BUFFER OVERVIEW
This document shows the usage of buffers during one frame

INITIALIZATION STAGE	MODIFICATION STAGE	RELEASE STAGE

Init Flags

- sets all flags to its default values

Update Alive Buffers

- copies pointers of living particles from AliveSwapBuffer (that was filled in previous frame) to AliveBuffer
- updates counter in AliveCounterBuffer to the number of living particles

Reset Force

- resets the forces that were used to calculate the position of all living particles in the last frame

Emitter

- every emitter has its own region in the ParticleBuffer
- emits into slots that are still empty or where particles are marked as dead

- adds pointers to new particles in AlivePointerBuffer
- increments AliveCounterBuffer for every written particle

- uses the EmitterCounterBuffer to regulate the number of emissions

An arbitrary amount of emitters can be used. Each emitter occupies its own region in the ParticleBuffer and the total buffersize automatically adapts to the sum of all emitter binsizes.

Selections

- pointers to living particles are added to the SelectionPointerBuffer when the belonging particles were selected with a selector
- the SelectionCounterBuffer is incremented per selected particle
- the ID of the successful selector is stored per particle in the SelectionIndexBuffer
- Selections set a SelectionFlag in the FlagBuffer (so that subsequent modifiers use the SelectionPointerBuffer instead of the AlivePointerBuffer)

Selectors can be logically combined. Note that the ID of the FIRST successful selector is stored in the SelectionIndexBuffer.

Modifications

- dependent on the SelectionFlag in the FlagBuffer modifications are applied to particles addressed by the AlivePointerBuffer or the SelectionPointerBuffer
- if modifiers are applied to selections, the SelectionIndexBuffer can be used to address data in DynamicBuffers (which provide data for the modifiers)

Selectors and Modificators can introduce arbitrary attributes. The particlesystem-structure automatically adapts to the them, so it is very easy to introduce completely new attributes by writing own Selectors and/or modificators.

Update Alive Buffers

- pointers to alive particles are copied from AlivePointerBuffer to AliveSwapBuffer

Iterator

- positions, ages & lifespans get updated for each living particle

---

ParticleBuffer
- stores all particle data

AlivePointerBuffer
- stores pointers to living particle
- used in nearly all shaders to address particles efficiently

AliveCounterBuffer
- stores the number of alive particles
- used in nearly all shaders to address particles

AliveSwapBuffer
- is used to copy pointers to alive particles between successive frames

EmitterCounterBuffer
- used in emitters to control emission of particles

SelectionPointerBuffer
- stores pointers to selected particles
- used in modifier shaders to address selected particles

SelectionCounterBuffer
- stores the number of selected particles
- used in modifier shaders to address selected particles

SelectionIndexBuffer
- stores the index of the selector that sucessfully selected a particle
- used in modifier shaders to assign data to selected particles

FlagBuffer
- stores flags to control behavior of shaders