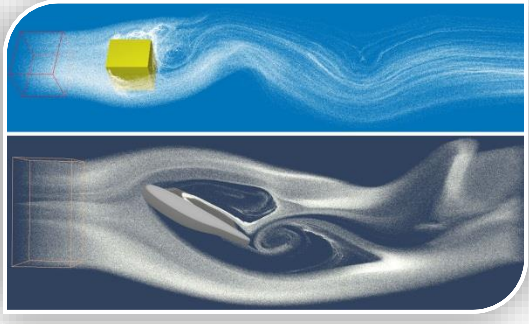
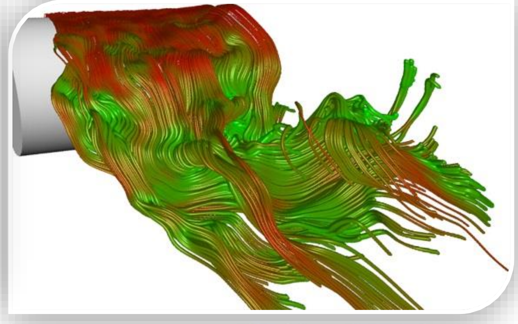
Master Practical Course

# Interactive Visual Data Analysis





tum.3D

computer graphics & visualization

# Today

- Assignment 12: Order Independet Transparency

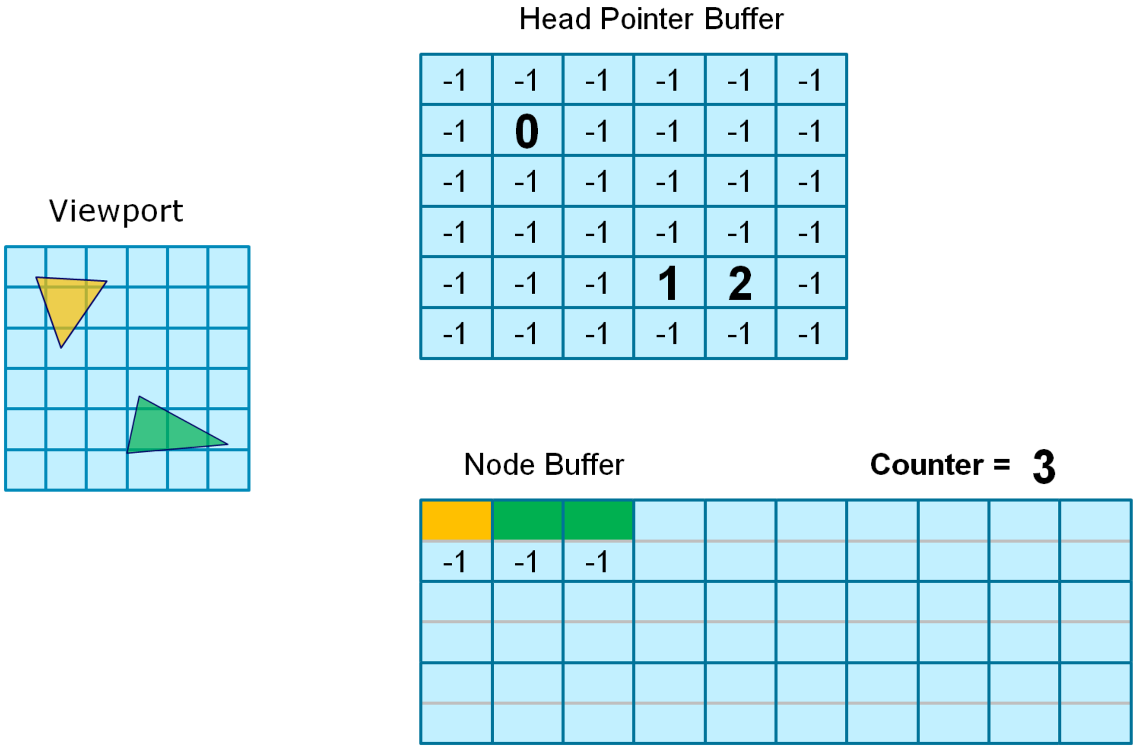
**Real-Time Concurrent Linked List Construction on the GPU**,
J.C. Yang, J. Hensley, H. Grün, and N. Thibieroz,
*Computer Graphics Forum*. Vol. 29. No. 4., 2010.

# Concurrent Linked Lists

- Links: Paper, Slides
- State-of-the-art for OIT (order independent transparency)
- Alternatives:
  - Depth Peeling
  - Stencil routed K-Buffer
  - …
- Transparent objects are rendered in two passes
  - 1) Draw objects and store fragments in an unsorted list (per-pixel)
  - 2) Sort lists, blend and draw onto render target

# Linked List Data Structure

- Global Counter
  - `RWStructuredBuffer` internal counter
- Head Pointers
  - 2D integer texture
- Node Buffer
  - Multiple 2D textures
  - Or structured buffer



Head Pointer Buffer

| | | | | | |
|---|---|---|---|---|---|
| -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | 0 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | 1 | 2 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 |

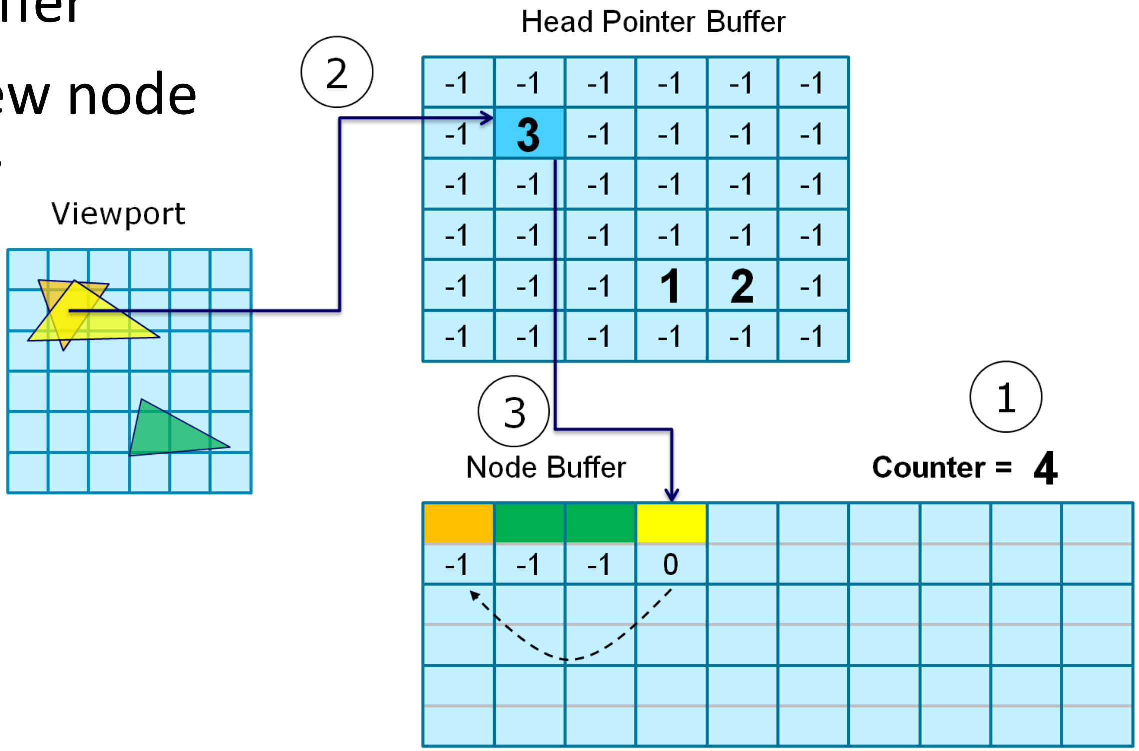
Node Buffer

Counter = 3

| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 🟨 | 🟩 | 🟩 | | | | | | | |
| -1 | -1 | -1 | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |
| | | | | | | | | | |

# Linked List Data Structure

- Inserting a fragment
  - Step 1) Get next free space and increment the counter.
  - Step 2) Atomic exchange with the head pointer buffer
  - Step 3) Insert the new node into the node buffer



Head Pointer Buffer

| | | | | | |
|---|---|---|---|---|---|
| -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | 3 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 |
| -1 | -1 | -1 | 1 | 2 | -1 |
| -1 | -1 | -1 | -1 | -1 | -1 |

Node Buffer

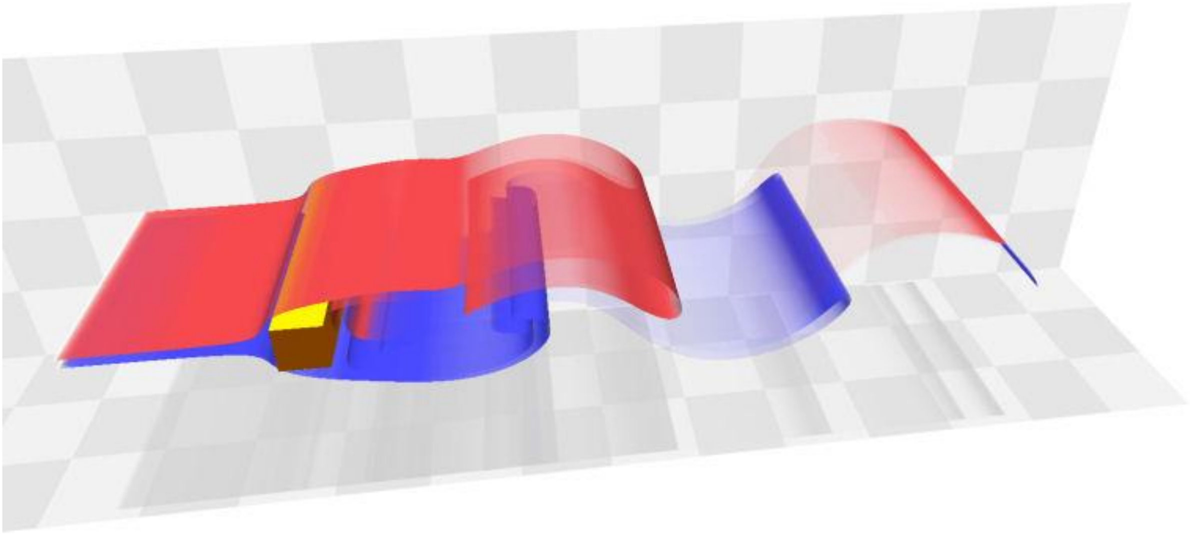
| | | | | | | | | | |
|---|---|---|---|---|---|---|---|---|---|
| 🟧 | 🟩 | 🟩 | 🟨 | | | | | | |
| -1 | -1 | -1 | 0 | | | | | | |

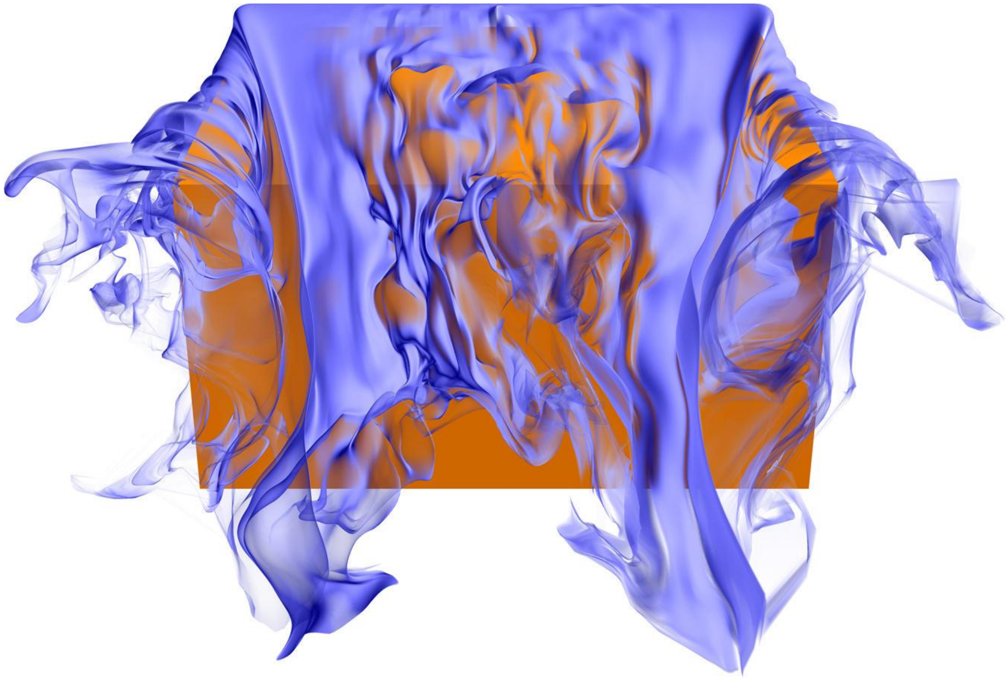
Counter = 4

# Sorting Pass

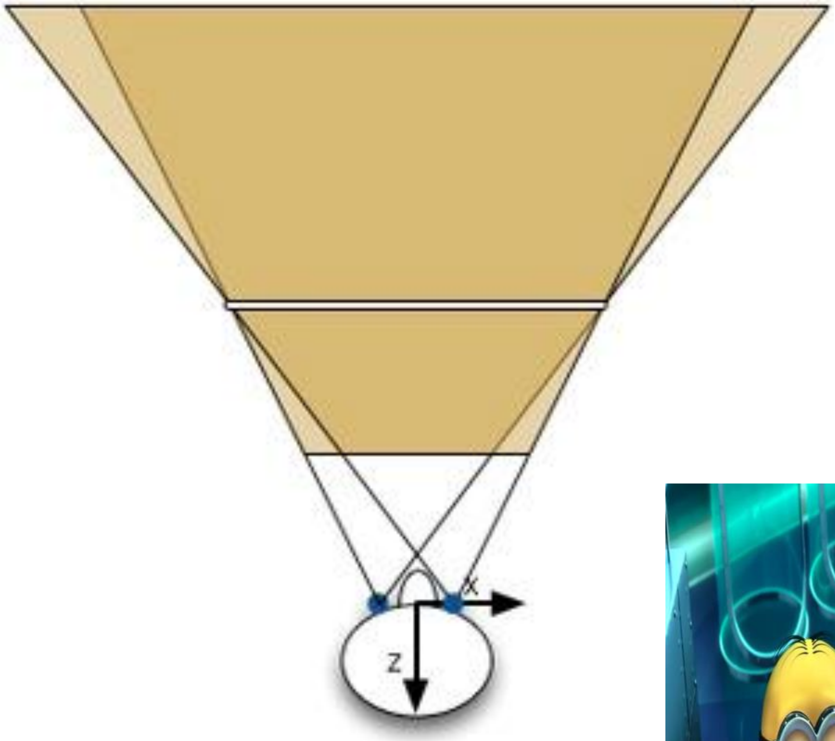
- Sorting pass
  - Fullscreen quad
  - In PS:
    - Load first k fragments in list
    - Sort (insertion sort) → List of k nearest fragments
    - Blend

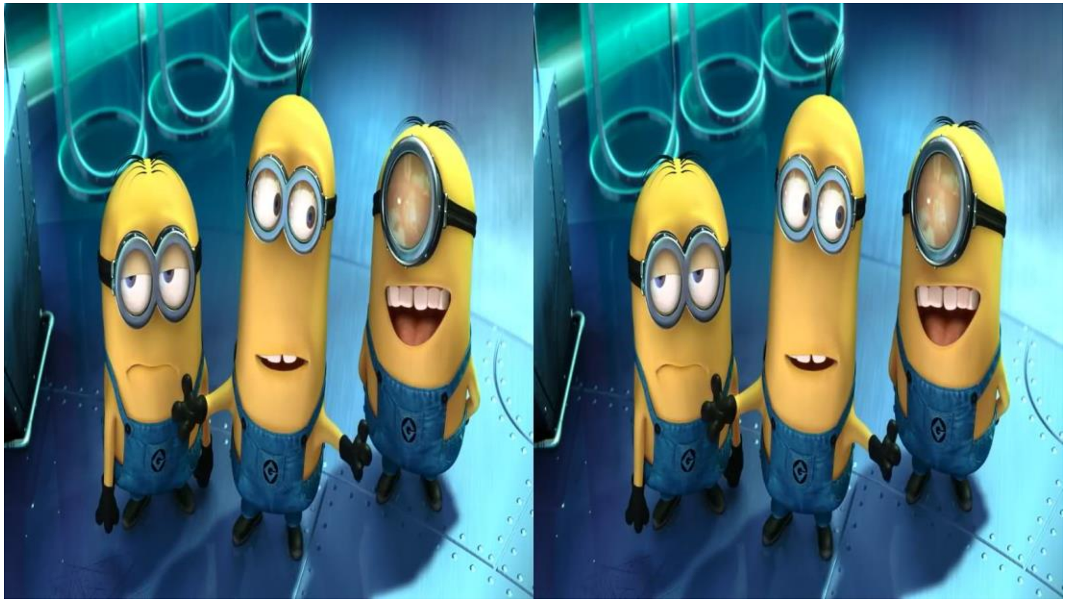
# Multiple, Shaded Smoke Surfaces





# Optional: Stereo Rendering





# Q&A

## Questions ?