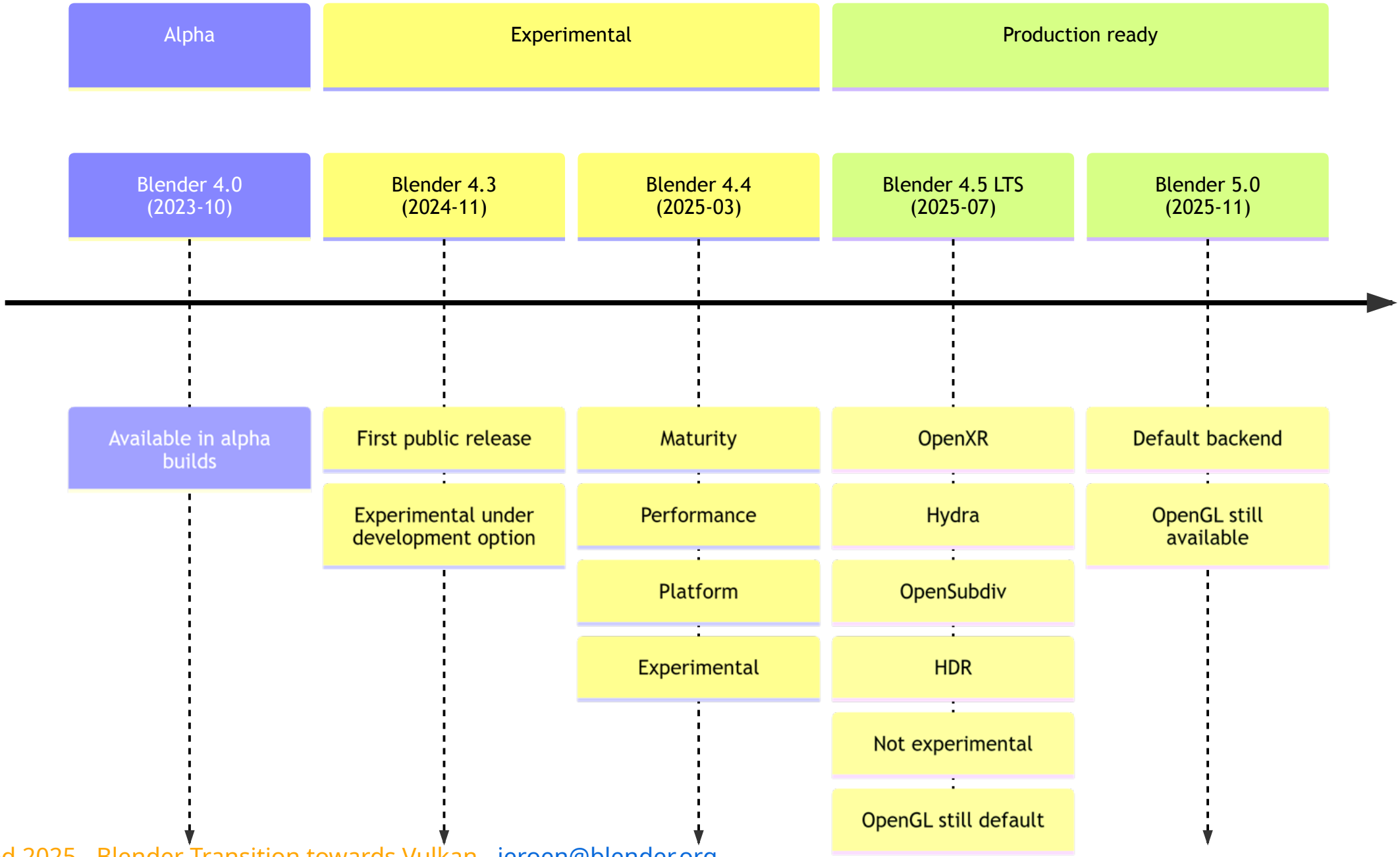
```
blender --debug-gpu-renderdoc
```

```
#include "GPU_debug.hh"

void MyRenderer::render() {
  GPU_debug_capture_begin(__func__);
  GPU_debug_group_begin("MyRenderer");

  /* Do stuff */

  GPU_debug_group_end();
  GPU_debug_capture_end();
}
```


Alpha
Experimental
Production ready
Blender 4.0 (2023-10)
Blender 4.3 (2024-11)
Blender 4.4 (2025-03)
Blender 4.5 LTS (2025-07)
Blender 5.0 (2025-11)
Available in alpha builds
First public release
Experimental under development option
Maturity
Performance
Platform
Experimental
OpenXR
Hydra
OpenSubdiv
HDR
Not experimental
OpenGL still default
Default backend
OpenGL still available

Become part of the development community

- New to open source development: Google summer of code
- developer.blender.org
- chat.blender.org

Kind of developments

- `VK_EXT_descriptor_buffer`
- `VK_EXT_graphics_pipeline_library`
- Local read support for non tiled architectures.
- Code optimizations
- ...

Time for questions