

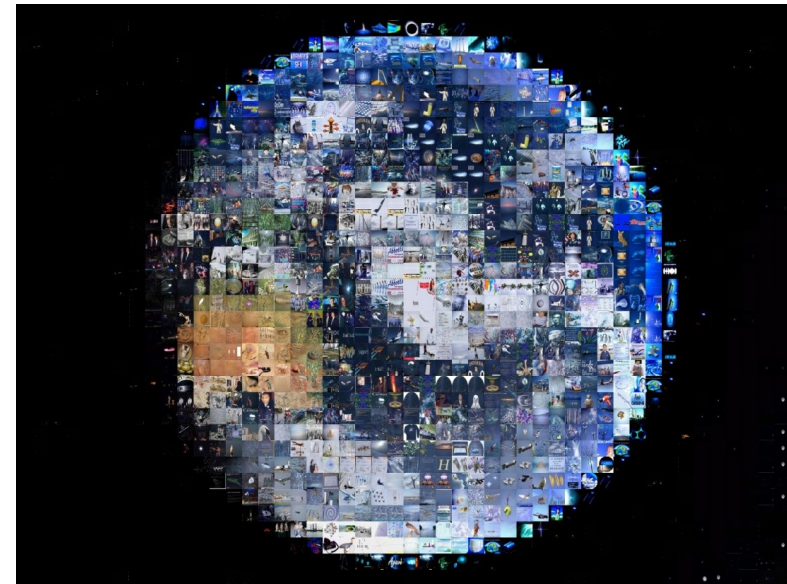
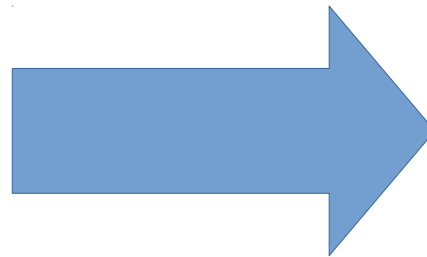
ActivePointers:

Software Address Translation Layer on GPUs

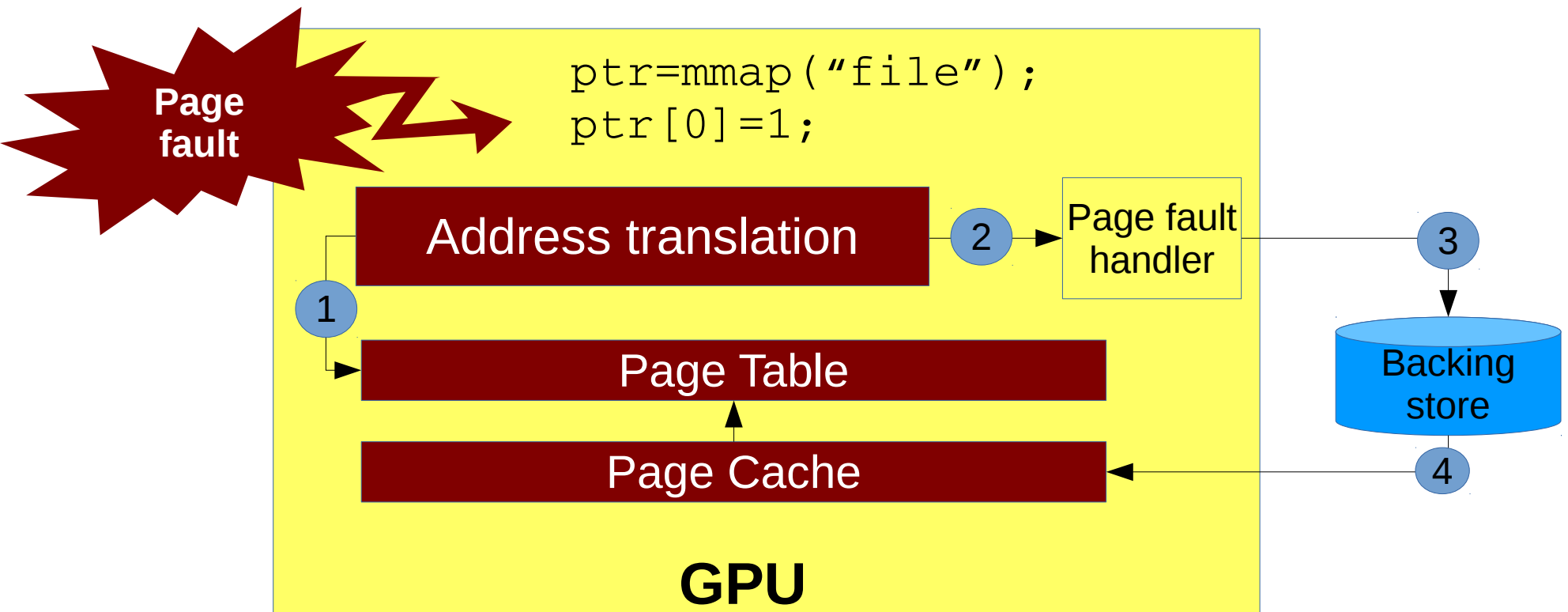
June 22 10.50 Session 9A GPUs 2

- Goal: **huge data set** processing on GPUs
- Challenge: **data-driven** file access

Image collage from **40 GBs** of tiny files

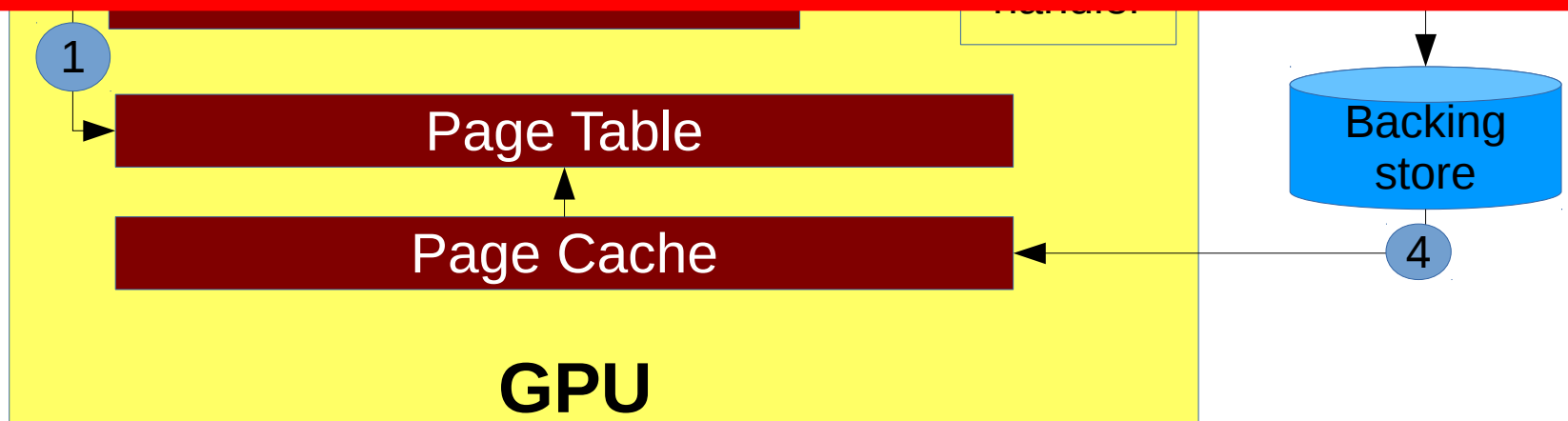


- Solution: **native support** for file `mmap()` on GPUs
- Required: **GPU-centric** Virtual Memory Management and GPU Page Fault execution



- Solution: **native support** for file `mmap()` on GPUs
- Required: **GPU-centric** Virtual Memory Management and GPU Page Fault execution

Challenge: lack of hardware support



Solution: Software Address Translation

- Key idea

Translation is cached in

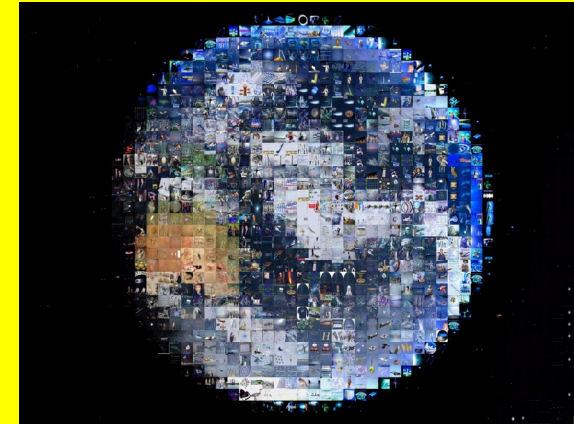
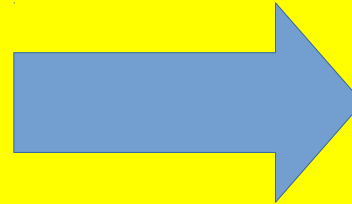
hardware registers (no TLB lookup)

- Key insight

Translation overhead **hidden**

behind memory accesses of **other threads**

- End-to-end evaluation on K80 GPU



Pointer access (via mmap) to 40GB DB file in GPU memory

- **Zero translation overhead**
- **2.6x** over 12 CPU cores with 256-bit AVX
- **3.5x** over CPU + GPU

ActivePointers:

Software Address Translation Layer on GPUs

June 22 10.50 Session 9A GPUs 2