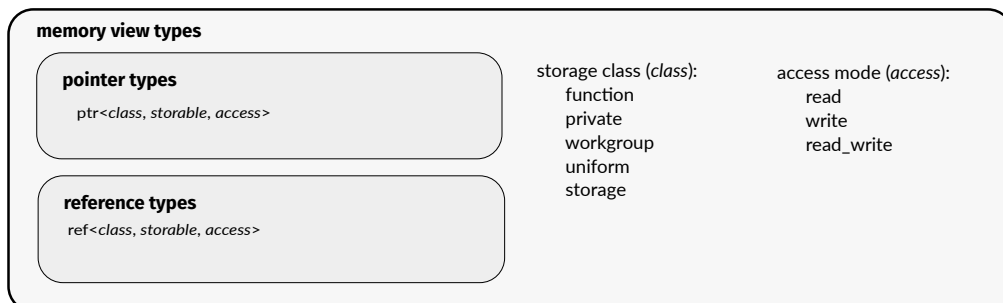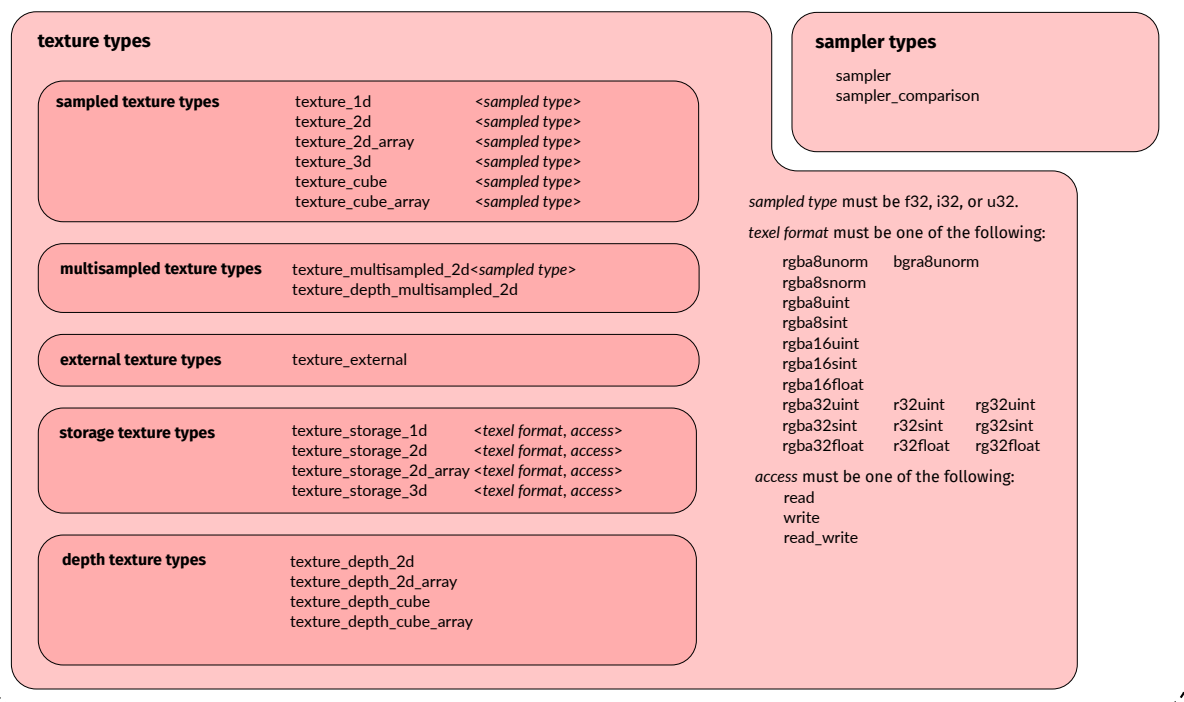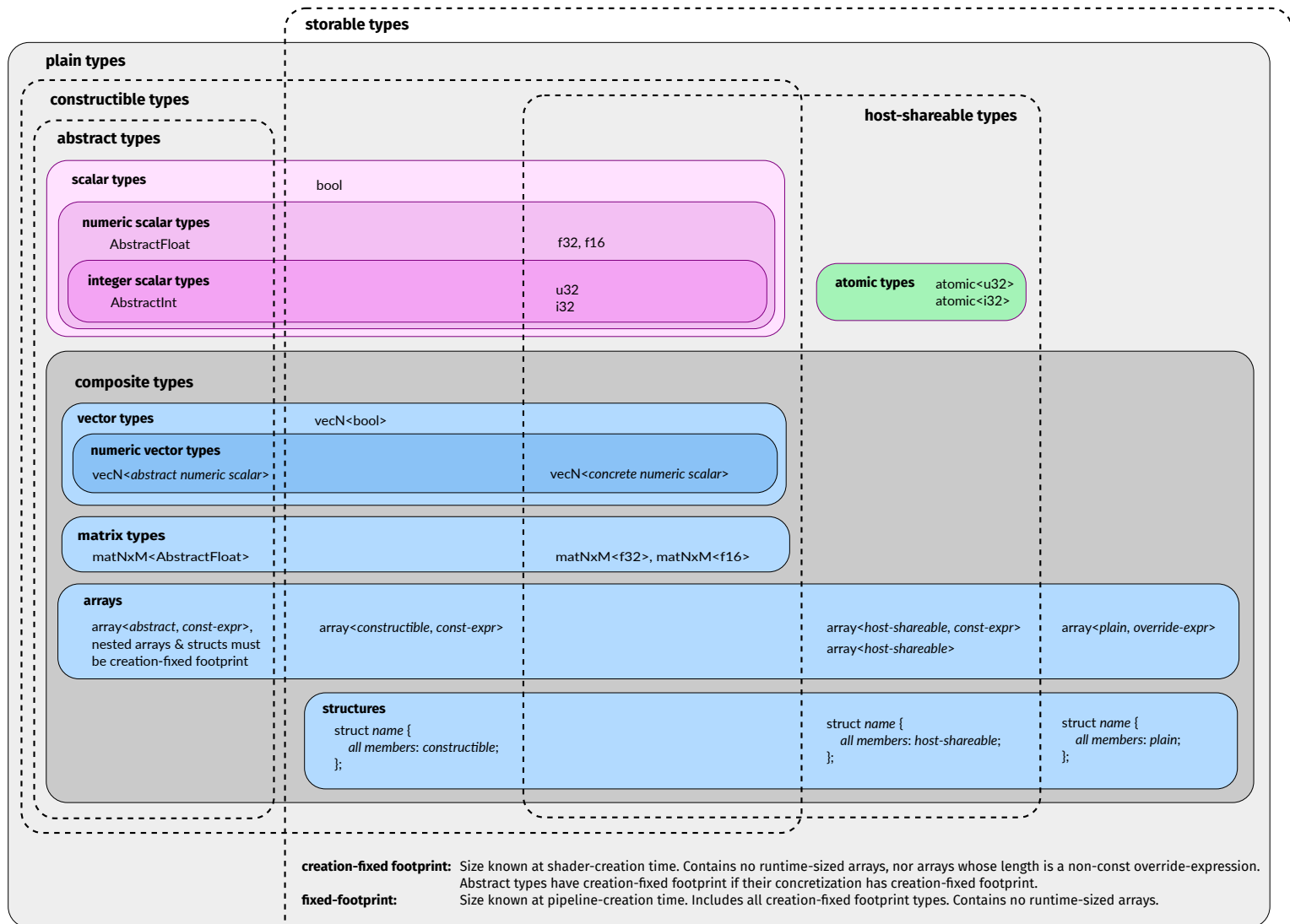# WebGPU Shading Language (WGSL) type categories

storable types

plain types

constructible types

abstract types

scalar types bool

numeric scalar types
AbstractFloat   f32, f16

integer scalar types
AbstractInt   u32
i32

host-shareable types

atomic types   atomic<u32>
atomic<i32>

composite types

vector types   vecN<bool>

numeric vector types
vecN<*abstract numeric scalar*>   vecN<*concrete numeric scalar*>

matrix types
matNxM<AbstractFloat>   matNxM<f32>, matNxM<f16>

arrays
array<*abstract, const-expr*>,   array<*constructible, const-expr*>   array<*host-shareable, const-expr*>   array<*plain, override-expr*>
nested arrays & structs must   array<*host-shareable*>
be creation-fixed footprint

structures
struct *name* {   struct *name* {   struct *name* {
   all members: constructible;   *all members: host-shareable*;   *all members: plain*;
};   };   };

creation-fixed footprint: Size known at shader-creation time. Contains no runtime-sized arrays, nor arrays whose length is a non-const override-expression. Abstract types have creation-fixed footprint if their concretization has creation-fixed footprint.

fixed-footprint: Size known at pipeline-creation time. Includes all creation-fixed footprint types. Contains no runtime-sized arrays.

texture types

sampled texture types
texture_1d	<*sampled type*>
texture_2d	<*sampled type*>
texture_2d_array	<*sampled type*>
texture_3d	<*sampled type*>
texture_cube	<*sampled type*>
texture_cube_array	<*sampled type*>

multisampled texture types
texture_multisampled_2d<*sampled type*>
texture_depth_multisampled_2d

external texture types
texture_external

storage texture types
texture_storage_1d	<*texel format, access*>
texture_storage_2d	<*texel format, access*>
texture_storage_2d_array	<*texel format, access*>
texture_storage_3d	<*texel format, access*>

depth texture types
texture_depth_2d
texture_depth_2d_array
texture_depth_cube
texture_depth_cube_array

sampler types

sampler
sampler_comparison

sampled type must be f32, i32, or u32.

texel format must be one of the following:

rgba8unorm	bgra8unorm	
rgba8snorm		
rgba8uint		
rgba8sint		
rgba16uint		
rgba16sint		
rgba16float		
rgba32uint	r32uint	rg32uint
rgba32sint	r32sint	rg32sint
rgba32float	r32float	rg32float

access must be one of the following:
   read
   write
   read_write

memory view types

pointer types
ptr<*class, storable, access*>

reference types
ref<*class, storable, access*>

storage class (*class*):
   function
   private
   workgroup
   uniform
   storage

access mode (*access*):
   read
   write
   read_write

(N, M = 2, 3, 4)